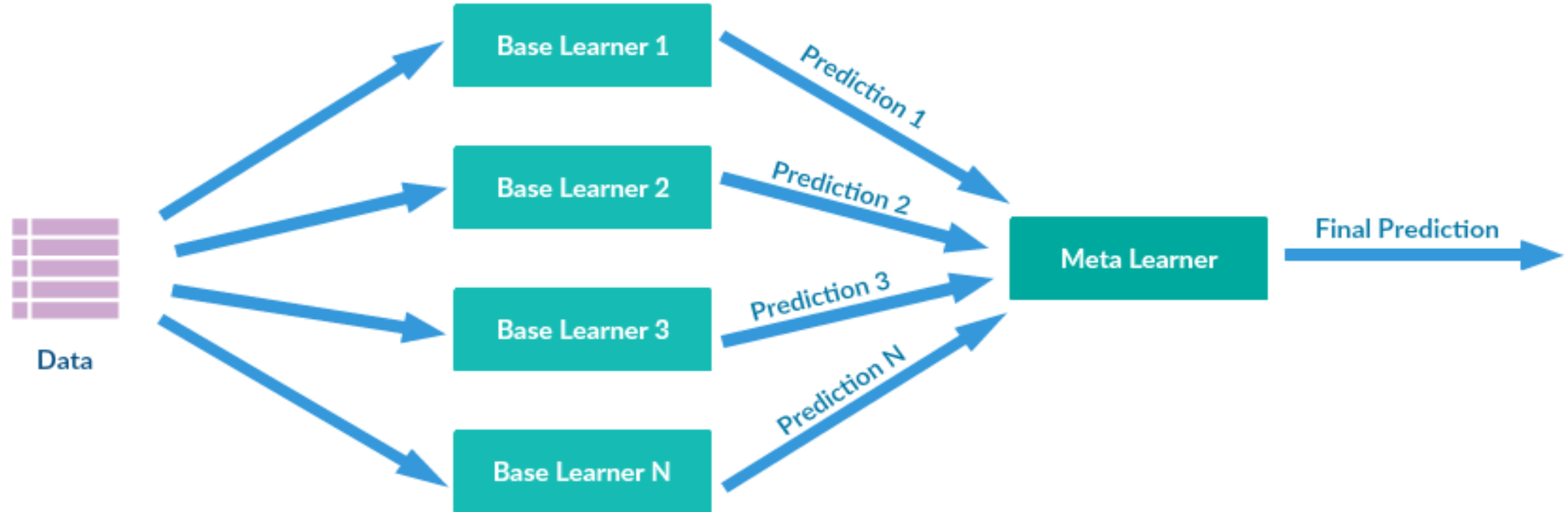


Present by

Mustafa Erdogan/02200201029

# MAKINE OGRENIMINDE BAGGING AGACLARI

1. Ensemble Learning
2. Bagging Nedir?
3. Bagging Uygulamaları
4. Bagging Vs Boosting
5. Bagging Nasıl Çalışır?
6. Bagging Yararları Ve Zorlukları



# ENSEMBLE LEARNING

- Bagging: *Bootstrap Aggregating*
- Boosting: *Ardışık ansambl öğrenme*
- Stacking: *Bir meta-model kullanarak birden fazla model tahminini birleştirme*



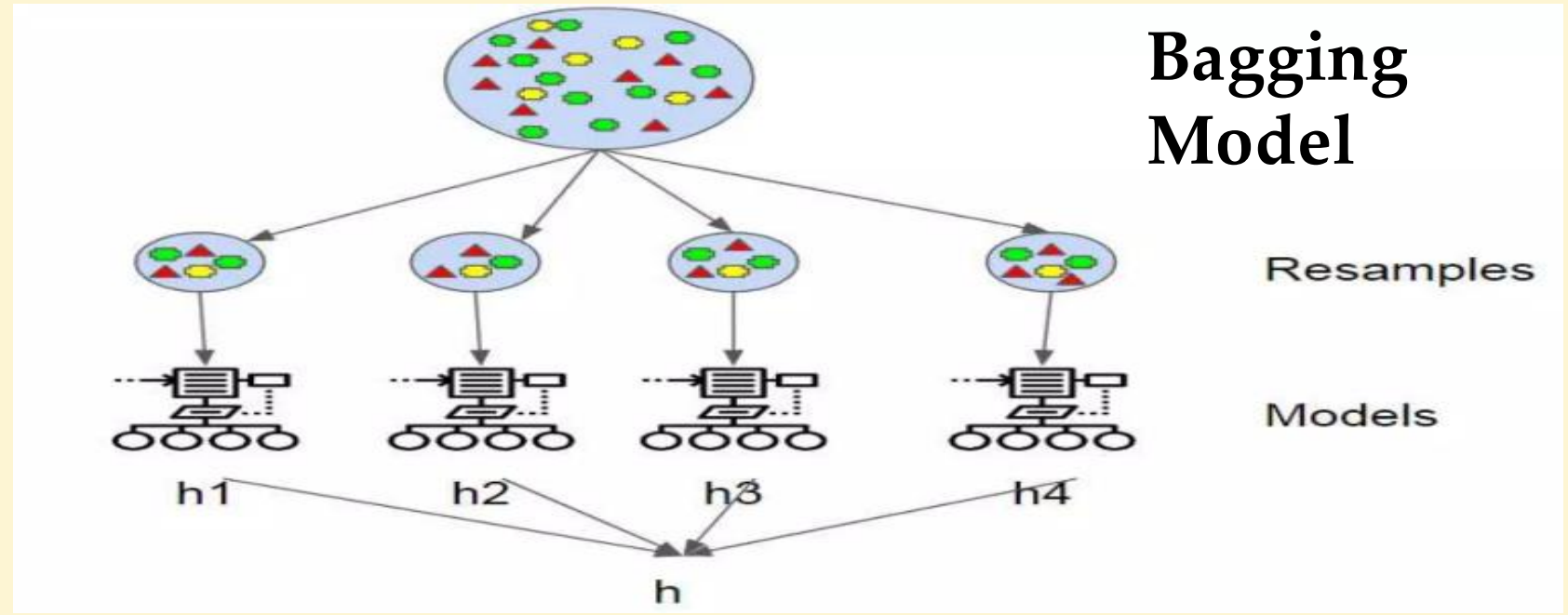
- ❖ İstatistik ve makine öğreniminde ensemble metodlar, herhangi bir kurucu öğrenme algoritmasının tek başına elde edebileceğinden daha iyi tahmin performansı elde etmek için çoklu öğrenme algoritmaları kullanır.
- ❖ Ensemble Learning ,belirli bir hesaplamalı zeka problemini çözmek için sınıflandırıcılar veya uzmanlar gibi birden fazla modelin stratejik olarak oluşturulduğu ve birleştirildiği süreçtir.
- ❖ ensemble learning, fine-grained sınıflandırma görevlerinde daha iyi ayırım yapabilen ve modelin daha güvenilir sonuçlar üretebilmesine yardımcı olan bir yöntem olarak öne çıkar

# BAGGING NEDİR?

5

Bagging olarak adlandırılan önyükleme toplama, istatistiksel Sınıflandırma ve regresyonda kullanılan makine öğrenimi algoritmalarının kararlılığını ve doğruluğunu geliştirmek için tasarlanmış bir makine öğrenimi topluluğu meta-algoritmasıdır. Aynı zamanda varyansı azaltır ve aşırı uyumun önlenmesine yardımcı olur.

Tanım: Bagging (Bootstrap Aggregating), makine öğrenimi algoritmalarının doğruluğunu ve kararlılığını artırma yöntemidir.



## Kullanım Alanları:

- Sınıflandırma Görevleri: Spam tespiti, görüntü tanıma.
- Regresyon Görevleri: Ev fiyatlarını tahmin etme, borsa eğilimlerini tahmin etme.



- ❑ Stabilité sağlar.
- ❑ Karar ağaçlarında kullanılır.
- ❑ İstatistiksel sınıflandırma ve regresyonda kullanılan makine öğrenimi algoritmalarının doğruluğunu artırır.
- ❑ Ağa izinsiz giriş tespit sistemlerinin performansını artırır.

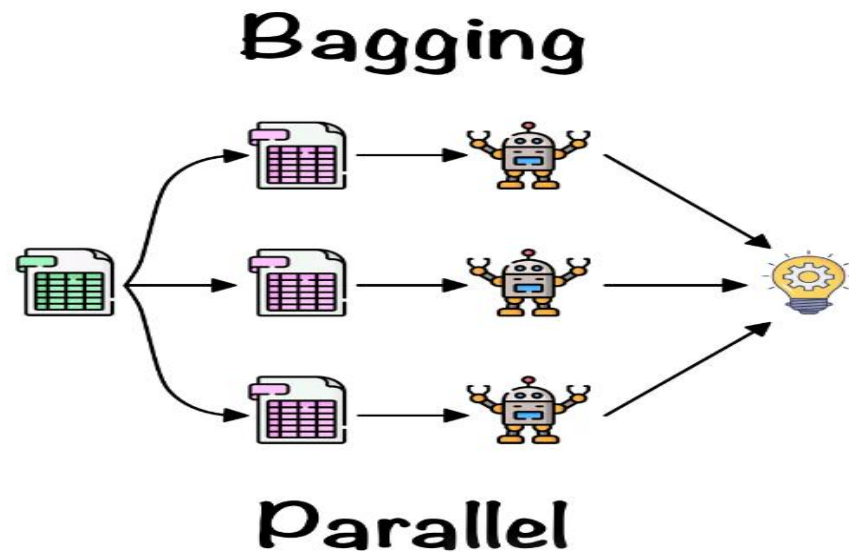
## BAGGING UYGULAMALARI

# BAGGING VS BOOSTING

7

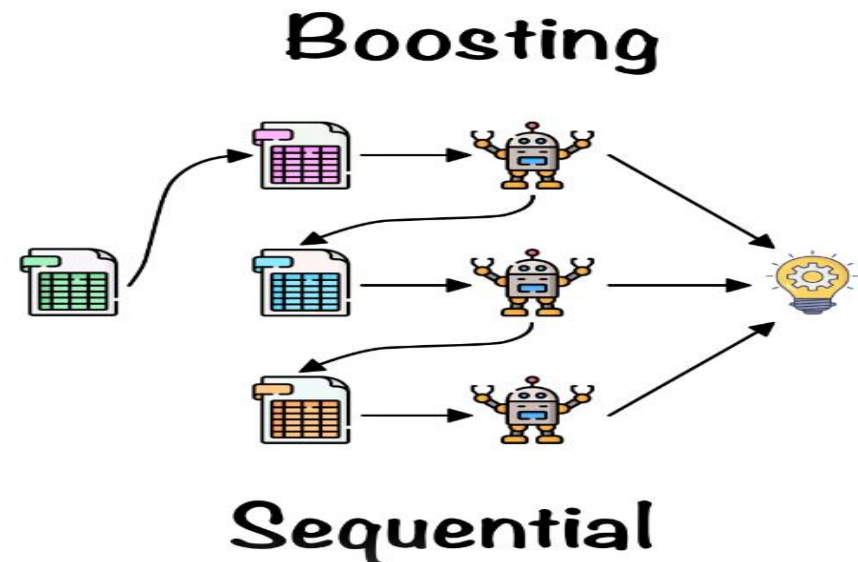
## Bagging

- Varyansı azaltır.
- Modelleri paralel olarak eğitir.
- Basit ve aşırı öğrenmeye daha az eğilimli.



## Boosting

- Yanlılığı azaltır.
- Modelleri ardışık olarak eğitir.
- Daha karmaşık, ancak daha yüksek doğruluk elde edebilir.



# BAGGING NASIL ÇALIŞIR?

8

## Bootstrapping

- Tanım: *Yeniden örnekleme ile rastgele örnekleme.*
- Süreç
  1. N boyutundaki bir veri setinden yeniden örnekleme ile N örnek almak.
  2. Her alt küme bazı yinelenen ve bazı eksik orijinal örneklerle sahip olur.

```
from sklearn.utils import resample

# Orijinal veri seti
X, y = load_data()

# Bootstrap örnekleme
X_resampled, y_resampled = resample(X, y, replace=True, n_samples=len(X))
```



# BAGGING NASIL ÇALIŞIR?

## Paralel Eğitim

Tanım: *Birden fazla modeli aynı anda eğitme.*

## Avantajlar:

- ✓ *Tüm modeller aynı anda eğitilir.*
- ✓ *Birbirinden bağımsızdır, etkileşim yoktur.*
- Hız: Çok çekirdekli işlemcilerden yararlanır.
- Bağımsızlık: Her model verinin farklı yönlerini öğrenir.

```
from sklearn.ensemble import BaggingClassifier
from sklearn.tree import DecisionTreeClassifier

# Bagging sınıflandırıcısını başlat
bagging = BaggingClassifier(base_estimator=DecisionTreeClassifier(), n_estimators=10,

# Modeli eğit
bagging.fit(X_train, y_train)
```

# BAGGING NASIL ÇALIŞIR?

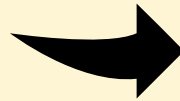
## Birleştirme(Aggregation)

Tanım: *Birden fazla modelin çıktılarının birleştirilmesi.*

### Yöntemler:

- Sınıflandırma→ Çoğunluk oyu.
- Regresyon→ Ortalama.

```
# Bagging modeli ile tahmin et  
y_pred = bagging.predict(X_test)
```



Regresyon için  
bagging.predict\_proba(X\_test).mean(axis=1) ile  
olasılıkların ortalamasını kullan

## Bagging Yararları

- Uygulama Kolaylığı:
  - Basit kavram ve mevcut modellerle kolayca uygulanabilir.
  - Scikit-learn ve diğer kütüphaneler yerleşik destek sağlar.
- Varyansın Azaltılması:
  - Birden fazla modelin birleştirilmesi aşırı öğrenmeyi azaltır.
  - Daha istikrarlı ve doğru tahminler sağlar.

## Bagging Zorlukları

- Yorumlanabilirlik Kaybı:
- Birleştirilen modeller tek modellere göre daha zor yorumlanır.
- Tek tek karar ağaçları yorumlanabilir, ancak ansambl yorumu zorlaştırır.
- Hesaplama Açısından Pahalı Olması:
- Birden fazla modelin eğitimi daha fazla hesaplama kaynağı gerektirir.
- Büyük veri setleri için birleştirme zaman alıcı olabilir.

## Train models with accuracy

Train models and print their accuracy

```
#Train different models and print their accuracy
results = model_selection.cross_val_score(model, X_fit, y_fit, cv=kfold)
for i in range(len(results)):
    print("Model: "+str(i)+" Accuracy is: "+str(results[i]))
```

```
Model: 0 Accuracy is: 1.0
Model: 1 Accuracy is: 1.0
Model: 2 Accuracy is: 1.0
Model: 3 Accuracy is: 0.9090909090909091
Model: 4 Accuracy is: 1.0
Model: 5 Accuracy is: 1.0
Model: 6 Accuracy is: 0.9
Model: 7 Accuracy is: 1.0
Model: 8 Accuracy is: 1.0
Model: 9 Accuracy is: 0.7
```

- **Özet:**
- Bagging, model doğruluğunu ve kararlılığını artıran güçlü bir ansambl yöntemidir.
- Bootstrapping, paralel eğitim ve birleştirme içerir.
- Varyansın azaltılması ve uygulama kolaylığı gibi önemli yararları olsa da, yorumlanabilirlik kaybı ve hesaplama maliyeti gibi zorlukları da vardır.
- **Anahtar Nokta:** Özellikle Random Forests şeklinde Bagging, basitlik ve performans arasında denge sağlayan yaygın bir tekniktir.

Bu sunum, bagged trees konusunu kapsamlı bir şekilde ele alarak, temel kavramlar, süreçler, yararlar ve zorlukları kapsar. Bagging'in nasıl çalıştığını ve Python'da scikit-learn kullanarak nasıl uygulanabileceğini kod örnekleriyle açıklar.

THANK  
YOU