

Big Data

Günümüzde teknolojinin hızla gelişmesiyle bilgisayar, cep telefonu, tablet gibi akıllı cihazların ve internet teknolojisinin hızla yayılması ile üretilen veriler artmış ve “büyük veri” kavramı ortaya çıkmıştır.

Tanım olarak Big Data, verinin analiz edilip sınıflandırılmış, anlamlı ve işlenebilir hale dönüştürülmüş halidir.

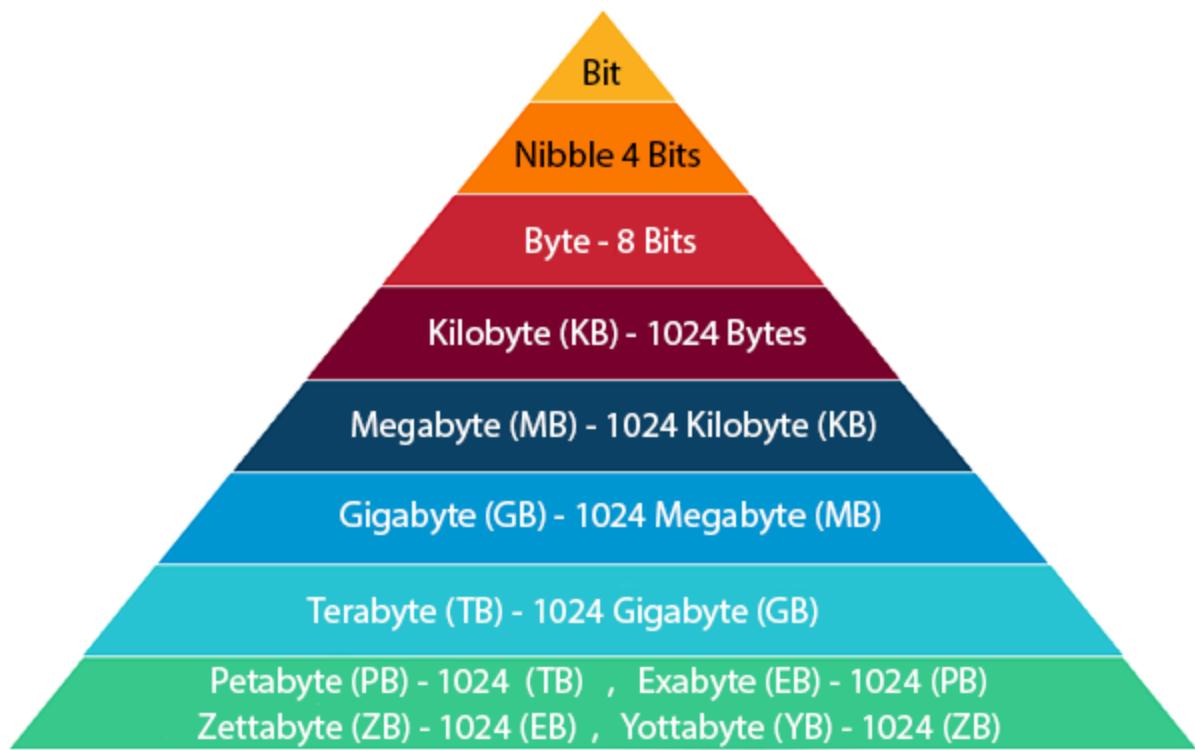
Büyük veri; verinin hacmi, veri hızı, veri çeşitliliği, verinin düzensiz veya yanlış olması, verinin değerli olması olmak üzere beş bileşenden oluşur.



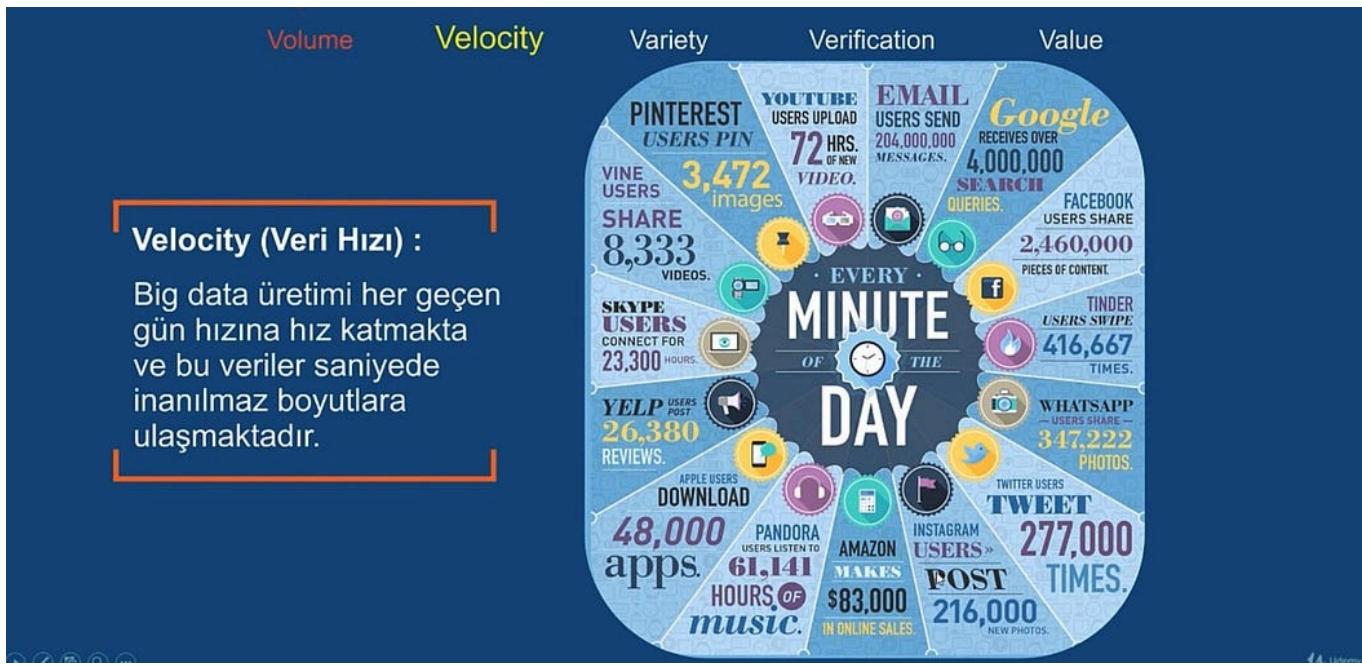
Veri hacmi

Verinin yüksek hacimli olduğunu belirtir.

Örneğin bir uçağın motorunda ve diğer bileşenlerinde milyonlarca sensör mevcuttur. Bu sensörler uçakların yaptığı her bir hareketi anlık olarak toplayarak terabaytlar seviyesinde veri üretmektedir.



Velocity(Veri Hızı):



Velocity (Veri Hızı) :

Big data üretimi her geçen gün hızına hız katmakta ve bu veriler saniyede inanılmaz boyutlara ulaşmaktadır.

Adı üzerinde olduğu gibi Velocity, Veri Hızı anlamına gelmektedir. Bunun anlamı ise saniyeler veya dakikalar içerisinde size ne kadar veri ulaştığıdır. Özellikle ve özellikle herkesin bildiği gibi Sosyal Medya'da binlerce işlem yapılmıyor. Burada gördüğünüz bir Sosyal Medya infografiği.

Örneğin;

- Google'da dakikada 4 milyon arama.
- Youtube'da ki kullanıcılar dakikada 72 saatlik video yükleyipmiş.
- Facebook'da dakikada 2 milyon 460 bin içerik paylaşılıyormuş.
- Amazon dakikada 83 bin dolarlık online satış yapıyormuş.
- Twitter'da dakikada 277 bin tweet atılıyormuş.

Infograf'ta da gördüğünüz gibi inanılmaz bir veri akışı yalnızca dakikalar içerisinde oluşuyor ve işlenmek için size ulaşıyor.

Variety(Veri Çeşitliliği):

Volume

Velocity

Variety

Verification

Value

Variety (Veri Çeşitliliği) :

Verilerin belirli bir yapısı yoktur, genellikle değişkendir. Resimler, ses dosyaları, text dosyaları örnek verilebilir



Big Data'da verilerin belirli bir yapısı veya formatı yoktur. Json, txt, fotoğraf, video veya Sosyal Medya gibi verilerin çeşitliliğini düşünürsek elimizde büyük bir veri yığını olmuş olur. Big Data bunların hepsini aynı formatta toplayıp hepsini işleyebilir.

Verification (Gerçeklik):

Volume

Velocity

Variety

Verification

Value

Verification-Veracity (Doğrulama):

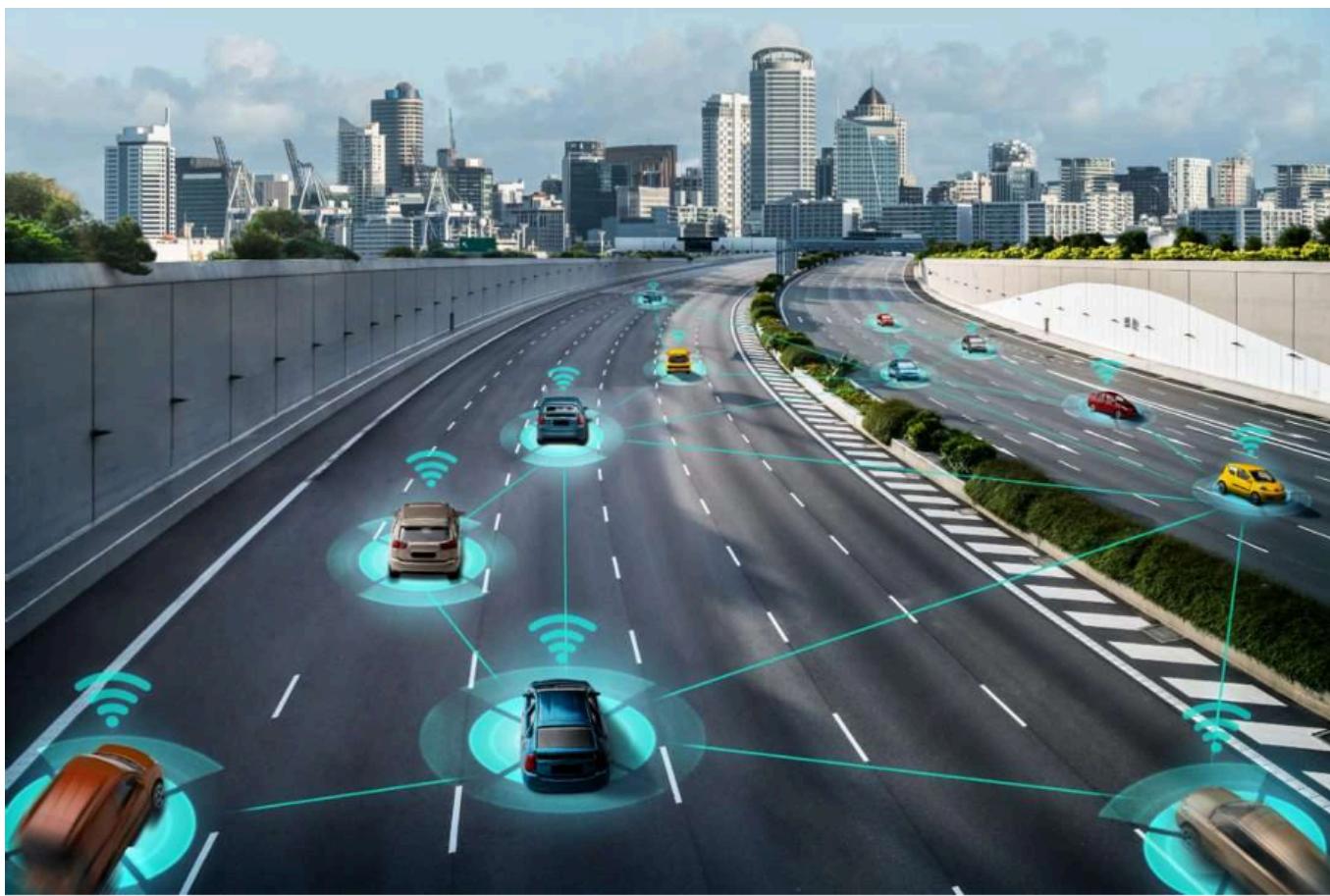
Veriler içerisinde anlamsız kayıtlar olabilir. Anlamsız kayıtlar analizlerimizin sonuçlarını etkilediği için bu kayıtları temizlememiz gerekmektedir.



Big Data içerisinde çok büyük veri yığınları bulunduğuundan içerisinde anlamsız verilerin olması kadar normal bir durum olamaz. Veri Madenciliği'nde bu anlamsız verilere "Kirli yada Gürültücü Veri" deniyor. Bu anlamsız kayıtların sağlıklı bir sonuç alınabilmesi için temizlenmesi gerekmektedir.

Örneğin trafikteki araçların süratlerini değerlendirirken bir araçtan eksi sonuç geliyorsa bu durum yanlıştır. Çünkü fizikte sürat her zaman pozitif bir değerdir. Bu sebepten bu değerlerin

temizlenmesi gerekmektedir.



Value(Değer- Anlamlandırma):

Volume	Velocity	Variety	Verification	Value

Value (Değerli Veri) :
büyük verinin üretimi ve işlenmesi katmanlarında elde edilen verilerin şirketimiz için artı değer sağlıyor olması gerekiyor.

NETFLIX
Topladığımız veriler karlılık,müşteri memnuniyeti ve kalitenin artırılması açısından önemlidir.
Netflix tavsiye sistemi ile satış oranlarını ölçüde arttırmıştır

Google Adsense
Google da aradığımız bir ürünü haber sitelerinde görebiliyoruz Tesadüf mü ?

Aliexpress,Amazon vs
Aliexpress, Amazon gibi platformlar, Big Data teknolojisi sayesinde ürünlerin satış oranını artırmayı başarmıştır.

Big Data'nın en önemli birleşenlerinden biriside Value yani katma değer yaratmasıdır. Bunu şirketlerin yaptığı çalışmalarla örneklendirirsek çok daha iyi anlaşılacaktır.

Netflix:

NETFLIX

NETFLIX

Watch Instantly • Just for Kids • Taste Profile • DVDs

Movies, TV shows, actors, directors, genres



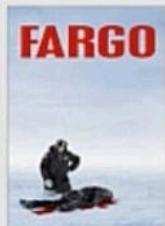
Visually-striking Imaginative Sci-Fi & Fantasy

Based on your interest in...



Critically-acclaimed Quirky Comedies

Based on your interest in...



$$\frac{\text{BBC MINISERIE 1990}}{\text{Michael DOBBS}} + \frac{\text{KEVIN SPACEY} \div \text{MORAL}(\text{MORAL}^2)}{44.000.000 \text{ NETFLIX} + \text{medlemmer}} + \frac{\text{MEDIER} \{ \text{POLITIK} \}}{\sqrt{\frac{\text{SEX}}{\text{LØGN}}} \div \text{HAPPY END}} = \$$$

Netfilix aslında 99–2000 yıllarında kurulmuştur ama parlama zamanı 2008–2009 yıllarında olmuştur.

Yıldızının parlamasına yol açan en büyük teknolojilerden bir taneside Big Data teknolojisidir. Peki bunu nasıl başarabildi!? Bildiğiniz gibi bizde dahil olmak üzere çoğu web sitesi kullanıcılarının loglarını tutmaktadır. Ancak Netflix bunu birkaç adım ileri taşımıştır. Kullanıcının videoları izleme sürelerini, kaç kere duruklarını, ses seviyenizi hatta ve hatta izlediğiniz filmlerdeki oyuncuların sayfalarını beğenip beğenmediğinizi, tweet atıp atmadığınıza kadar birçok bilgiyi tutmaktadır ve sizinle alakalı bir model oluşturmaktadır. Dünyanın herhangi bir yerinde sizinle aynı yada çok benzer bir modele sahip kullanıcıyı eşleştirir.

Örneğin çok sıkı bir Christoper Nolan hayranınız, bilim Kurgu izlemeye bayılıyorsunuz ve tüm filmleri durdurmadan son ses izliyorsunuz. Bu eşleşme sayesinde eşleştiniz ki kişi başka bir film

izlediğinde sizin bu filmi beğenme ihtimaliniz çok yüksek diye size bu filmi öneriyor. Özetle, Netflix bizi bizden daha iyi tanıyor. Bunun altında ise Big Data'nın gücü yatıyor.

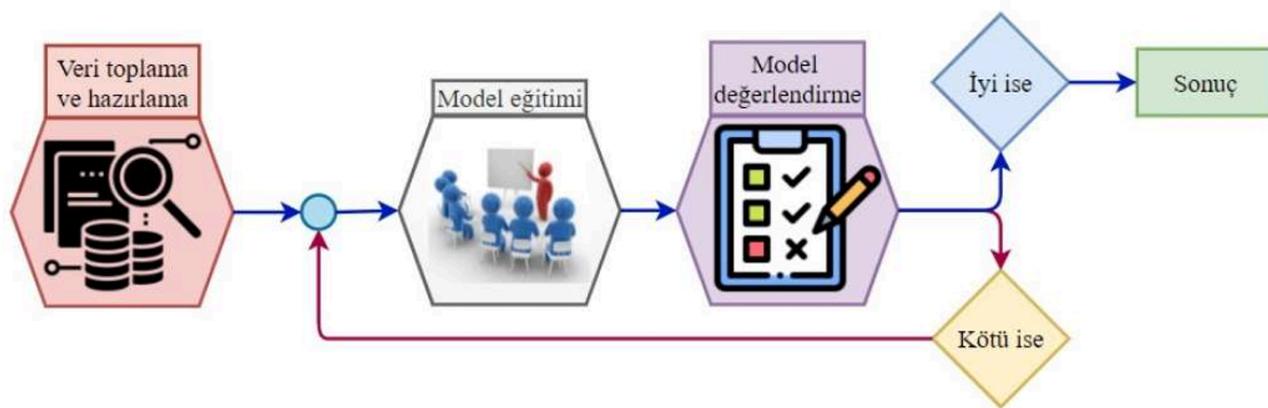
Makine Öğrenmesi Temel Anlatım

Büyük veriyi değerlendirmek için makine öğrenme yöntemleri kullanılır.

Makine öğrenmesi, bilgisayarın meydana gelen bir olay ile ilgili topladığı bilgi ve tecrübeleri öğrenebilmesi amacıyla matematiksel modellerin kullanılmasıdır. Makine öğrenmesinde temel amaç elde edilen veriler ile gelecekte oluşabilecek benzer olaylar hakkında kararlar verebilmek veya geçmişteki durumlar hakkında sonuç oluşturmaktır.

İZLEYELİM :

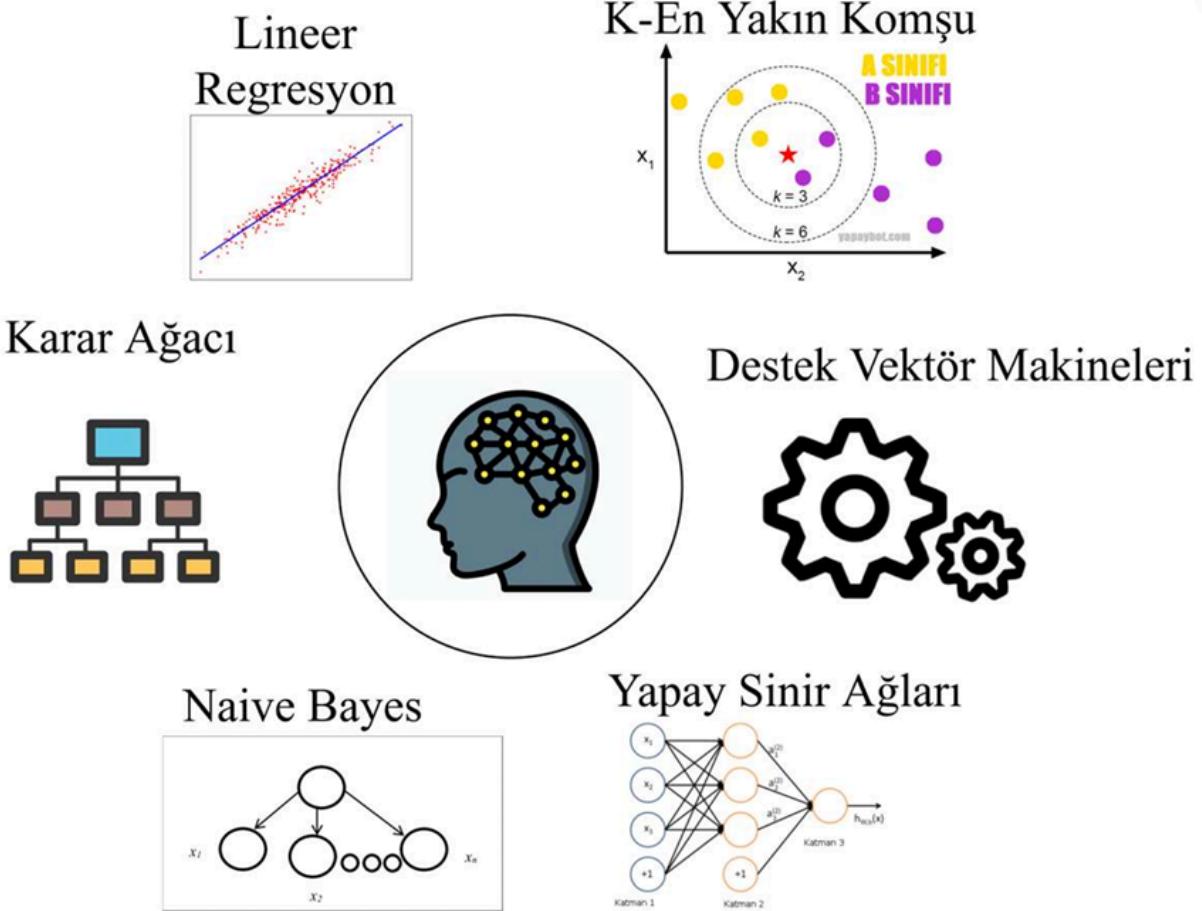
<https://www.youtube.com/watch?v=-O-E1nFm6-A>



Makine öğrenmesi sürecini temel olarak dört aşamadan oluşmaktadır. İlk aşamada, akıllı cihazlardan veriler toplanarak işlenir. İşleme sürecinde uygun olmayan veriler veri setinden çıkarılarak veri bütünlüğü sağlanır. İkinci aşamada, veri setindeki veriler bir kısmı eğitim verisi diğer kısmı test verisi olarak ikiye ayrılır. Eğitim verileri makine öğrenmesindeki modelleri eğitmek için kullanılır. Üçüncü aşamada, modellerden elde edilen sonuçlar test verileri ile analiz edilerek makine öğrenmesi modelinin doğruluğu test edilir. Son aşamada ise, test verilerinden elde edilen sonuçlar değerlendirilir.

Makine Öğrenmesi Teknikleri

Makine öğrenmesi tekniklerinde sıkılıkla naive-bayes algoritmaları, destek vektör makineleri, karar ağacı algoritmaları, k-en yakın komşu algoritmaları kullanılmaktadır.



Makine öğrenmesinde, modellerin eğitilme şekillerine göre temel olarak **Denetimli Öğrenme** (Supervised Learning) ve **Denetimsiz Öğrenme** (Unsupervised Learning) olmak üzere iki ana yaklaşım bulunur.

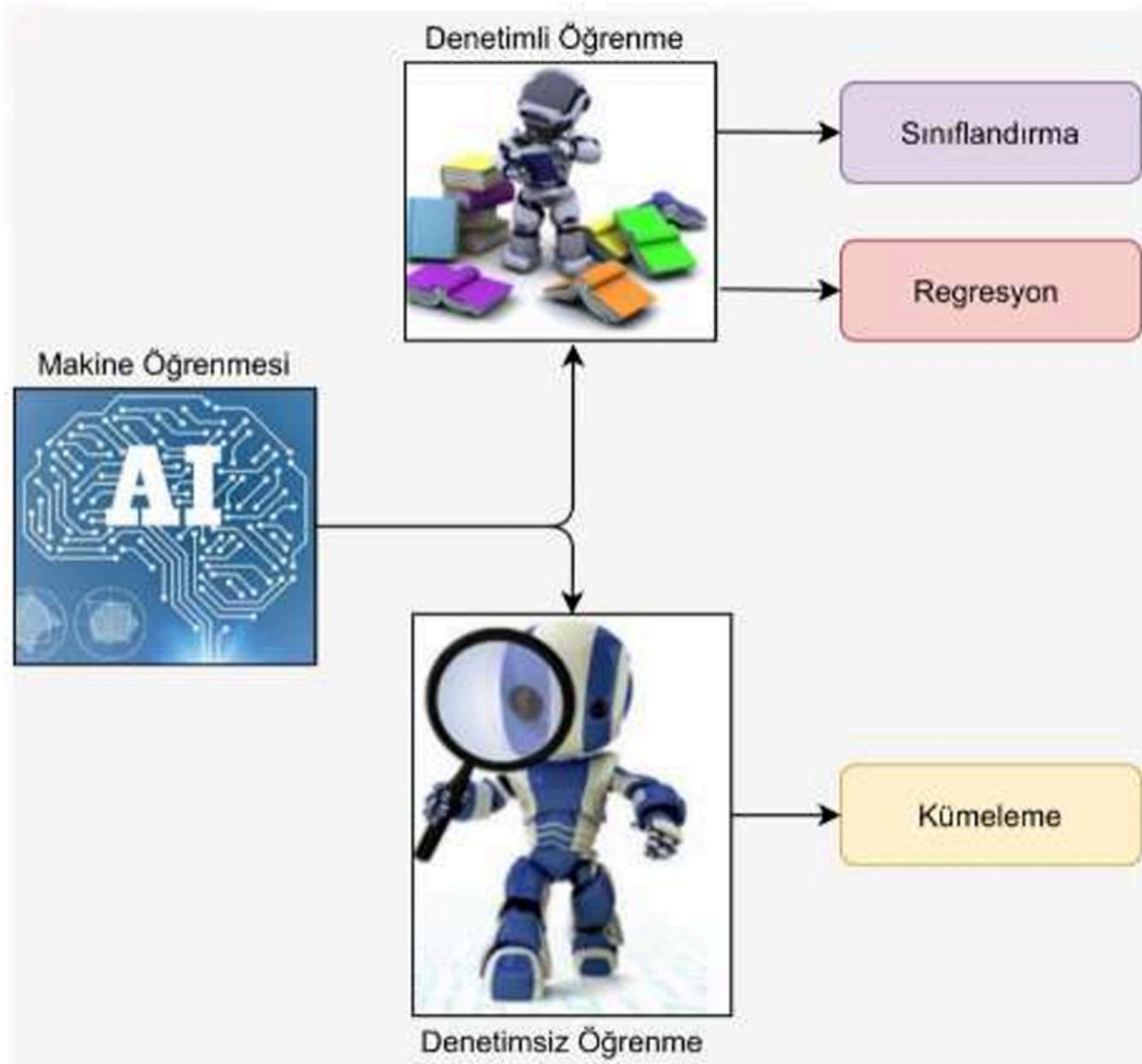
Denetimli Öğrenme (Supervised Learning)

- **Temel Fikir:** Model, **etiketlenmiş** verilerle eğitilir. Yani, girdi verisi ile birlikte bu girdiye karşılık gelen **doğru çıktı** (etiket) modele sağlanır. Tıpkı bir öğretmenin gözetiminde öğrenmek gibidir.
- **Veri Tipi:** **Etiketli** (labeled) veri kümesi kullanılır.
- **Amaç:** Modelin, girdi ve çıktı arasındaki ilişkiyi öğrenerek **yeni girdiler için doğru çıktıyı tahmin etmesidir**.
- **Kullanım Alanları (Örnekler):**
 - **Sınıflandırma (Classification):** Bir veriyi önceden tanımlanmış kategorilerden birine atama.
 - Örn: Bir e-postanın **spam** ya da **spam değil** olarak sınıflandırılması.
 - Örn: Bir resimdeki nesnenin (kedi, köpek vb.) tanımlanması.
 - **Regresyon (Regression):** Sürekli bir çıktı değeri tahmin etme.

- Örn: Evin büyüklüğü, konumu gibi verilere dayanarak **ev fiyatının** tahmin edilmesi.
- Örn: Hava durumuna göre **trafik yoğunluğunun** tahmin edilmesi.

Denetimsiz Öğrenme (Unsupervised Learning)

- **Temel Fikir:** Model, **etiketlenmemiş** verilerle eğitilir. Modele yalnızca girdi verisi verilir ve doğru çıktı bilgisi sağlanmaz. Model, verilerin içindeki **gizli yapıları, örüntüleri** ve ilişkileri kendi başına keşfetmeye çalışır. .

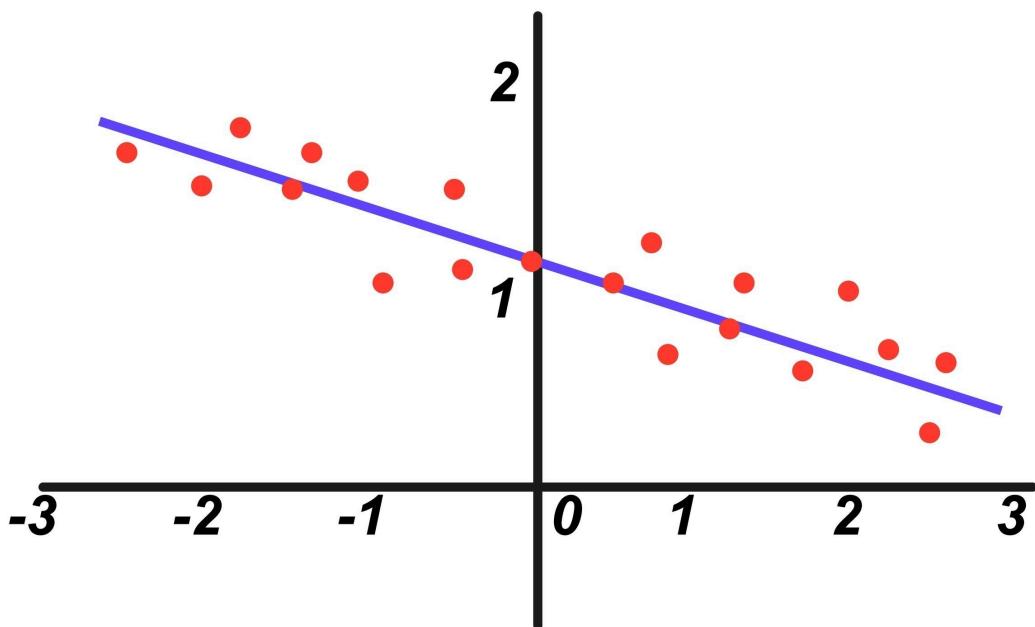


Regresyon (Regression) Lineer Regresyon

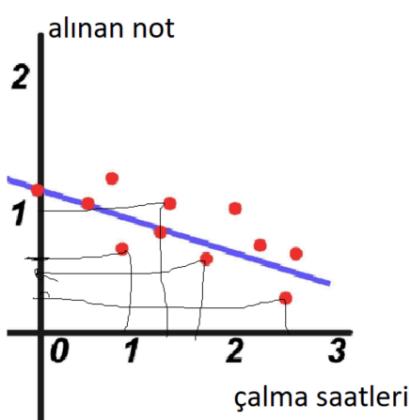
- **Amaç:** **Sürekli (Continuous)** ve **Sayısal** bir çıktı değeri tahmin etmektir.
- **Çıktı Tipi:** Sonsuz sayıda değer alabilen reel sayılar kümesi.

- **Örnekler:**

- **Ev fiyatı tahmini:** Bir evin özelliklerine (büyülüklük, oda sayısı vb.) dayanarak tam fiyatını (örneğin 500.000 TL) tahmin etmek.
- **Sıcaklık tahmini:** Yarınki hava sıcaklığının derece cinsinden (örneğin 25.5°C) tahmin edilmesi.
- **Satış geliri tahmini:** Bir reklam kampanyasından elde edilecek gelir miktarının tahmin edilmesi.
- Veri noktalarına en uygun geçen bir **doğru veya eğri** (en uygun çizgi - best-fit line) bulmayı amaçlar.



Simple linear regression



nekadar etklediği
özellikim
 $y = mx + c$
sabitim ismini
yazana 5
puan

çıktı
başarı notu

$f(x) = mx + c$

(2) → Not

Simple linear regression

```
from sklearn.linear_model import LinearRegression
import pandas as pd

data = pd.read_csv("Student_Marks.csv")

Y = data[["Marks"]]
X = data[["number_courses", "time_study"]]

model = LinearRegression()

model.fit(X, Y)

# 3, 6.335, 32.357

model_tahmin = model.predict([[3, 6.335]]) # = 32.33884377

print("Tahmin :", model_tahmin)

print("Score :", model.score(X, Y))
```

CSV = <https://www.kaggle.com/datasets/yasserh/student-marks-dataset/data>

Sınıflandırma (Classification) Karar Ağaçları - KNN

- **Amaç:** Bir veri noktasını önceden tanımlanmış **ayırık (Discrete)** ve **Kategorik** sınıflardan birine atamaktır.
- **Çıktı Tipi:** Sınırlı ve sonlu sayıda kategori (etiket).
 - Kedi, Köpek, Kuş veya 0, 1

- Örnekler:
 - **Spam filtresi:** Bir e-postanın **Spam** veya **Spam Değil** olarak etiketlenmesi.
 - **Hastalık teşhisi:** Hastanın semptomlarına göre bir hastalığa **sahip olup olmadığı** (**Evet/Hayır**) veya hangi hastalık **tipine** sahip olduğu.
 - **Görüntü tanıma:** Bir resimde **Kedi** mi yoksa **Köpek** mi olduğunu belirleme.

NAİVE Bayes

Naive Bayes öğrenme teknigi temeli Bayes teoremine dayanır. Bayes Teoremi bir sonucun sebebinin bulurken sonucun hangi olasılıkla hangi sebepten kaynaklandığını bulur.

Koşullu Olasılık

$$P(A|B)$$



B olayı gerçekleştiğinde A'nın gerçekleşme olasılığı

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Probability of B occurring
given evidence A has already
occurred

Probability of A occurring

Probability of A occurring
given evidence B has already
occurred

Probability of B occurring

Örnek:

Ülkemizde yaşayan ve matematik profesörü ya da şoförlük mesleklerinden biri bilinen bir kişi var. Bu kişinin tek bildiğimiz özelliği, araştırma yapmayı seviyor olması. **Bu kişinin matematik profesörü olma olasılığı kaçtır?**



Matematik profesörlerinin ne kadar araştırma yapmayı sever? %90

Şoförlerin ne kadar araştırma yapmayı sever? %15

Ülkemizdeki matematik profesörü sayısı (yaklaşık) : 500

Ülkemizdeki şoför sayısı (yaklaşık) : 700.000

$P(A)$: Kişinin matematik profesörü olma ihtimali

$P(B)$: Kişinin araştırma yapmayı sevme ihtimali

Soru: Araştırma yapmayı seven kişinin matematik profesörü olma ihtimali

$P(A|B)=?$

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Matematik profesörlerinin araştırma yapmayı sevme ihtimali
Kişinin matematik profesörü olma ihtimali
Kişinin araştırma yapmayı sevme ihtimali

$$P(A|B) = \frac{0,9 \cdot \frac{500}{700500}}{\frac{500+0,9+700.000 \cdot 0,15}{700500}} \approx 0,0043$$

$P(A|B) \approx \%0,4$

2023-06-02 13:09:31

VERİ SETİ

EĞİTİM	YAŞ	CİNSİYET	KABUL
ORTA	YAŞLI	ERKEK	EVET
İLK	GENÇ	ERKEK	HAYIR
YÜKSEK	ORTA	KADIN	HAYIR
ORTA	ORTA	ERKEK	EVET
İLK	ORTA	ERKEK	EVET
YÜSKEK	YAŞLI	KADIN	EVET
İLK	GENÇ	KADIN	HAYIR
ORTA	ORTA	KADIN	EVET

Yukarıdaki veri setine

X1: EĞİTİM = YÜKSEK

X2: YAŞ = ORTA,

X3 : CİNSİYET = KADIN

verisi eklenecektir.

Bu veri için KABUL = ?

Adım 1:

KABUL'deki Evet ve Hayır sınıflarının olasılıkları hesaplanır.

C1 : KABUL = EVET ise $P(C_1) : 5/8$

C2 : KABUL = HAYIR : ise $P(C_2) : 3/8$

Adım 2 :

Nitelik (X)	Değer	Evet Sayısı	$P(X C_1)$ (Evet Olasılığı)	Hayır Sayısı	$P(X C_2)$ (Hayır Olasılığı)
EĞİTİM	YÜKSEK	1	1/5	1	1/3
YAŞ	ORTA	3	3/5	1	1/3
CİNSİYET	KADIN	2	2/5	2	2/3

Adım 3: $P(X|C_1) * P(C_1)$ ve $P(X|C_2)P(C_2)$ değerleri hesaplanır, en büyüğü sınıfı belirler.

$$P(X_1 | C_1) = P(EĞİTİM = YÜKSEK | KABUL = EVET) : 1/5$$

$$P(X_2 | C_1) = P(YAŞ = ORTA | KABUL = EVET) : 3/5$$

$$P(X_3 | C_1) = P(CİNSİYET = KADIN | KABUL = EVET) : 2/5$$

$$P(X | C_1) = P(X | KABUL = EVET) : 1 / 5 \quad 3/5 \quad 2/5 = 6 / 125$$

$$P(X | C_1) = 6 / 125$$

$$P(X_1 | C_2) = P(EĞİTİM = YÜKSEK | KABUL = HAYIR) : 1/3$$

$$P(X_2 | C_2) = P(YAŞ = ORTA | KABUL = HAYIR) : 1/3$$

$$P(X_3 | C_2) = P(CİNSİYET = KADIN | KABUL = HAYIR) : 2/3$$

$$P(X | C_2) = P(X | KABUL = HAYIR) : 1/3 \quad 1/3 \quad 1/3 = 2/27$$

$$P(X | C_2) = 2/27 * 3/8 = 0.028$$

$P(X | C_1) P(C_1) = 6 / 125 \quad 5 / 8 = 0.03$ ihtimalle evet koşulu

$P(X | C_2) P(C_2) = 2/27 \quad 3/8 = 0.028$ ihtimalle hayır koşulu

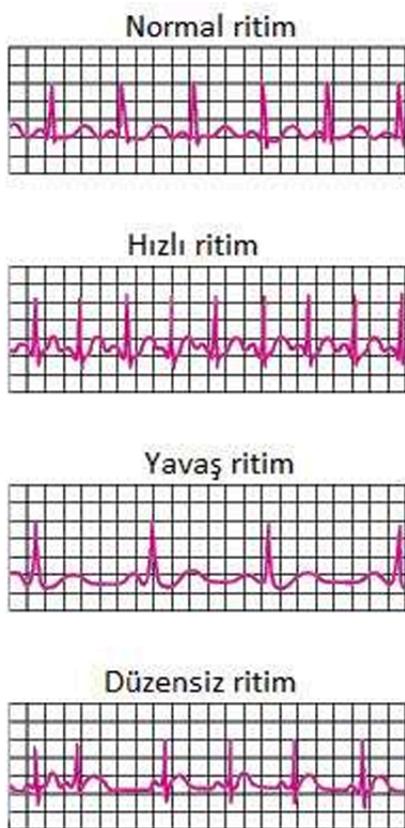
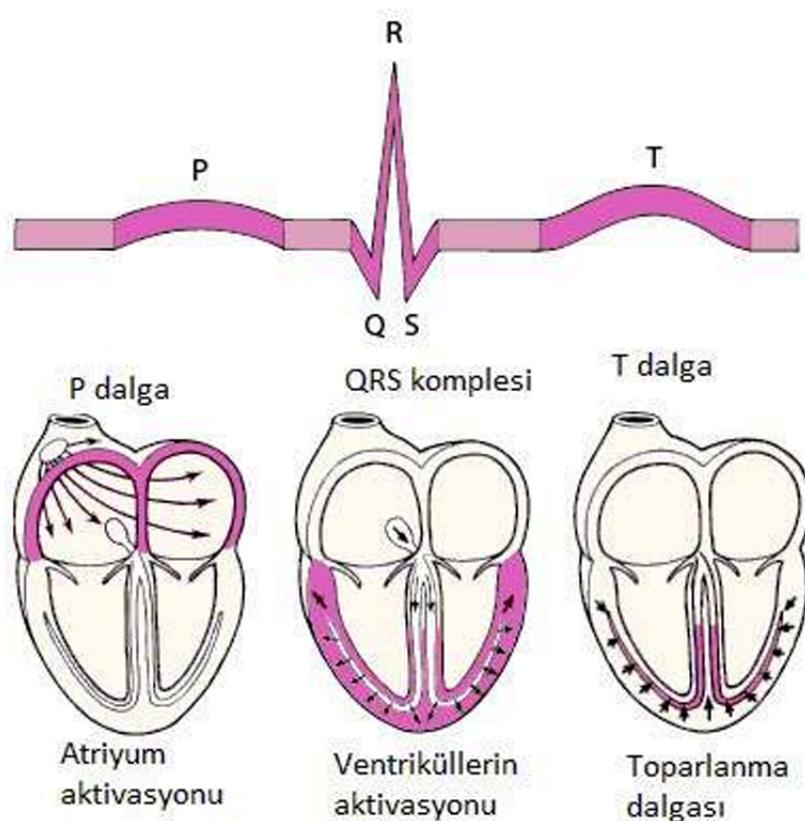
KABUL = EVET

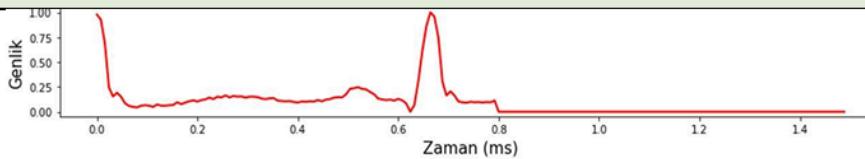
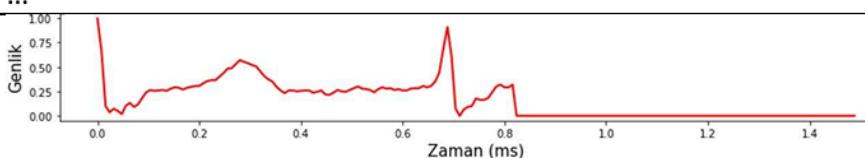
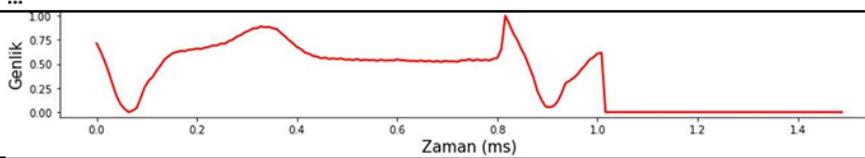
Sonuç : $P(X | C_1) P(C_1) > P(X | C_2) P(C_2)$ olduğu için KABUL = EVET olacaktır.

KOD uygulama

- **N (0):** Normal Atım
- **S (1):** Supraventriküler Ektopik Atım
- **V (2):** Ventriküler Ektopik Atım
- **F (3):** Füzyon Atımı
- **Q (4):** Bilinmeyen Atım (Sınıflandırılamayan Atım)

data = [ECG Heartbeat Categorization Dataset | Kaggle](#)



Veri Sayısı	GİRİŞ PARAMETRESİ		ÇIKIŞ PARAMETRESİ
	EKG Sinyali		
1			Normal Sinüs Ritmi
...
...
18200			Supraventriküler erken atım
...
...
21892			Sınıflandırılamayan atım
...
...
...
109446			Sınıflandırılmayan ritim

TEST VERİ SETİ

EĞİTİM VERİ SETİ

```

import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

from sklearn.naive_bayes import GaussianNB # CategoricalNB yerine GaussianNB önerilir
from sklearn.metrics import confusion_matrix, accuracy_score

# Dosyaları yükle
train = pd.read_csv("mitbih_train.csv", header=None)
test = pd.read_csv("mitbih_test.csv", header=None)

# Veriyi ayır

X_train = train.iloc[:, :187].values
y_train = train.iloc[:, 187].values
X_test = test.iloc[:, :187].values

```

```

y_test = test.iloc[:, 187].values

# Modeli eğit
gnb = GaussianNB()
gnb.fit(X_train, y_train)
y_pred = gnb.predict(X_test)

# Hata Matrisini Oluştur
cm = confusion_matrix(y_test, y_pred)
classes = ['N', 'S', 'V', 'F', 'Q'] # Sınıf isimleri (MIT-BIH için standart)
cm_df = pd.DataFrame(cm, index=classes, columns=classes)

# Çizim
plt.figure(figsize=(10,6))
sns.heatmap(cm_df, annot=True, fmt="d", cmap="YlGnBu")
plt.title("Confusion Matrix - GaussianNB")
plt.ylabel("Gerçek Sınıflar")
plt.xlabel("Tahmin Edilen Sınıflar")
plt.show()

print("Accuracy:", accuracy_score(y_test, y_pred))

```

```

import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import ComplementNB
from sklearn.metrics import confusion_matrix, accuracy_score
import seaborn as sns
import matplotlib.pyplot as plt

# 1. Veri Yükleme
veri = pd.read_csv("hayvanatbahcesi.csv", encoding='unicode_escape')

# 2. Tüm Veri Seti Üzerinde One-Hot Encoding Uygulama
veri_encoded = pd.get_dummies(veri.drop(["sinifi"], axis=1))

# 3. Giriş ve Çıkış Matrislerini Oluşturma

```

```
girisler = np.array(veri_encoded)
cikis = np.array(veri["sinifi"])

# 4. Veriyi Bölme
X_train, X_test, y_train, y_test = train_test_split(
    girisler, cikis, test_size=0.35, random_state=109
)

# 5. Modeli Eğitme ve Tahmin
gnb = ComplementNB()

gnb.fit(X_train, y_train)

y_pred = gnb.predict(X_test)

y_only = gnb.predict(X_test[0:1, :])

print(" test verisi " ,X_test[0:1, :])

print(" gerçek değer:", y_test[0:1])

print("Tek bir örnek için tahmin:", y_only)

sinif_isimleri = {

    1: 'Memeli',
    2: 'Kuş',
    3: 'Sürünge',
    4: 'Balık',
    5: 'Amfibi',
    6: 'Böcek',
    7: 'Omurgasız'
```

```
}

cm = confusion_matrix(y_test, y_pred)

# Gerçekte var olan sınıfların ID'lerini alıyoruz

sinif_idleri = np.unique(y_test)

# ID'leri kullanarak isim listesini oluşturuyoruz

classes = [sinif_isimleri.get(int(id), f'Sınıf {id}') for id in sinif_idleri]

cm_df = pd.DataFrame(cm, columns=classes, index=classes)

plt.figure(figsize=(10,6))

sns.heatmap(cm_df, annot=True, fmt="d", cmap="YlGnBu")

plt.title("Karışıklık Matrisi (ComplementNB) – Sınıf İsimleri ile")

plt.ylabel("Gerçek Sınıflar")

plt.xlabel("Tahmin Edilen Sınıflar")

plt.show()

# Metrikler

print("Accuracy:", accuracy_score(y_test, y_pred))

print("\nSınıflandırma Raporu:\n")
```