Makine Öğrenmesi Nedir?

Karmaşık Sistemler ve Veri Bilimi Çalıştayı
@Boğaziçi Üniversitesi
04.05.2019

Ben Kimim?

Metin USLU

İş

- Veri Bilimci @ LC Waikiki - 02.2019
- Büyük Veri Analisti | Bilgi Teknolojileri ve İletişim Kurumu (BTK) - 01.2017 - 02.2019
- Yönetim Bilişim Sistemleri Uzm Yrd. | Kuveyt Türk Katılım Bankası - 10.2015 - 01.2017
- . . .

Eğitim

İstanbul Üniversitesi | Bilgisayar Programlama
Hacettepe Üniversitesi | İstatistik
Konya NEU | Bilgisayar Mühendisliği (Öğrenci)

İletişim 

@metinuslu - uslumetin@gmail.com

Sunum Planı

1. LC Waikiki’de Neler Yapıyoruz?
2. Büyük Veri ve Yapay Zeka Laboratuvarı: BTK BAB
3. Makine Öğrenmesi Nedir?

LC Waikiki | Mağaza İçi Analitik (In Store Analytics)

Kaynak: https://www.necdisplay.com/images/whitepapers/Facial_Recognition_Retail_Benefits.jpg

Mağaza Müşterilerinin

Üzerinde Çalıştıklarımız

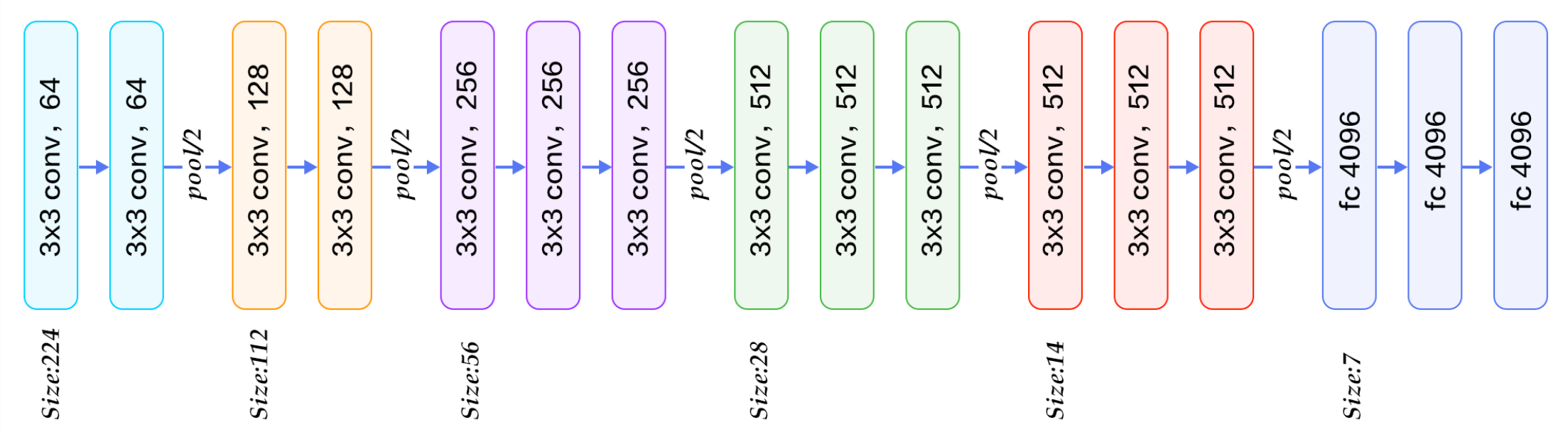
- **Cinsiyet** - VGG-16 - Acc.: %96
- **Yaş** (Görünen) - VGG-16 - MAE: ±4.65
- **Boy Uzunluğu** - Object Detection ve Pose Estimation

Neler Yapmak İstiyoruz?

- **Emotion**
- **Persona** (Profil, Eğilim)

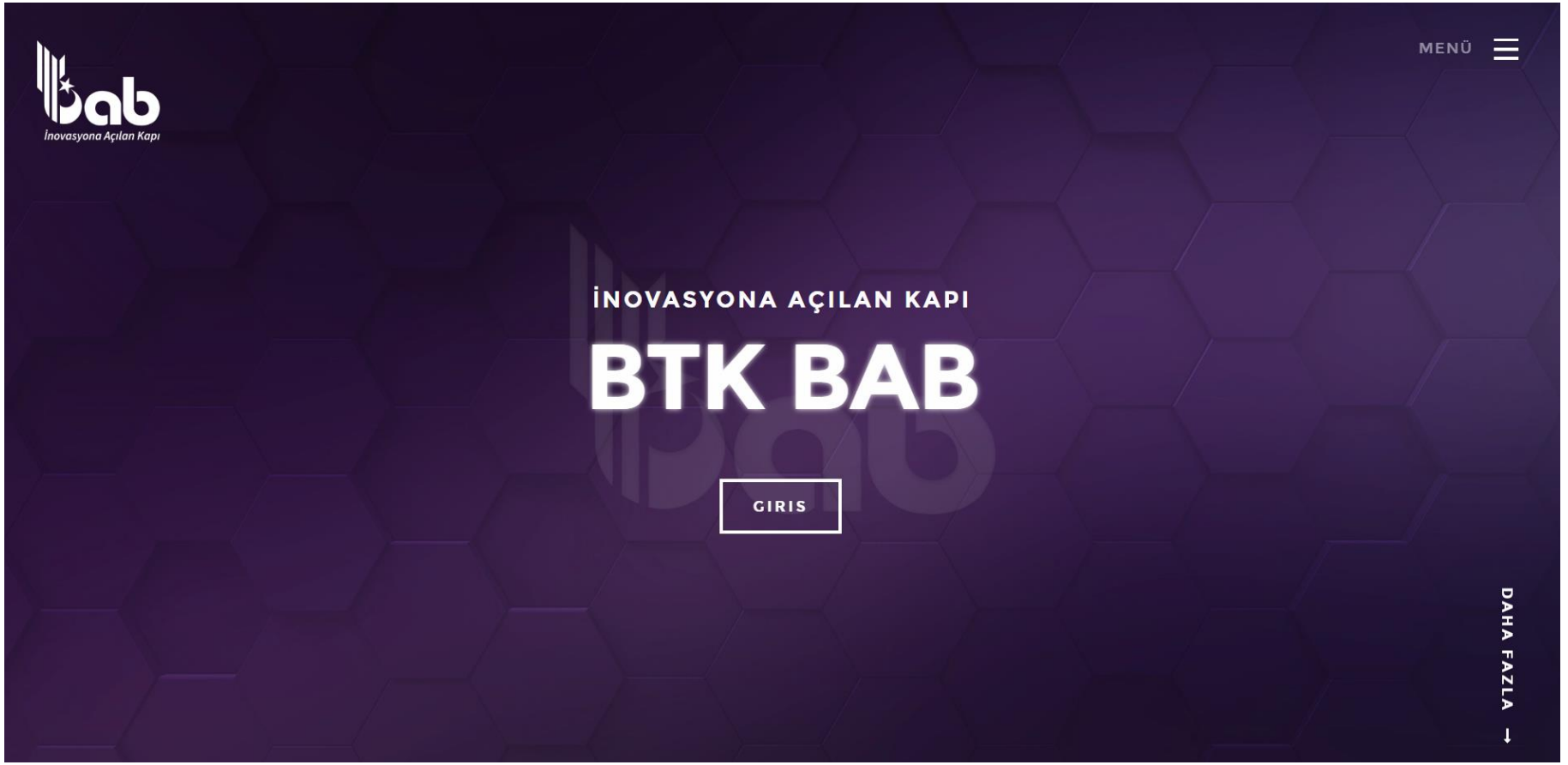
LC Waikiki | Mağaza İçi Analitik (In Store Analytics)

VGG-16



Kaynak: https://www.quora.com/What-is-the-VGG-neural-network

BTK BAB | Büyük Veri ve Yapay Zeka Lab.



Kaynak: bab.btk.gov.tr

BTK BAB | Büyük Veri ve Yapay Zeka Lab.

Yazılım Bileşenleri:

- Vertica
- GreenPlum
- Hadoop Ekosistemi
- Apache Spark
- PostgreSQL
- ElasticSearch
- Cassandra
- Mongo
- Neo4J
- Python
- Weka
- R & R Studio Server
- Knime
- . . .

Donanım Bileşenleri:

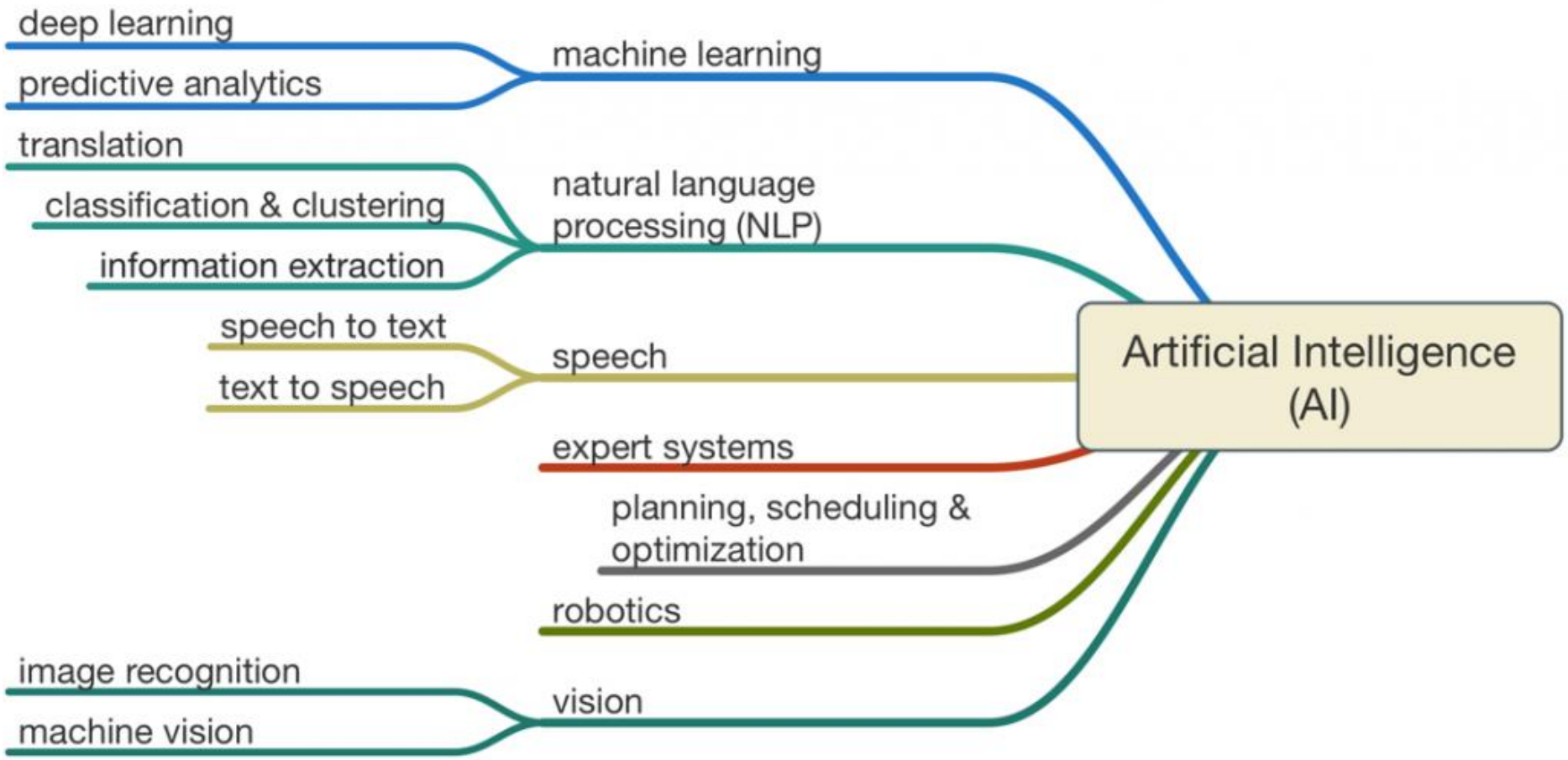
- 2.560 GB Ram
- 250 TB SAS Disk
- 4.800 GB SSD Disk
- 400 Core Intel Xeon Gold 6148 2.4 Ghz İşlemci
- 10 Gigabit Network

OS & Diğer Araçlar:

- Sunucu: Centos 7
- İstemci: Ubuntu Desktop 16.04
- Public Verisetleri
- Postman, DBeaver Robo3T, v.d.

Daha fazlası için **bab.btk.gov.tr**

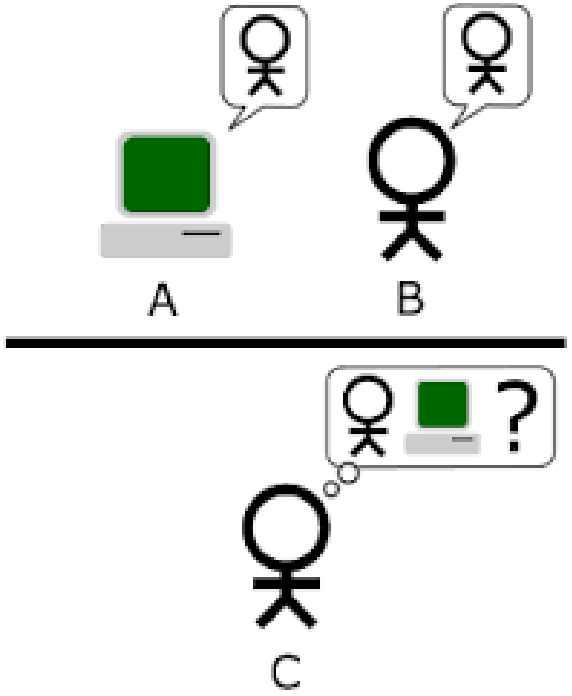
Yapay Zeka



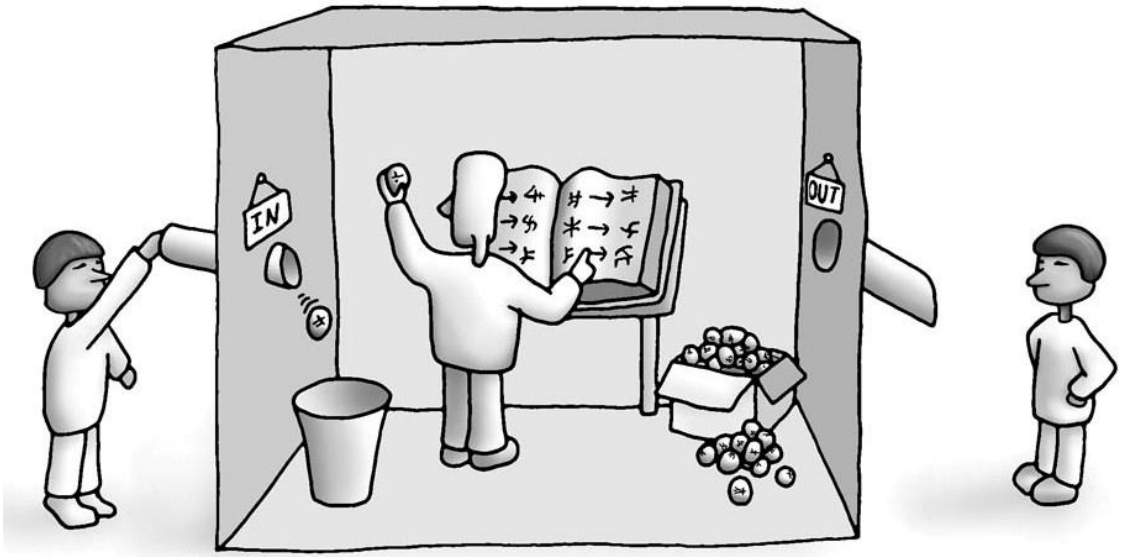
Kaynak: http://sanket.khandare.com/wp-content/uploads/2017/07/AI-subsets-1024x502.png

Yapay Zeka Testleri

1- Turing Testi



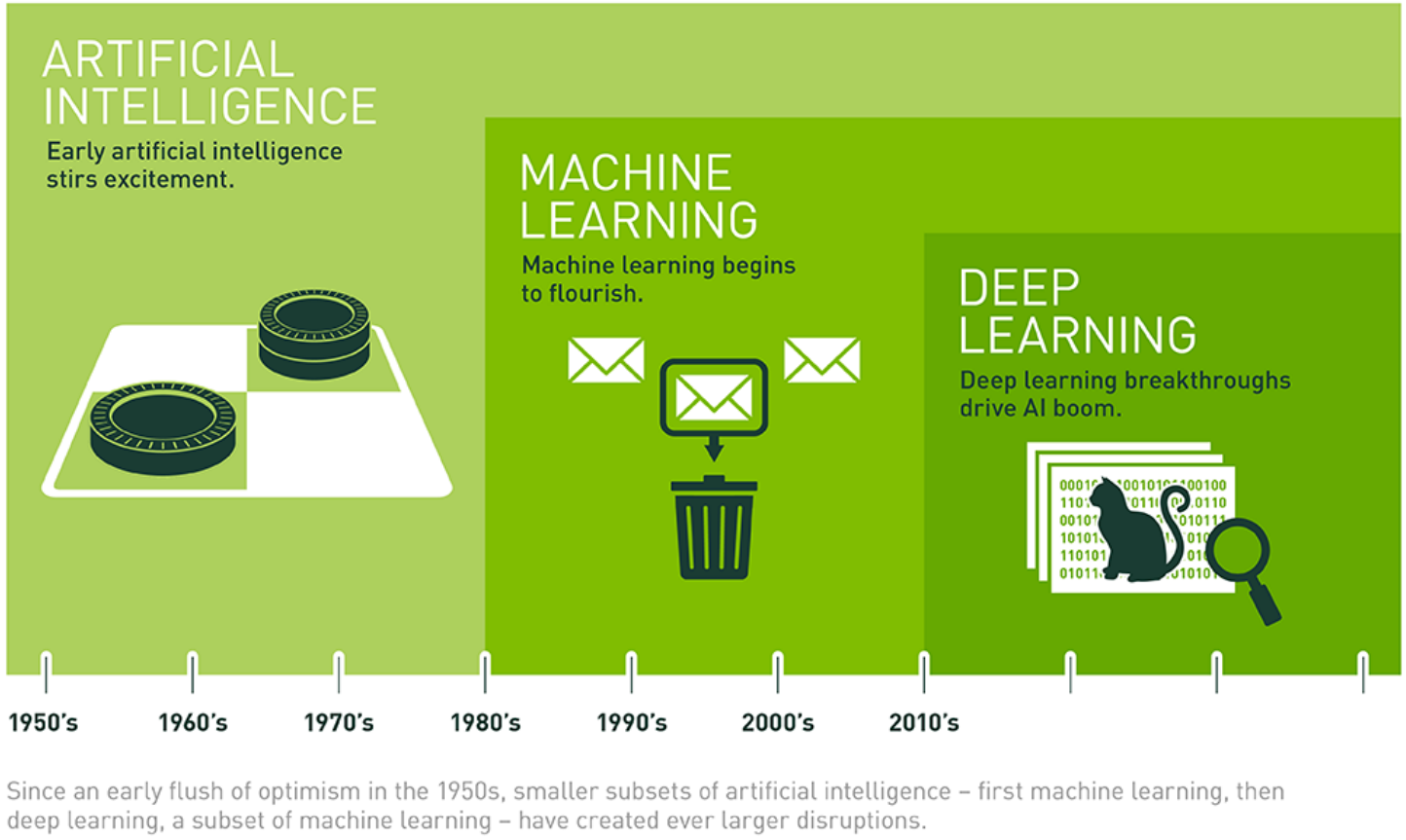
2- Çin Odası Deneyi



Kaynak 1: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly91cGxvYWQud2lraW1lZGlhLm9yZy93aWtpcGVkaWEvY29tbW9ucy9lL2U0L1R1cmluZ19UZXN0X3ZlcnNpb25fMy5wbmc

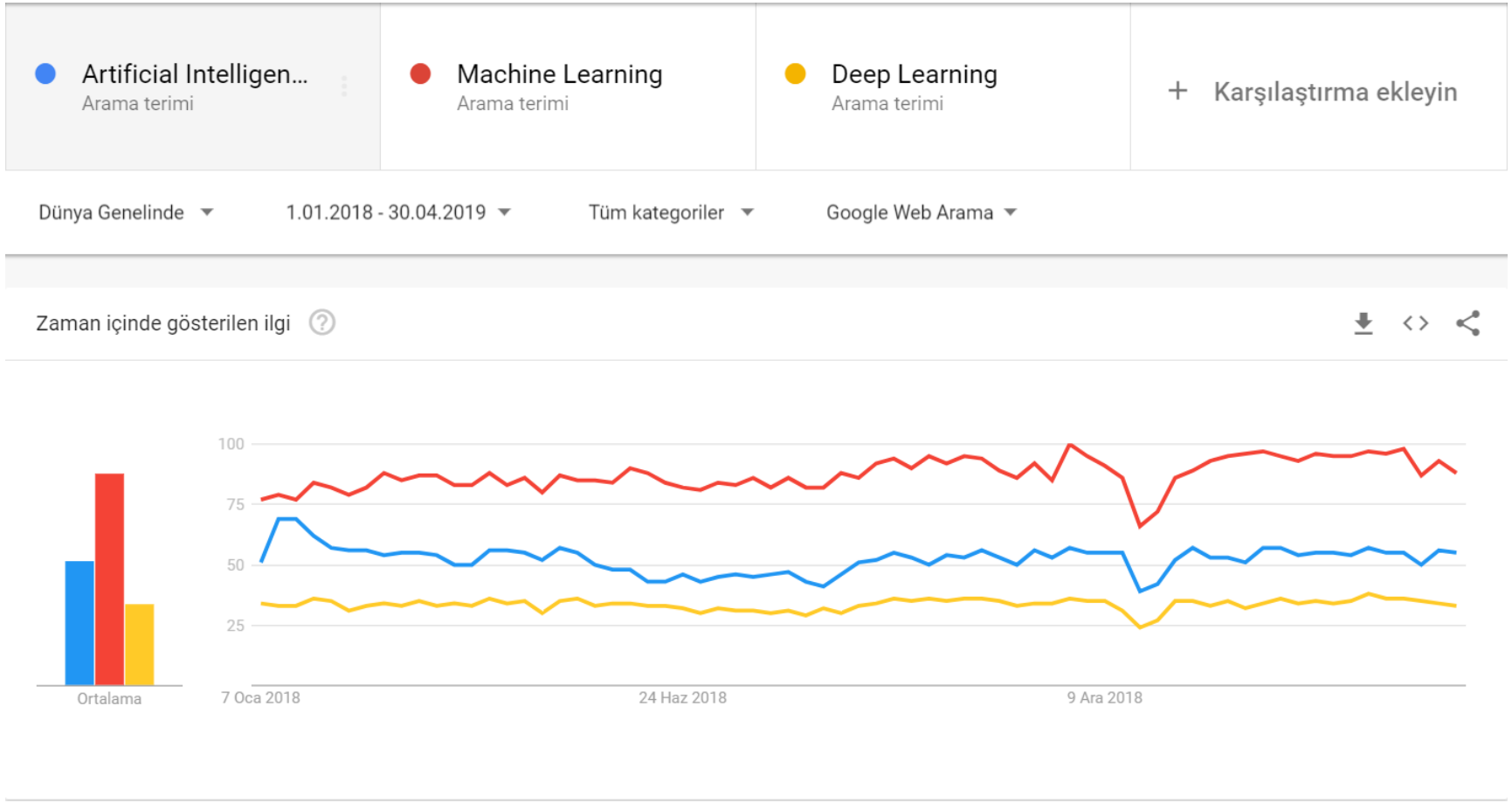
Kaynak 2: https://cdnelektrikport.4flyy.com//Content//201408/The-Chinese-Room.jpg

Yapay Zeka & Makine Öğrenmesi & Derin Öğrenme



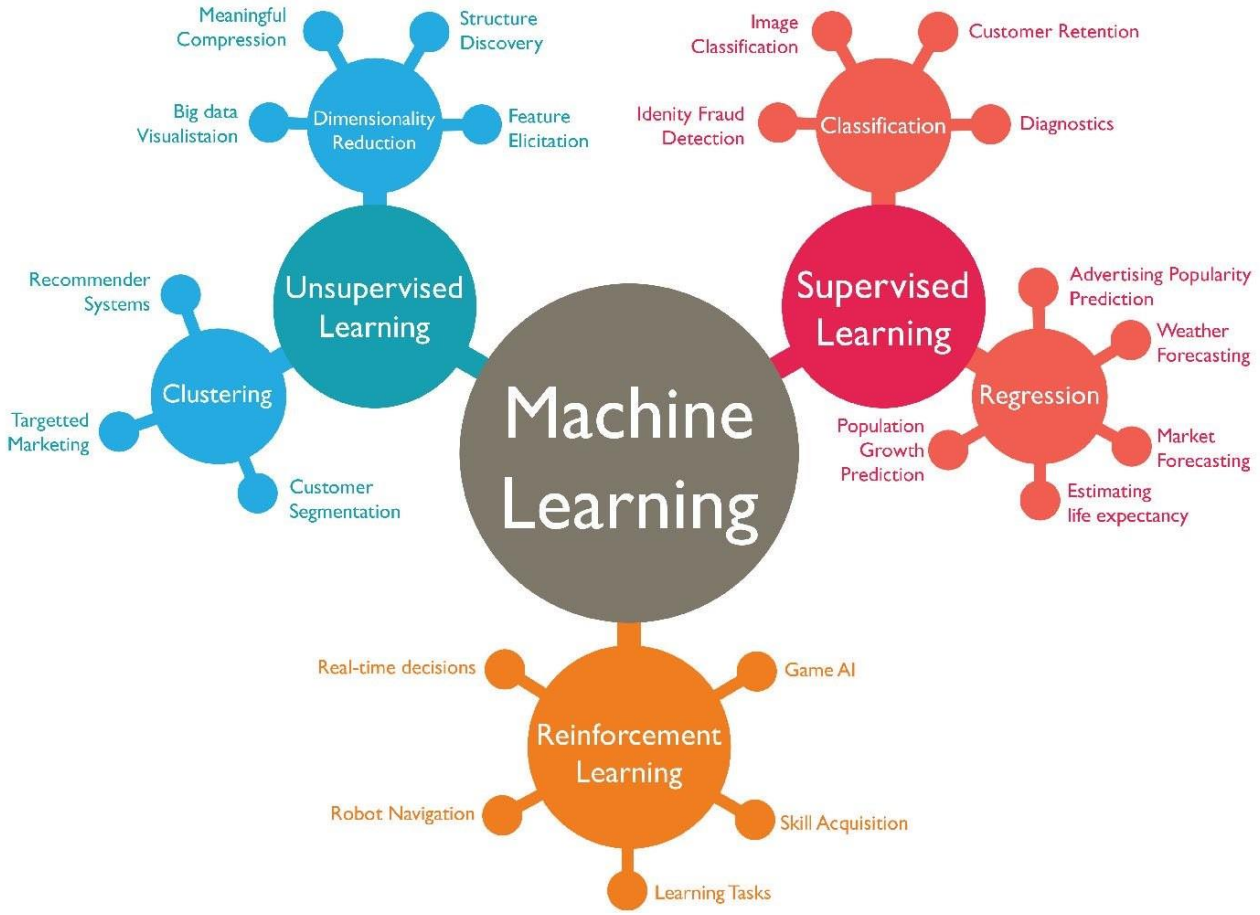
Kaynak: https://blogs.nvidia.com/wp-content/uploads/2016/07/Deep_Learning_Icons_R5_PNG.jpg.png

Yapay Zeka & Makine Öğrenmesi & Derin Öğrenme



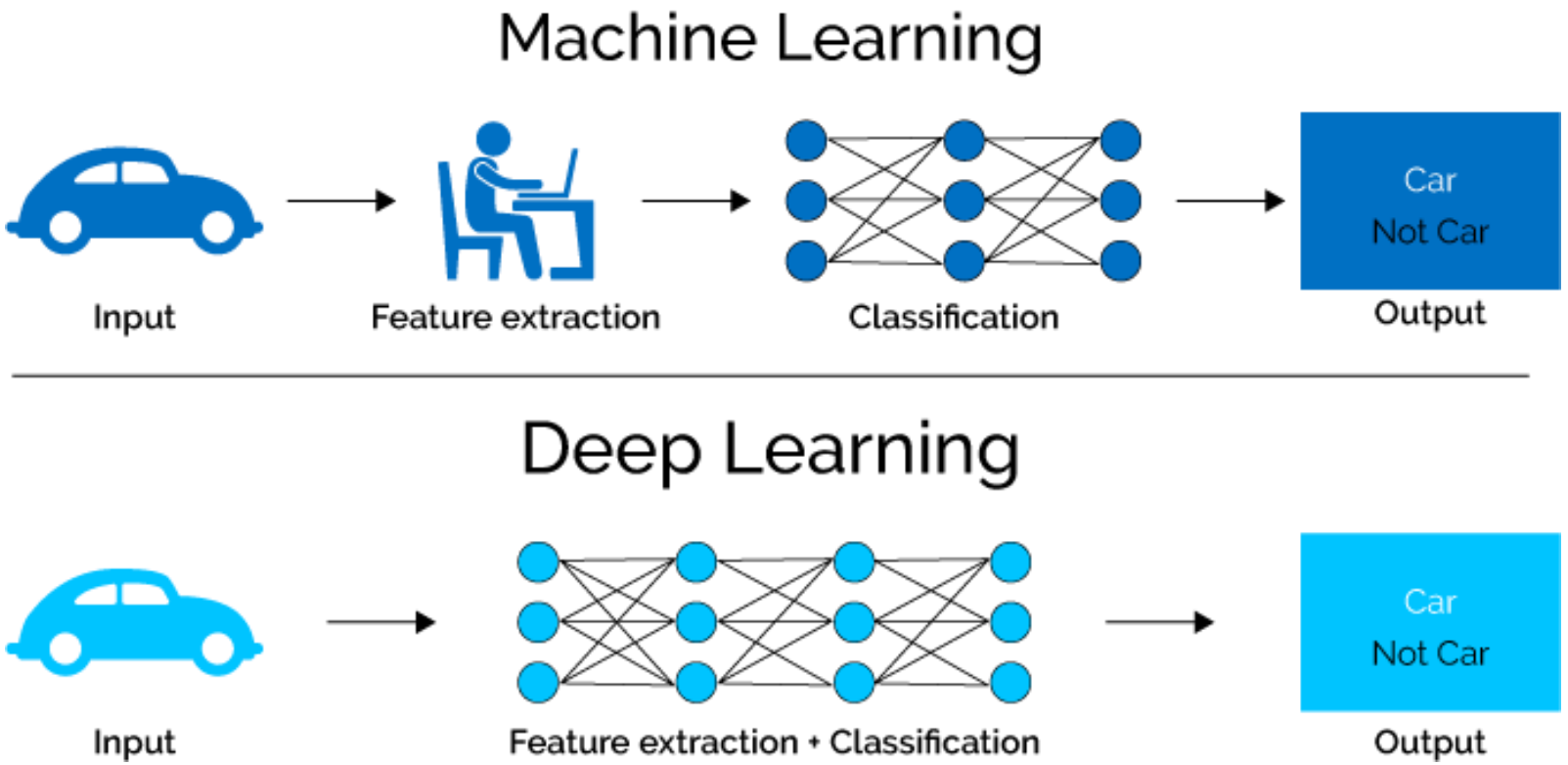
Kaynak: https://trends.google.com.tr/trends/explore?date=2018-01-01%202019-04-30&q=Artificial%20Intelligence,Machine%20Learning,Deep%20Learning

Makine Öğrenmesi



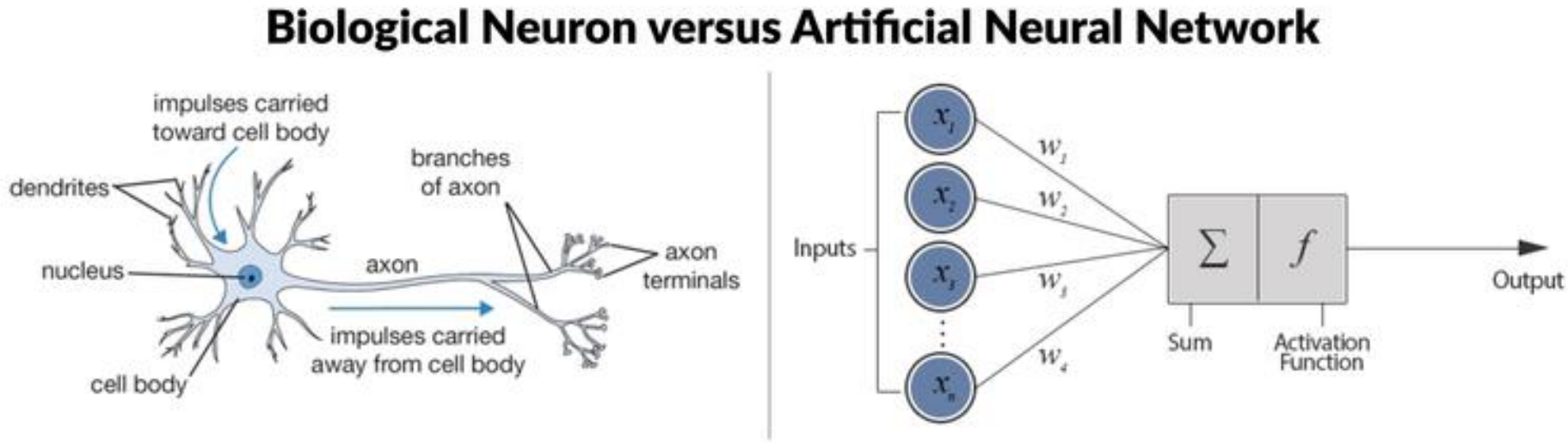
Kaynak: https://blogs.nvidia.com/wp-content/uploads/2016/07/Deep_Learning_Icons_R5_PNG.jpg.png

Makine Öğrenmesi vs Derin Öğrenme



Kaynak: https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063

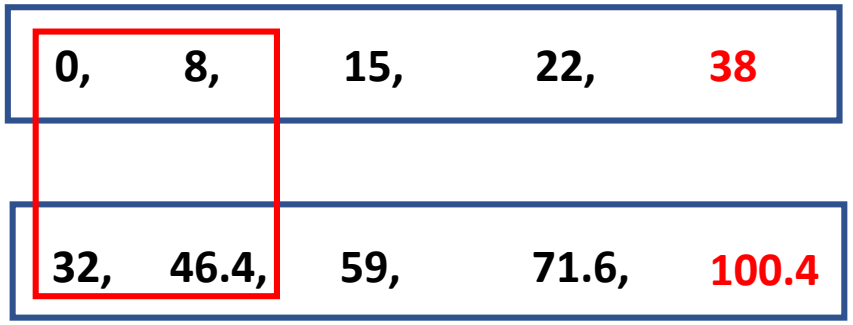
Biyolojik Sinir Sistemi vs Yapay Sinir Ağları



Kaynak: https://www.datacamp.com/community/tutorials/deep-learning-python

Peki Makineler Nasıl Öğrenir?

Bir problem ile başlayalım;



Adım 1: 0 * k + A = 32 ➔ A=32

Adım 2: 8 * k + A = 46.4 ➔ 8 * k + 32 = 46.4 ➔ k=1.8

Adım 3: 38 * 1.8 + 32 = 100.4

Peki Makineler Nasıl Öğrenir?

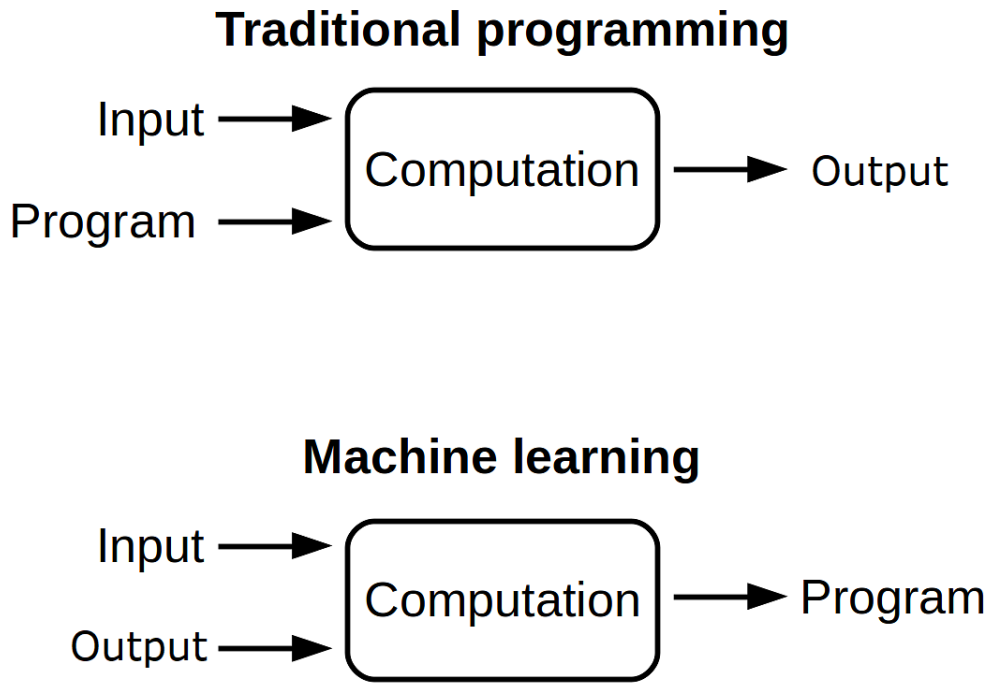
Input	0, 8, 15, 22, 38,
Output	32, 46.4, 59, 71.6, 100.4

F= C*1.8 + 32

F: Fahrenheit

C: Celsius

Geleneksel Prog. vs Makine Öğrenmesi

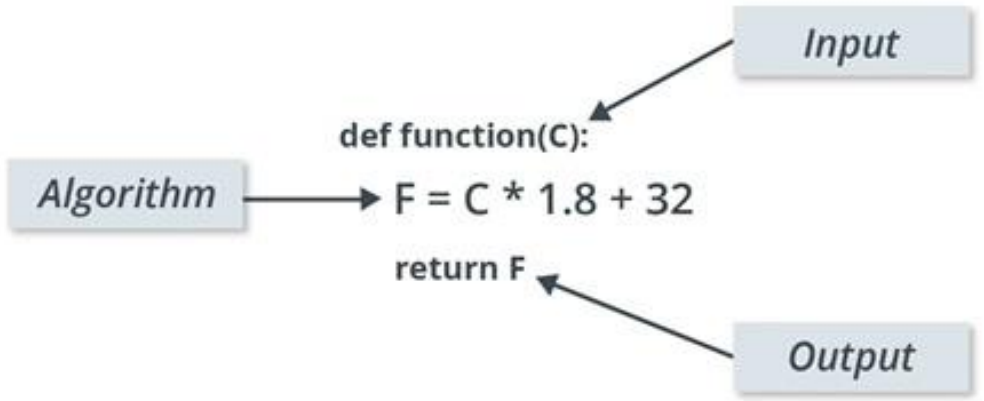


Geleneksel Programlama

F = C*1.8 + 32

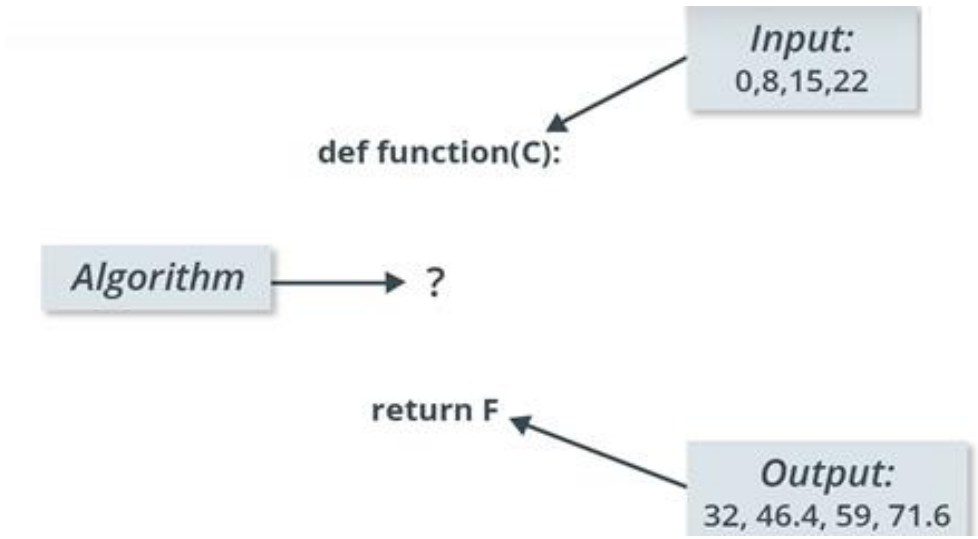
Python

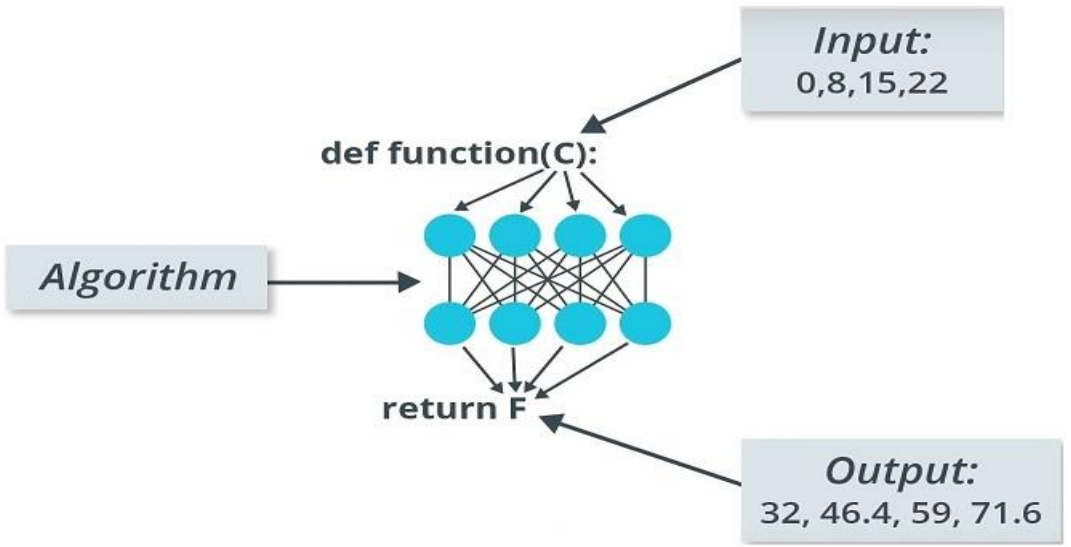
```
def Cel2Fah(C):
    F = C * 1.8 + 32
    return F

print("Celsius to Fahrenheit:", Cel2Fah(100))
```



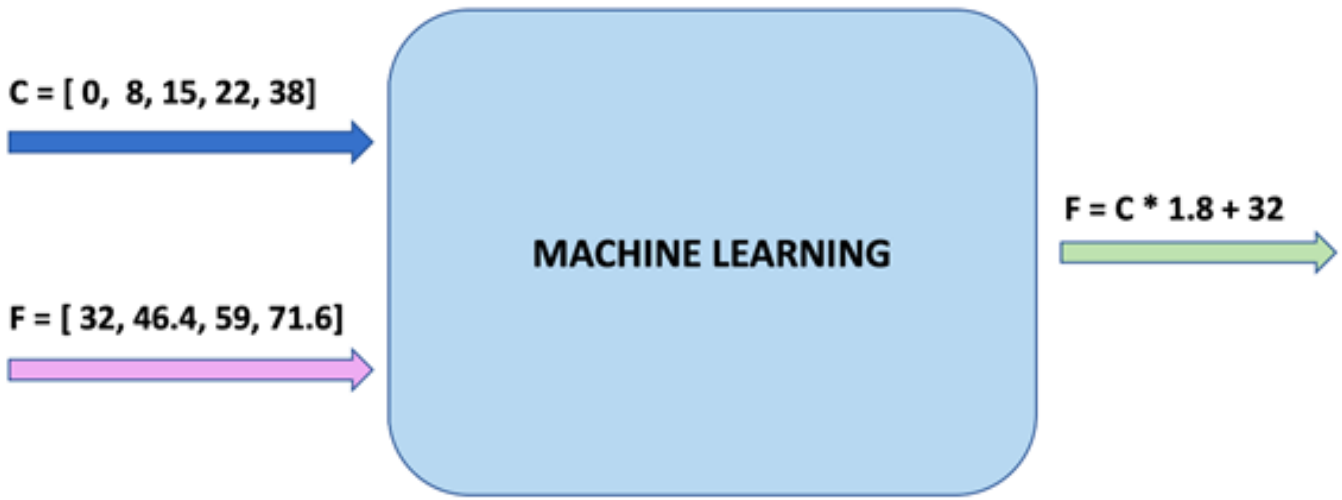
Makine Öğrenmesi

Python





Makine Öğrenmesi

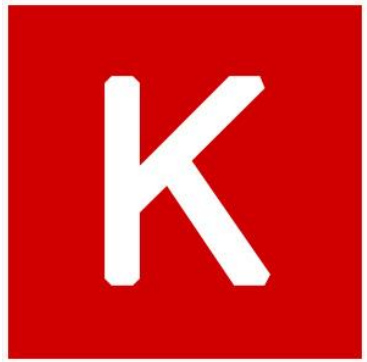


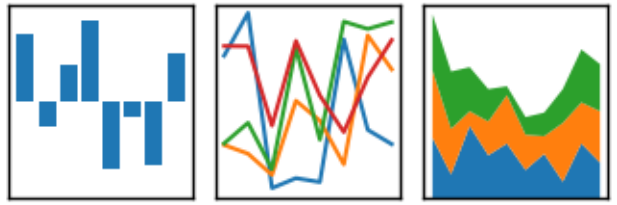
Python Uyg. (Veri Seti & Kütüphaneler)

No	Input	Output
1	-65	-85
2	-52	-61,6
3	-40	-40
4	-26	-14,8
5	-19	-2,2
. . .	. . .	. . .
. . .	. . .	. . .
. . .	. . .	. . .
46	520	968
47	533	991,4
48	546	1014,8
49	559	1038,2
50	572	1061,6

Python - Jupyter / Google Colab üzerinde, Input(Celsius) ve Output(Fahrenheit) değerlerinden oluşan 50 gözlemli bir veri seti ile **Derin Sinir Ağları** ve **Doğrusal Regresyon** kullanarak Makine Öğrenmesi uygulaması gerçekleştireceğim.

Kütüphaneler
Tensorflow
Keras
ScikitLearn
Pandas
Numpy
Matplotlib





Python Uyg. (Sinir Ağları)

```
!python -V
!conda env list
```

```
Python 3.6.8 :: Anaconda, Inc.
```

```
# conda environments:
#
base                     C:\ProgramData\Anaconda3
py36                  *  C:\ProgramData\Anaconda3\envs\py36
py36_gpu                 C:\ProgramData\Anaconda3\envs\py36_gpu
```

```
from __future__ import absolute_import, division, print_function, unicode_literals
import tensorflow as tf
tf.logging.set_verbosity(tf.logging.ERROR)

from sklearn.model_selection import train_test_split
from sklearn.utils import shuffle
import numpy as np              # Low-level numerical Python library.
import pandas as pd
import os
```

```
print("Working Directory:", os.getcwd())
print("Tensorflow Version:", tf.__version__)
print("Numpy Version:", np.__version__)
print("Pandas Version:", pd.__version__)
```

```
Working Directory: C:\Users\Metin USLU\Desktop\KaVe Sunum
Tensorflow Version: 1.13.1
Numpy Version: 1.15.4
Pandas Version: 0.24.1
```

Python Uyg. (Sinir Ağları)

```
data = pd.read_excel('dataset.xlsx', index=False, index_col=None, header=0, sheet_name='Sheet2', skiprows=2)

data = shuffle(data)
# data.head()

celsius = data['Celsius'].values
fahrenheit = data['Fahrenheit'].values

for i, c in enumerate(celsius[0:10]):
  print("{}. {} Celsius ==> {} Fahrenheit".format(i+1, c, round(fahrenheit[i],3)))
```

```
1. 221 Celsius ==> 429.8 Fahrenheit
2. 260 Celsius ==> 500.0 Fahrenheit
3. 247 Celsius ==> 476.6 Fahrenheit
4. 23 Celsius ==> 73.4 Fahrenheit
5. 572 Celsius ==> 1061.6 Fahrenheit
6. -65 Celsius ==> -85.0 Fahrenheit
7. 104 Celsius ==> 219.2 Fahrenheit
8. 338 Celsius ==> 640.4 Fahrenheit
9. 403 Celsius ==> 757.4 Fahrenheit
10. 390 Celsius ==> 734.0 Fahrenheit
```

```
l0 = tf.keras.layers.Dense(units=1, input_shape=[1])
```

```
model = tf.keras.Sequential([l0])
```

```
# model = tf.keras.Sequential([
#   tf.keras.layers.Dense(units=1, input_shape=[1])
# ])
```

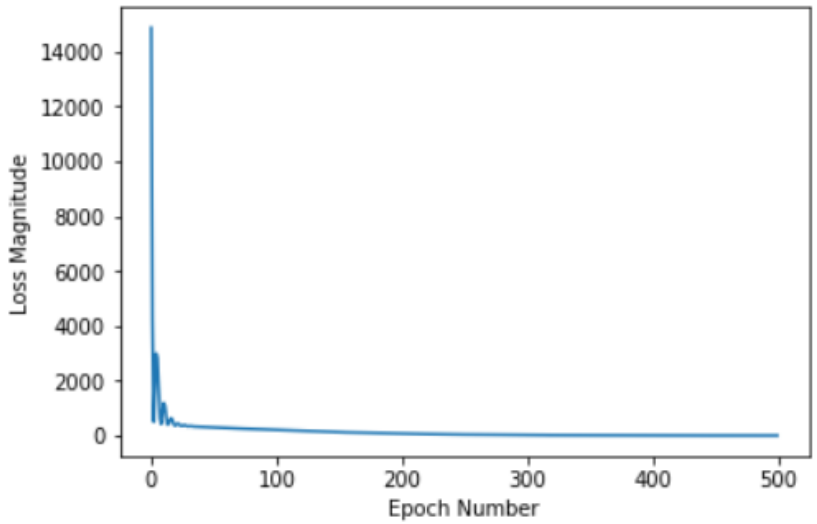
```
model.compile(loss='mean_squared_error',
              optimizer=tf.keras.optimizers.Adam(0.1))
```

```
history = model.fit(celsius, fahrenheit, epochs=500, verbose=True)
print("Finished training the model")
```

Python Uyg. (Sinir Ağları)

```
1 import matplotlib.pyplot as plt
2 plt.xlabel('Epoch Number')
3 plt.ylabel("Loss Magnitude")
4 plt.plot(history.history['loss'])
```

```
[<matplotlib.lines.Line2D at 0x1ee2b5c89e8>]
```




```
1 print("These are the layer variables: {}".format(l0.get_weights()))
```

```
These are the layer variables: [array([[1.8024685]], dtype=float32), array([31.022383], dtype=float32)]
```

```
1 # print(l0.get_weights())
2 print("Fahrenheit = Celsius*{} + {}".format(float(l0.get_weights()[0]), float(l0.get_weights()[1])))
```

```
Fahrenheit = Celsius*1.8024685382843018 + 31.022382736206055
```

Python Uyg. (Sinir Ağları)

Celsius 100 için Fahrenheit : ?

- Sinir Ağları Modeli => 31.022 + 1.802*Celcius

  31.022 + 1.802*100 = 211.269

- Cel2Fah => 100*1.8 + 32 = 212

```
1  print("Celsius to Fahrenheit:", model.predict([100]))
```

```
Celsius to Fahrenheit: [[211.26924]]
```

```
1  def Cel2Fah(C):
2      F = C * 1.8 + 32
3      return F
```

```
1  print("Celsius to Fahrenheit:", Cel2Fah(100))
```

```
Celsius to Fahrenheit: 212.0
```

Python Uyg. (Regresyon)

```
!python -V
!conda env list
```

```
Python 3.6.8 :: Anaconda, Inc.
```

```
# conda environments:
#
base                     C:\ProgramData\Anaconda3
py36                  *  C:\ProgramData\Anaconda3\envs\py36
py36_gpu                 C:\ProgramData\Anaconda3\envs\py36_gpu
```

```
from sklearn import linear_model
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.model_selection import train_test_split
import numpy as np
import pandas as pd
import xlrd
import os
import matplotlib.pyplot as plt
```

```
print("Working Directory:", os.getcwd())
print("Numpy Version:", np.__version__)
print("Pandas Version:", pd.__version__)
print("xlrd Version:", xlrd.__version__)
```

```
Working Directory: C:\Users\Metin USLU\Desktop\KaVe Sunum
Numpy Version: 1.15.4
Pandas Version: 0.24.1
xlrd Version: 1.2.0
```

Python Uyg. (Regresyon)

```
1  data = pd.read_excel('dataset.xlsx', index=False, index_col=None, header=0, sheet_name='Sheet2', skiprows=2)
```

```
1  data.head()
```

	Celsius	Fahrenheit
0	-65	-85.0
1	-52	-61.6
2	-40	-40.0
3	-26	-14.8
4	-19	-2.2

```
1  data.shape
```

```
(50, 2)
```

```
1  X = data.iloc[:, :-1]
2  y = data.iloc[:, 1]
3
4  X_Train, X_Test, y_Train, y_Test = train_test_split(X, y, test_size = .10, random_state = 0, shuffle=True)
```

Python Uyg. (Regresyon)

```
1  model = linear_model.LinearRegression()
```

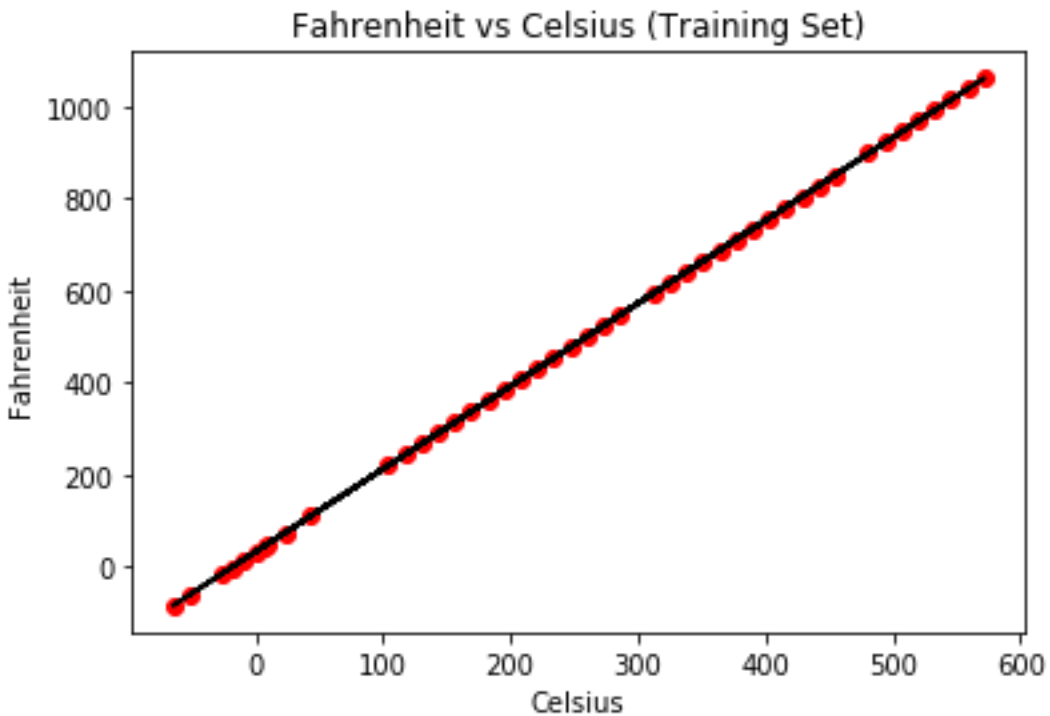
```
1  model.fit(X_Train, y_Train)
```

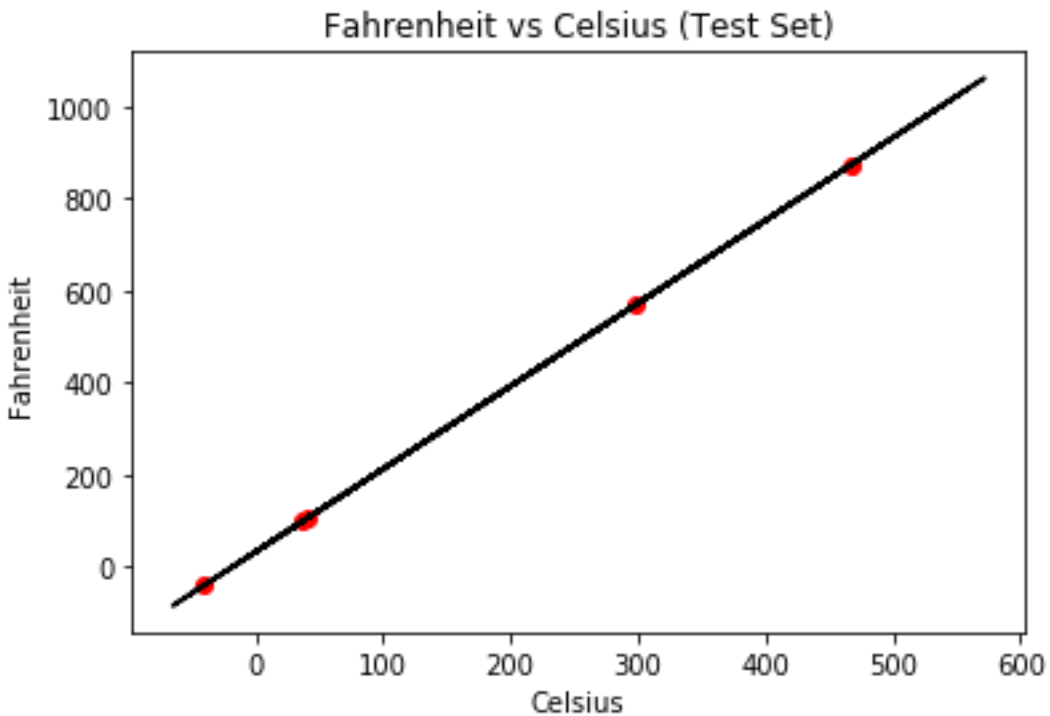
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
         normalize=False)
```

```
1  y_Pred = model.predict(X_Test)
2  y_Pred
```

```
array([570.2, 105.8,  98.6, 874.4, -40. ])
```

Python Uyg. (Regresyon)





Python Uyg. (Regresyon)

```
1 print("Fahrenheit = Celsius*{} + {}".format(model.coef_[0], model.intercept_) )
```

```
Fahrenheit = Celsius*1.800000000000001 + 31.999999999999773
```

Regresyon Modeli

y= b0 + b1*x1 == > y= Celcius*1.8 + 31.999

Python Uyg. (Regresyon)

Celsius 100 için Fahrenheit : ?

- Regresyon Modeli => y= Celcius*1.8 + 31.999
  100*1.8 + 31.999 = 211.999
- Cel2Fah => 100*1.8 + 32 = 212

```
y_new = model.predict([[100]])
print(y_new)
```

```
[212.]
```

```
def Cel2Fah(C):
    F = C * 1.8 + 32
    return F
```

```
print("Celsius to Fahrenheit:", Cel2Fah(100))
```

```
Celsius to Fahrenheit: 212.0
```

Modeller

- ❖Cel2Fah => Celsius*1.8 + 32
- ❖Sinir Ağı Modeli => Celcius*1.802 + 31.022
- ❖Regresyon Modeli => Celcius*1.8 + 31.999

Github Repository : https://github.com/metinuslu/kave2019_machinelearning

Teşekkürler

Metin USLU

İletişim

uslumetin@gmail.com

@metinuslu