Yıldız Technical University

Geographic Information Systems 2

Şevval Köntek

21046064

Table of Contents

PROJECT OBJECTIVE

This Geographic Information Systems (GIS) project was conducted to analyze the impacts of climate change on vegetation health in Turkey between 2016 and 2020. The project integrates NDVI data obtained from the Sentinel-2 satellite with key climatic and environmental parameters such as temperature, precipitation, soil moisture, and Sentinel-1 radar data to examine temporal and spatial variations in vegetation health.
Additionally, land use information was digitized, and the effects of these variables on vegetation health were evaluated using statistical models. The developed linear regression model aims to estimate the influence of climate data and land use on vegetation health, thereby contributing to the development of sustainable agricultural and environmental management strategies.
The project also seeks to provide farmers with data-driven insights to better understand the impacts of climate change and to optimize irrigation planning, crop selection, and agricultural practices accordingly.

PROBLEM STATEMENT

The effects of climate change on agriculture are becoming increasingly evident and demand urgent attention. Fluctuations in temperature, alterations in precipitation patterns, and extreme weather events such as droughts, floods, and frosts pose serious threats to agricultural production, reducing crop yields and negatively impacting soil and water quality.

MACHINE LEARNING INTEGRATION AND ANALYSES

Within the scope of this project, machine learning models were employed to investigate the relationship between climatic data and vegetation cover.

- **Data Used:**
  - Sentinel-2 NDVI (Vegetation Health Indicator)
  - Sentinel-1 VV Radar Data (providing information on soil structure and moisture content)
  - CHIRPS Precipitation Data
  - MOD11A1 Temperature Data
  - GLDAS Soil Moisture and Specific Humidity Data
  - Land Use (Annual Classification Data)
- **Data Processing:**
  - Raster datasets obtained from the Google Earth Engine (GEE) platform were processed within the same environment.
  - Data were converted into monthly averages and exported in CSV format.
  - All datasets were merged using Python, with inconsistencies and missing data cleaned.
  - Since land cover data are categorical, they were converted into numerical format using the One-Hot Encoding method.
- **Modeling Process:**
  - NDVI (vegetation health) was predicted using a Linear Regression model based on climatic variables and land cover information.
  - Model performance was evaluated using the R^2 score and Mean Squared Error (MSE).
  - Error analysis was conducted to determine under which conditions predictions were more accurate.

I. Sentinel-2 NDVI

NDVI (Normalized Difference Vegetation Index) data derived from Sentinel-2 satellite imagery provide valuable information about vegetation density and health.
NDVI values range between 0 and 1:

- High values (approximately 0.6 – 0.9) indicate dense and healthy vegetation.
- Low values (0 – 0.2) indicate poorly developed or stressed vegetation.

Below is the Google Earth Engine JavaScript code used to calculate monthly average NDVI values within Turkey's boundaries for the years 2016–2020 and export the results as a CSV file.

```javascript
// Türkiye sınırları
var turkey = ee.FeatureCollection('USDOS/LSIB_SIMPLE/2017')
               .filter(ee.Filter.eq('country_na', 'Turkey'));

// Başlangıç ve bitiş yılı
var startYear = 2016;
var endYear = 2020;

// NDVI hesaplayan fonksiyon (Sentinel-2 için)
function addNDVI(image) {
  var ndvi = image.normalizedDifference(['B8', 'B4']).rename('NDVI');
  return image.addBands(ndvi);
}

// Aylık NDVI'leri tutacak liste
var monthlyNDVI = [];

// Her yıl ve ay için dön
for (var year = startYear; year <= endYear; year++) {
  for (var month = 1; month <= 12; month++) {

    var startDate = ee.Date.fromYMD(year, month, 1);
    var endDate = startDate.advance(1, 'month');

    var collection = ee.ImageCollection('COPERNICUS/S2')
      .filterBounds(turkey)
      .filterDate(startDate, endDate)
      .filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE', 20))
      .map(addNDVI)
      .select('NDVI');

    var meanImage = collection.mean().clip(turkey)
      .set({
        'year': year,
        'month': month,
        'system:time_start': startDate.millis()
      });

    monthlyNDVI.push(meanImage);
  }
}

// Aylık NDVI koleksiyonu
var ndviCollection = ee.ImageCollection.fromImages(monthlyNDVI);

// Ortalama NDVI değerini çıkart ve Feature'e dönüştür
var features = ndviCollection.map(function(img){
  var stats = img.reduceRegion({
    reducer: ee.Reducer.mean(),
    geometry: turkey.geometry(),
    scale: 1000,
    maxPixels: 1e13
  });

  return ee.Feature(null, {
    'date': img.date().format('YYYY-MM'),
    'mean_NDVI': stats.get('NDVI')
  });
});

// CSV olarak dışa aktar
Export.table.toDrive({
  collection: features,
  description: 'Turkey_Monthly_Mean_NDVI_Sentinel2_2016_2020',
  fileFormat: 'CSV'
});
```

(Figure 1.1 – JavaScript code for downloading Sentinel-2 NDVI data)

B8 = Near Infrared (NIR), B4 = Red band → used to calculate NDVI.

Images with a cloud cover ratio below 20% were selected.

Monthly averages of NDVI values derived from Sentinel-2 imagery were exported in CSV format.

	A
1	Column1
2	system:index,date,mean_NDVI,.geo
3	0,2016-01,0.1788654594104548,"{""type"":""MultiPoint"",""coordinates"":[]}"
4	1,2016-02,0.1782420604088113,"{""type"":""MultiPoint"",""coordinates"":[]}"
5	2,2016-03,0.24251056801975507,"{""type"":""MultiPoint"",""coordinates"":[]}"
6	3,2016-04,0.3392672171174924 7,"{""type"":""MultiPoint"",""coordinates"":[]}"
7	4,2016-05,0.341194656919399,"{""type"":""MultiPoint"",""coordinates"":[]}"
8	5,2016-06,0.3305327683472788,"{""type"":""MultiPoint"",""coordinates"":[]}"
9	6,2016-07,0.29594312282481566,"{""type"":""MultiPoint"",""coordinates"":[]}"
10	7,2016-08,0.24965775046757652,"{""type"":""MultiPoint"",""coordinates"":[]}"
11	8,2016-09,0.240334025370648 7,"{""type"":""MultiPoint"",""coordinates"":[]}"
12	9,2016-10,0.2185955974222681,"{""type"":""MultiPoint"",""coordinates"":[]}"
13	10,2016-11,0.1859755385428084,"{""type"":""MultiPoint"",""coordinates"":[]}"
14	11,2016-12,0.12796806358975021,"{""type"":""MultiPoint"",""coordinates"":[]}"
15	12,2017-01,0.09084066645463339,"{""type"":""MultiPoint"",""coordinates"":[]}"
16	13,2017-02,0.10767234604136186,"{""type"":""MultiPoint"",""coordinates"":[]}"
17	14,2017-03,0.18051713660247473,"{""type"":""MultiPoint"",""coordinates"":[]}"
18	15,2017-04,0.28835663321939525,"{""type"":""MultiPoint"",""coordinates"":[]}"
19	16,2017-05,0.33284593824442016,"{""type"":""MultiPoint"",""coordinates"":[]}"
20	17,2017-06,0.3632347373805914,"{""type"":""MultiPoint"",""coordinates"":[]}"
21	18,2017-07,0.3121681965130157,"{""type"":""MultiPoint"",""coordinates"":[]}"
22	19,2017-08,0.2602038524168834,"{""type"":""MultiPoint"",""coordinates"":[]}"
23	20,2017-09,0.25034320063373117,"{""type"":""MultiPoint"",""coordinates"":[]}"
24	21,2017-10,0.23823346382423083,"{""type"":""MultiPoint"",""coordinates"":[]}"
25	22,2017-11,0.20665119356430828,"{""type"":""MultiPoint"",""coordinates"":[]}"
26	23,2017-12,0.16823304829393457,"{""type"":""MultiPoint"",""coordinates"":[]}"
27	24,2018-01,0.17640953806611526,"{""type"":""MultiPoint"",""coordinates"":[]}"
28	25,2018-02,0.15529847892283624,"{""type"":""MultiPoint"",""coordinates"":[]}"
29	26,2018-03,0.26498356771838943,"{""type"":""MultiPoint"",""coordinates"":[]}"
30	27,2018-04,0.3534362833909234,"{""type"":""MultiPoint"",""coordinates"":[]}"

Turkey_Monthly_Mean_NDVI_Sentin | Sayfa1 | +

34	31,2018-08,0.26610988853860734,"{""type"":""MultiPoint"",""coordinates"":[]}"
35	32,2018-09,0.25139859579726787,"{""type"":""MultiPoint"",""coordinates"":[]}"
36	33,2018-10,0.21702538452400988,"{""type"":""MultiPoint"",""coordinates"":[]}"
37	34,2018-11,0.21538064200223125,"{""type"":""MultiPoint"",""coordinates"":[]}"
38	35,2018-12,0.17850998260244783,"{""type"":""MultiPoint"",""coordinates"":[]}"
39	36,2019-01,0.11822188778494083,"{""type"":""MultiPoint"",""coordinates"":[]}"
40	37,2019-02,0.19436028280472004,"{""type"":""MultiPoint"",""coordinates"":[]}"
41	38,2019-03,0.22291168938354042,"{""type"":""MultiPoint"",""coordinates"":[]}"
42	39,2019-04,0.27863251249585175,"{""type"":""MultiPoint"",""coordinates"":[]}"
43	40,2019-05,0.3591136735947753,"{""type"":""MultiPoint"",""coordinates"":[]}"
44	41,2019-06,0.32552069105997933,"{""type"":""MultiPoint"",""coordinates"":[]}"
45	42,2019-07,0.3172256039383575,"{""type"":""MultiPoint"",""coordinates"":[]}"
46	43,2019-08,0.28050642805134807,"{""type"":""MultiPoint"",""coordinates"":[]}"
47	44,2019-09,0.2600765880716091,"{""type"":""MultiPoint"",""coordinates"":[]}"
48	45,2019-10,0.23471334969639357,"{""type"":""MultiPoint"",""coordinates"":[]}"
49	46,2019-11,0.21385883677450016,"{""type"":""MultiPoint"",""coordinates"":[]}"
50	47,2019-12,0.17354998749074954,"{""type"":""MultiPoint"",""coordinates"":[]}"
51	48,2020-01,0.13984659923349102,"{""type"":""MultiPoint"",""coordinates"":[]}"
52	49,2020-02,0.18590813020437083,"{""type"":""MultiPoint"",""coordinates"":[]}"
53	50,2020-03,0.20389988635776682,"{""type"":""MultiPoint"",""coordinates"":[]}"
54	51,2020-04,0.3126130895317023,"{""type"":""MultiPoint"",""coordinates"":[]}"
55	52,2020-05,0.3633907979172662,"{""type"":""MultiPoint"",""coordinates"":[]}"
56	53,2020-06,0.34652049365292137,"{""type"":""MultiPoint"",""coordinates"":[]}"
57	54,2020-07,0.3096699484487806,"{""type"":""MultiPoint"",""coordinates"":[]}"
58	55,2020-08,0.28282281254718544,"{""type"":""MultiPoint"",""coordinates"":[]}"
59	56,2020-09,0.24312956794975832,"{""type"":""MultiPoint"",""coordinates"":[]}"
60	57,2020-10,0.23053126921375328,"{""type"":""MultiPoint"",""coordinates"":[]}"
61	58,2020-11,0.211781213679550 7,"{""type"":""MultiPoint"",""coordinates"":[]}"
62	59,2020-12,0.16808783950274842,"{""type"":""MultiPoint"",""coordinates"":[]}"
63	

Turkey_Monthly_Mean_NDVI_Sentin | Sayfa1 | +

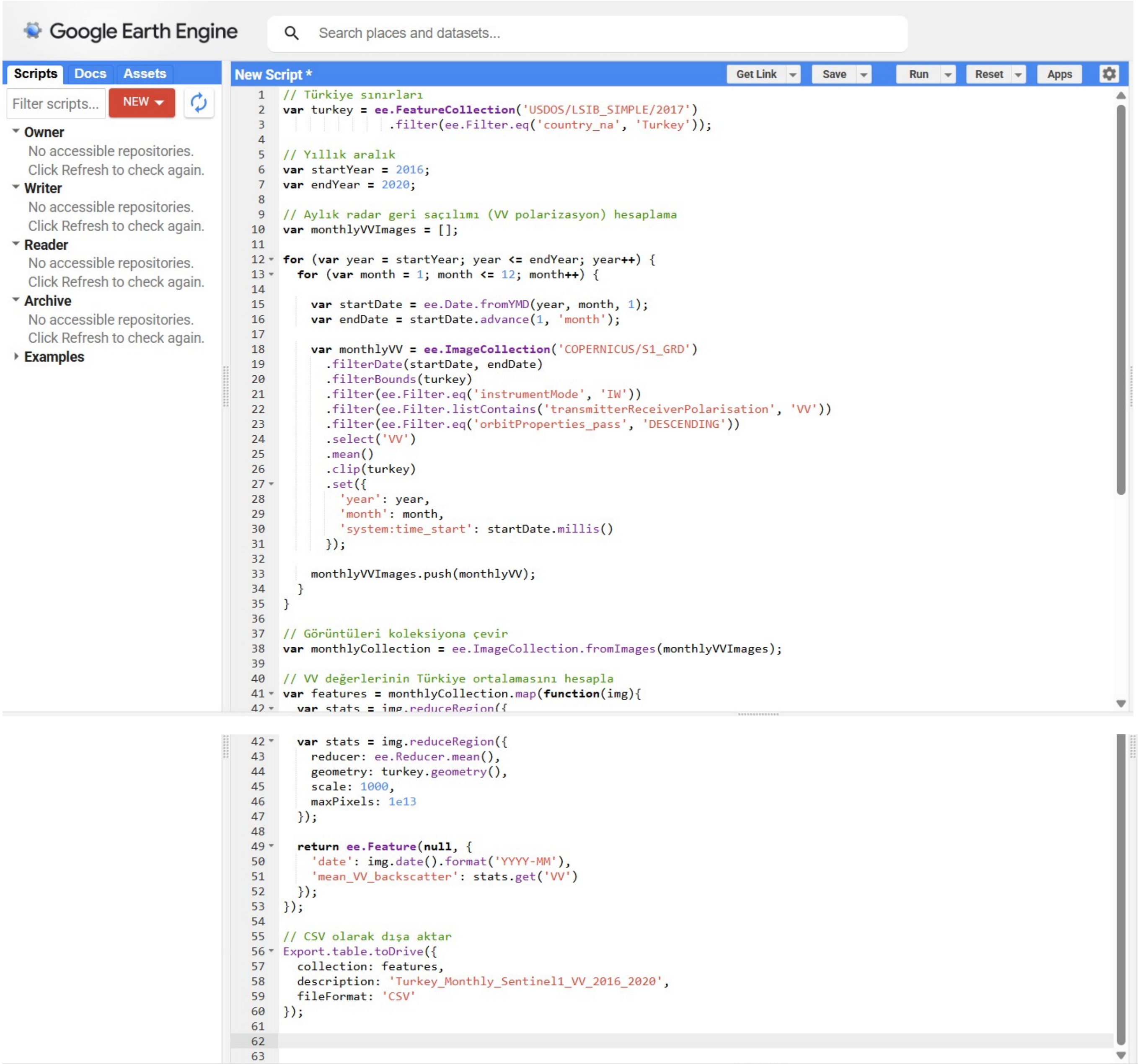
(Figure 1.2 – Screenshot of the downloaded Sentinel-2 NDVI CSV data)

2. Sentinel-1 VV Polarization Radar Backscatter

VV-polarized radar data (vertical transmit, vertical receive) from the Sentinel-1 satellite were used to analyze surface structure and soil moisture conditions. The radar backscatter data derived from Sentinel-1 GRD (Ground Range Detected) products are widely used for soil moisture and surface characterization.

- High VV backscatter values generally correspond to hard and artificial surfaces (e.g., buildings, asphalt).
- Low VV values indicate moist soil or smooth surfaces.

Monthly averages of VV-polarized Sentinel-1 SAR data for Turkey between 2016–2020 were computed and exported in CSV format. The relevant Google Earth Engine JavaScript code is provided below.




```javascript
// Türkiye sınırları
var turkey = ee.FeatureCollection('USDOS/LSIB_SIMPLE/2017')
               .filter(ee.Filter.eq('country_na', 'Turkey'));

// Yıllık aralık
var startYear = 2016;
var endYear = 2020;

// Aylık radar geri saçılımı (VV polarizasyon) hesaplama
var monthlyVVImages = [];

for (var year = startYear; year <= endYear; year++) {
  for (var month = 1; month <= 12; month++) {

    var startDate = ee.Date.fromYMD(year, month, 1);
    var endDate = startDate.advance(1, 'month');

    var monthlyVV = ee.ImageCollection('COPERNICUS/S1_GRD')
      .filterDate(startDate, endDate)
      .filterBounds(turkey)
      .filter(ee.Filter.eq('instrumentMode', 'IW'))
      .filter(ee.Filter.listContains('transmitterReceiverPolarisation', 'VV'))
      .filter(ee.Filter.eq('orbitProperties_pass', 'DESCENDING'))
      .select('VV')
      .mean()
      .clip(turkey)
      .set({
        'year': year,
        'month': month,
        'system:time_start': startDate.millis()
      });

    monthlyVVImages.push(monthlyVV);
  }
}

// Görüntüleri koleksiyona çevir
var monthlyCollection = ee.ImageCollection.fromImages(monthlyVVImages);

// VV değerlerinin Türkiye ortalamasını hesapla
var features = monthlyCollection.map(function(img){
  var stats = img.reduceRegion({
    reducer: ee.Reducer.mean(),
    geometry: turkey.geometry(),
    scale: 1000,
    maxPixels: 1e13
  });

  return ee.Feature(null, {
    'date': img.date().format('YYYY-MM'),
    'mean_VV_backscatter': stats.get('VV')
  });
});

// CSV olarak dışa aktar
Export.table.toDrive({
  collection: features,
  description: 'Turkey_Monthly_Sentinel1_VV_2016_2020',
  fileFormat: 'CSV'
});
```

(Figure 2.1 – JavaScript code for downloading Sentinel-1 SAR data)

VV value: Represents radar backscatter indicative of soil moisture. Moister surfaces typically exhibit higher backscatter.

CSV output of Sentinel-1 VV Polarization Radar Backscatter data.

1	system:index	date	mean_VV_backscatter	.geo
2	0	1.01.2016	-1.04394E+16	{"type":"MultiPoint","coordinates":[]}
3	1	1.02.2016	-1.05078E+16	{"type":"MultiPoint","coordinates":[]}
4	2	1.03.2016	-1.04498E+16	{"type":"MultiPoint","coordinates":[]}
5	3	1.04.2016	-1.09048E+15	{"type":"MultiPoint","coordinates":[]}
6	4	1.05.2016	-1.01537E+15	{"type":"MultiPoint","coordinates":[]}
7	5	1.06.2016	-1.00624E+16	{"type":"MultiPoint","coordinates":[]}
8	6	1.07.2016	-1.03775E+16	{"type":"MultiPoint","coordinates":[]}
9	7	1.08.2016	-1.05471E+16	{"type":"MultiPoint","coordinates":[]}
10	8	1.09.2016	-1.07048E+16	{"type":"MultiPoint","coordinates":[]}
11	9	1.10.2016	-1.08429E+16	{"type":"MultiPoint","coordinates":[]}
12	10	1.11.2016	-1.08715E+16	{"type":"MultiPoint","coordinates":[]}
13	11	1.12.2016	-1.08362E+16	{"type":"MultiPoint","coordinates":[]}
14	12	1.01.2017	-1.07436E+15	{"type":"MultiPoint","coordinates":[]}
15	13	1.02.2017	-1.11285E+16	{"type":"MultiPoint","coordinates":[]}
16	14	1.03.2017	-1.06802E+16	{"type":"MultiPoint","coordinates":[]}
17	15	1.04.2017	-1.07534E+16	{"type":"MultiPoint","coordinates":[]}
18	16	1.05.2017	-1.02371E+16	{"type":"MultiPoint","coordinates":[]}
19	17	1.06.2017	-1.00384E+16	{"type":"MultiPoint","coordinates":[]}
20	18	1.07.2017	-1.01499E+15	{"type":"MultiPoint","coordinates":[]}
21	19	1.08.2017	-1.02799E+16	{"type":"MultiPoint","coordinates":[]}
22	20	1.09.2017	-1.05042E+16	{"type":"MultiPoint","coordinates":[]}
23	21	1.10.2017	-1.04885E+16	{"type":"MultiPoint","coordinates":[]}
24	22	1.11.2017	-1.01575E+16	{"type":"MultiPoint","coordinates":[]}
25	23	1.12.2017	-1.01318E+16	{"type":"MultiPoint","coordinates":[]}
26	24	1.01.2018	-1.00385E+16	{"type":"MultiPoint","coordinates":[]}

Turkey_Monthly_Sentinel1_VV_201 Sayfa1 +

37	35	1.12.2018	-9.32994E+15	{"type":"MultiPoint","coordinates":[]}
38	36	1.01.2019	-9.82562E+15	{"type":"MultiPoint","coordinates":[]}
39	37	1.02.2019	-1.01285E+15	{"type":"MultiPoint","coordinates":[]}
40	38	1.03.2019	-1.06673E+16	{"type":"MultiPoint","coordinates":[]}
41	39	1.04.2019	-1.05126E+15	{"type":"MultiPoint","coordinates":[]}
42	40	1.05.2019	-1.03231E+16	{"type":"MultiPoint","coordinates":[]}
43	41	1.06.2019	-9.86653E+15	{"type":"MultiPoint","coordinates":[]}
44	42	1.07.2019	-9.97806E+15	{"type":"MultiPoint","coordinates":[]}
45	43	1.08.2019	-1.01592E+15	{"type":"MultiPoint","coordinates":[]}
46	44	1.09.2019	-1.03972E+16	{"type":"MultiPoint","coordinates":[]}
47	45	1.10.2019	-1.04255E+16	{"type":"MultiPoint","coordinates":[]}
48	46	1.11.2019	-1.05664E+16	{"type":"MultiPoint","coordinates":[]}
49	47	1.12.2019	-9.79163E+14	{"type":"MultiPoint","coordinates":[]}
50	48	1.01.2020	-1.03581E+16	{"type":"MultiPoint","coordinates":[]}
51	49	1.02.2020	-1.03398E+15	{"type":"MultiPoint","coordinates":[]}
52	50	1.03.2020	-1.04329E+15	{"type":"MultiPoint","coordinates":[]}
53	51	1.04.2020	-1.0478E+15	{"type":"MultiPoint","coordinates":[]}
54	52	1.05.2020	-9.987E+15	{"type":"MultiPoint","coordinates":[]}
55	53	1.06.2020	-9.86759E+15	{"type":"MultiPoint","coordinates":[]}
56	54	1.07.2020	-9.92878E+15	{"type":"MultiPoint","coordinates":[]}
57	55	1.08.2020	-1.01547E+16	{"type":"MultiPoint","coordinates":[]}
58	56	1.09.2020	-1.00595E+16	{"type":"MultiPoint","coordinates":[]}
59	57	1.10.2020	-1.03693E+16	{"type":"MultiPoint","coordinates":[]}
60	58	1.11.2020	-1.03726E+15	{"type":"MultiPoint","coordinates":[]}
61	59	1.12.2020	-1.01716E+16	{"type":"MultiPoint","coordinates":[]}
62				

Turkey_Monthly_Sentinel1_VV_201 Sayfa1 +

(Figure 2.2 – Screenshot of the downloaded Sentinel-1 SAR CSV data)

3. CHIRPS Monthly Average Precipitation Data for Turkey

The CHIRPS (Climate Hazards Group InfraRed Precipitation with Station Data) dataset is widely used for regional-scale long-term precipitation analyses. It provides daily or monthly total precipitation (in mm) and is valuable for drought monitoring and agricultural planning.

Using CHIRPS satellite data, monthly average precipitation values for Turkey between 2016–2020 were calculated and exported in CSV format.

:

```javascript
// Yıllık aralık
var startYear = 2016;
var endYear = 2020;

// Türkiye sınırları
var turkey = ee.FeatureCollection('USDOS/LSIB_SIMPLE/2017')
               .filter(ee.Filter.eq('country_na', 'Turkey'));

// Aylık ortalama yağışları tutacak liste
var monthlyImages = [];

// Her yıl ve her ay için dön
for (var year = startYear; year <= endYear; year++) {
  for (var month = 1; month <= 12; month++) {

    var startDate = ee.Date.fromYMD(year, month, 1);
    var endDate = startDate.advance(1, 'month');

    var monthlyMean = ee.ImageCollection('UCSB-CHG/CHIRPS/DAILY')
      .filterDate(startDate, endDate)
      .filterBounds(turkey)
      .mean()
      .clip(turkey)
      .set({
        'year': year,
        'month': month,
        'system:time_start': startDate.millis()
      });

    monthlyImages.push(monthlyMean);
  }
}

// Aylık verileri bir koleksiyon olarak tanımla
var monthlyCollection = ee.ImageCollection.fromImages(monthlyImages);

// Örnek: 2020 Ocak yağışını haritada göster
var sample = monthlyCollection.filter(ee.Filter.and(
  ee.Filter.eq('year', 2020),
  ee.Filter.eq('month', 1)
)).first();

Map.centerObject(turkey, 6);
Map.addLayer(sample, {
  min: 0,
  max: 100,
  palette: ['lightblue', 'blue', 'darkblue']
}, 'Ocak 2020 Yağış');


var features = monthlyCollection.map(function(img){
  var stats = img.reduceRegion({
    reducer: ee.Reducer.mean(),
    geometry: turkey.geometry(),
    scale: 5000,
    maxPixels: 1e13
  });

  return ee.Feature(null, {
    'date': img.date().format('YYYY-MM'),
    'mean_precip_mm': stats.get('precipitation')
  });
});

Export.table.toDrive({
  collection: features,
  description: 'Turkey_Monthly_Mean_Precip_2016_2020',
  fileFormat: 'CSV'
});
```

(Figure 3.1 – JavaScript code for downloading CHIRPS data)

- **CHIRPS verilerinin CSV çıktısı**

	A	B	C	D
1	system:index	date	mean_precip_mm	.geo
2	0	1.01.2016	3.23135E+15	{"type":"MultiPoint","coordinates":[]}
3	1	1.02.2016	1.80729E+14	{"type":"MultiPoint","coordinates":[]}
4	2	1.03.2016	2.30865E+16	{"type":"MultiPoint","coordinates":[]}
5	3	1.04.2016	1.3158E+16	{"type":"MultiPoint","coordinates":[]}
6	4	1.05.2016	2.34688E+15	{"type":"MultiPoint","coordinates":[]}
7	5	1.06.2016	1.20612E+16	{"type":"MultiPoint","coordinates":[]}
8	6	1.07.2016	4.76078E+16	{"type":"MultiPoint","coordinates":[]}
9	7	1.08.2016	5.33825E+15	{"type":"MultiPoint","coordinates":[]}
10	8	1.09.2016	8.56902E+15	{"type":"MultiPoint","coordinates":[]}
11	9	1.10.2016	7.428E+14	{"type":"MultiPoint","coordinates":[]}
12	10	1.11.2016	1.99734E+16	{"type":"MultiPoint","coordinates":[]}
13	11	1.12.2016	2.897E+16	{"type":"MultiPoint","coordinates":[]}
14	12	1.01.2017	2.23074E+16	{"type":"MultiPoint","coordinates":[]}
15	13	1.02.2017	8.60201E+15	{"type":"MultiPoint","coordinates":[]}
16	14	1.03.2017	2.04944E+15	{"type":"MultiPoint","coordinates":[]}
17	15	1.04.2017	2.00151E+15	{"type":"MultiPoint","coordinates":[]}
18	16	1.05.2017	2.29167E+16	{"type":"MultiPoint","coordinates":[]}
19	17	1.06.2017	1.24694E+16	{"type":"MultiPoint","coordinates":[]}
20	18	1.07.2017	4.25973E+16	{"type":"MultiPoint","coordinates":[]}
21	19	1.08.2017	5.01624E+15	{"type":"MultiPoint","coordinates":[]}
22	20	1.09.2017	4.76937E+15	{"type":"MultiPoint","coordinates":[]}
23	21	1.10.2017	1.60349E+16	{"type":"MultiPoint","coordinates":[]}
24	22	1.11.2017	2.18597E+16	{"type":"MultiPoint","coordinates":[]}
25	23	1.12.2017	2.14354E+15	{"type":"MultiPoint","coordinates":[]}
26	24	1.01.2018	2.80557E+15	{"type":"MultiPoint","coordinates":[]}

Turkey_Monthly_Mean_Precip_2016 | Sayfa1 | +

37	35	1.12.2018	4.51118E+16	{"type":"MultiPoint","coordinates":[]}
38	36	1.01.2019	4.12964E+15	{"type":"MultiPoint","coordinates":[]}
39	37	1.02.2019	2.30497E+16	{"type":"MultiPoint","coordinates":[]}
40	38	1.03.2019	2.10665E+15	{"type":"MultiPoint","coordinates":[]}
41	39	1.04.2019	2.38183E+16	{"type":"MultiPoint","coordinates":[]}
42	40	1.05.2019	1.668E+15	{"type":"MultiPoint","coordinates":[]}
43	41	1.06.2019	1.72231E+15	{"type":"MultiPoint","coordinates":[]}
44	42	1.07.2019	6.82617E+15	{"type":"MultiPoint","coordinates":[]}
45	43	1.08.2019	5.26114E+15	{"type":"MultiPoint","coordinates":[]}
46	44	1.09.2019	5.2737E+14	{"type":"MultiPoint","coordinates":[]}
47	45	1.10.2019	9.98662E+15	{"type":"MultiPoint","coordinates":[]}
48	46	1.11.2019	1.42109E+16	{"type":"MultiPoint","coordinates":[]}
49	47	1.12.2019	3.08569E+15	{"type":"MultiPoint","coordinates":[]}
50	48	1.01.2020	2.00021E+14	{"type":"MultiPoint","coordinates":[]}
51	49	1.02.2020	2.38045E+16	{"type":"MultiPoint","coordinates":[]}
52	50	1.03.2020	2.25536E+15	{"type":"MultiPoint","coordinates":[]}
53	51	1.04.2020	1.89535E+16	{"type":"MultiPoint","coordinates":[]}
54	52	1.05.2020	2.00731E+15	{"type":"MultiPoint","coordinates":[]}
55	53	1.06.2020	1.39407E+16	{"type":"MultiPoint","coordinates":[]}
56	54	1.07.2020	4.68371E+16	{"type":"MultiPoint","coordinates":[]}
57	55	1.08.2020	3.75432E+15	{"type":"MultiPoint","coordinates":[]}
58	56	1.09.2020	5.13575E+15	{"type":"MultiPoint","coordinates":[]}
59	57	1.10.2020	1.01804E+16	{"type":"MultiPoint","coordinates":[]}
60	58	1.11.2020	1.40845E+16	{"type":"MultiPoint","coordinates":[]}
61	59	1.12.2020	1.71391E+16	{"type":"MultiPoint","coordinates":[]}
62				

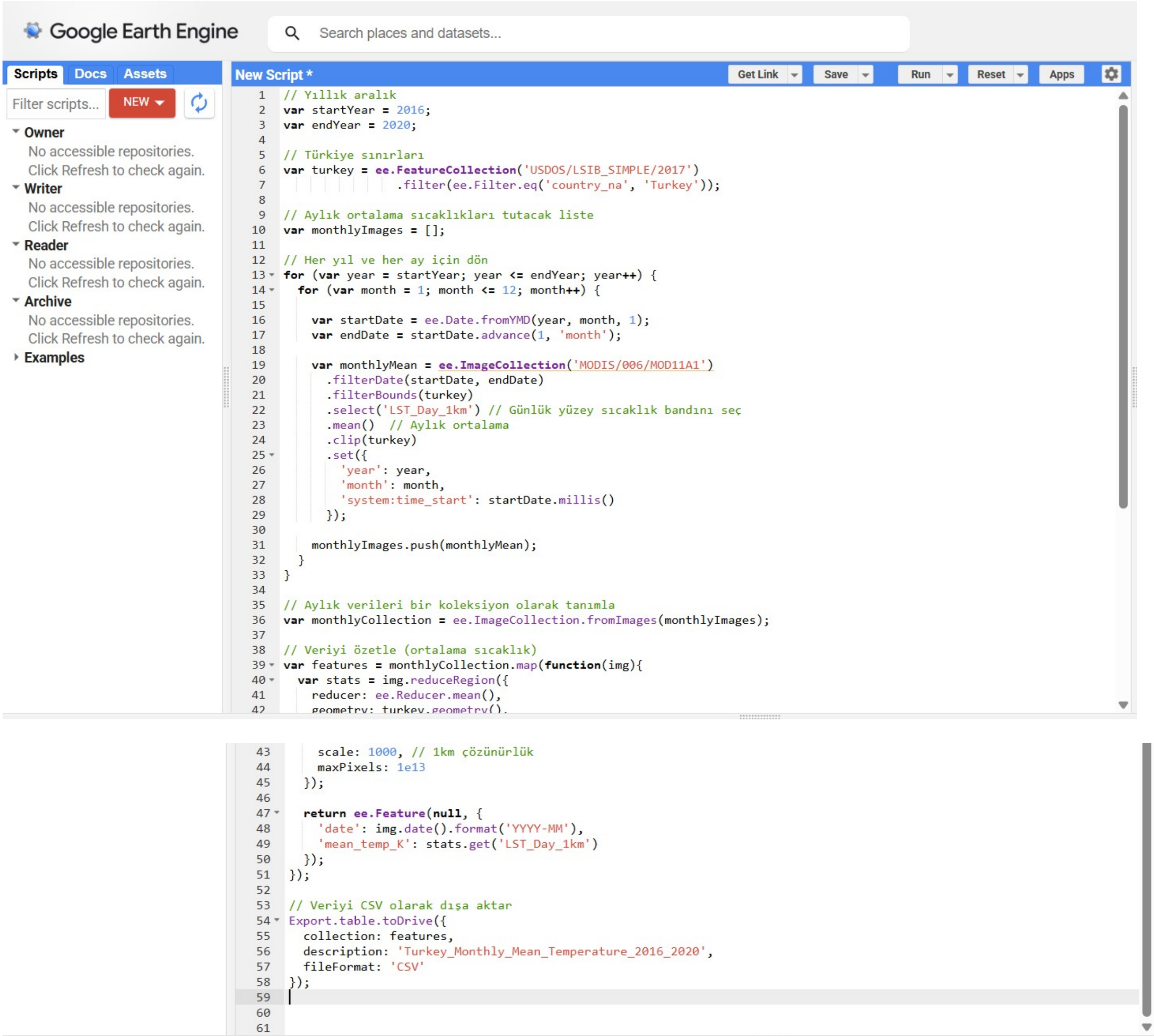
Turkey_Monthly_Mean_Precip_2016 | Sayfa1 | +

(Figure 3.2 – Screenshot of the downloaded CHIRPS CSV data)

4. MOD11A1 Monthly Average Temperature Data for Turkey

The MOD11A1 product, provided by the MODIS sensor, supplies daily land surface temperature (LST) data, separately for daytime and nighttime. It is particularly useful for monitoring seasonal temperature variations.

Using MOD11A1 satellite data, land surface temperatures across Turkey for the 2016–2020 period were calculated and exported in CSV format.




```javascript
// Yıllık aralık
var startYear = 2016;
var endYear = 2020;

// Türkiye sınırları
var turkey = ee.FeatureCollection('USDOS/LSIB_SIMPLE/2017')
                 .filter(ee.Filter.eq('country_na', 'Turkey'));

// Aylık ortalama sıcaklıkları tutacak liste
var monthlyImages = [];

// Her yıl ve her ay için dön
for (var year = startYear; year <= endYear; year++) {
  for (var month = 1; month <= 12; month++) {

    var startDate = ee.Date.fromYMD(year, month, 1);
    var endDate = startDate.advance(1, 'month');

    var monthlyMean = ee.ImageCollection('MODIS/006/MOD11A1')
      .filterDate(startDate, endDate)
      .filterBounds(turkey)
      .select('LST_Day_1km') // Günlük yüzey sıcaklık bandını seç
      .mean()  // Aylık ortalama
      .clip(turkey)
      .set({
        'year': year,
        'month': month,
        'system:time_start': startDate.millis()
      });

    monthlyImages.push(monthlyMean);
  }
}

// Aylık verileri bir koleksiyon olarak tanımla
var monthlyCollection = ee.ImageCollection.fromImages(monthlyImages);

// Veriyi özetle (ortalama sıcaklık)
var features = monthlyCollection.map(function(img){
  var stats = img.reduceRegion({
    reducer: ee.Reducer.mean(),
    geometry: turkey.geometry(),
    scale: 1000, // 1km çözünürlük
    maxPixels: 1e13
  });

  return ee.Feature(null, {
    'date': img.date().format('YYYY-MM'),
    'mean_temp_K': stats.get('LST_Day_1km')
  });
});

// Veriyi CSV olarak dışa aktar
Export.table.toDrive({
  collection: features,
  description: 'Turkey_Monthly_Mean_Temperature_2016_2020',
  fileFormat: 'CSV'
});
```

((Figure 4.1 – JavaScript code for downloading MOD11A1 data)

- **MOD11A1 verilerinin CSV görüntüsü**

	system:index	date	mean_temp_K	.geo
2	0	1.01.2016	1.36724E+16	{"type":"MultiPoint","coordinates":[]}
3	1	1.02.2016	1.41033E+15	{"type":"MultiPoint","coordinates":[]}
4	2	1.03.2016	1.44233E+16	{"type":"MultiPoint","coordinates":[]}
5	3	1.04.2016	1.48774E+16	{"type":"MultiPoint","coordinates":[]}
6	4	1.05.2016	1.49387E+16	{"type":"MultiPoint","coordinates":[]}
7	5	1.06.2016	1.52662E+16	{"type":"MultiPoint","coordinates":[]}
8	6	1.07.2016	1.54865E+15	{"type":"MultiPoint","coordinates":[]}
9	7	1.08.2016	1.55264E+15	{"type":"MultiPoint","coordinates":[]}
10	8	1.09.2016	1.52287E+16	{"type":"MultiPoint","coordinates":[]}
11	9	1.10.2016	1.492E+16	{"type":"MultiPoint","coordinates":[]}
12	10	1.11.2016	1.43385E+16	{"type":"MultiPoint","coordinates":[]}
13	11	1.12.2016	1.36782E+16	{"type":"MultiPoint","coordinates":[]}
14	12	1.01.2017	1.36155E+16	{"type":"MultiPoint","coordinates":[]}
15	13	1.02.2017	1.3913E+16	{"type":"MultiPoint","coordinates":[]}
16	14	1.03.2017	1.44579E+16	{"type":"MultiPoint","coordinates":[]}
17	15	1.04.2017	1.48162E+16	{"type":"MultiPoint","coordinates":[]}
18	16	1.05.2017	1.49652E+16	{"type":"MultiPoint","coordinates":[]}
19	17	1.06.2017	1.52783E+16	{"type":"MultiPoint","coordinates":[]}
20	18	1.07.2017	1.55564E+16	{"type":"MultiPoint","coordinates":[]}
21	19	1.08.2017	1.55172E+16	{"type":"MultiPoint","coordinates":[]}
22	20	1.09.2017	1.54204E+16	{"type":"MultiPoint","coordinates":[]}
23	21	1.10.2017	1.4813E+16	{"type":"MultiPoint","coordinates":[]}
24	22	1.11.2017	1.43335E+16	{"type":"MultiPoint","coordinates":[]}
25	23	1.12.2017	1.40433E+16	{"type":"MultiPoint","coordinates":[]}
26	24	1.01.2018	1.38824E+16	{"type":"MultiPoint","coordinates":[]}
37	35	1.12.2018	1.38578E+16	{"type":"MultiPoint","coordinates":[]}
38	36	1.01.2019	1.36605E+16	{"type":"MultiPoint","coordinates":[]}
39	37	1.02.2019	1.40449E+16	{"type":"MultiPoint","coordinates":[]}
40	38	1.03.2019	1.43421E+15	{"type":"MultiPoint","coordinates":[]}
41	39	1.04.2019	1.46612E+16	{"type":"MultiPoint","coordinates":[]}
42	40	1.05.2019	1.50597E+16	{"type":"MultiPoint","coordinates":[]}
43	41	1.06.2019	1.52957E+16	{"type":"MultiPoint","coordinates":[]}
44	42	1.07.2019	1.54306E+16	{"type":"MultiPoint","coordinates":[]}
45	43	1.08.2019	1.54713E+15	{"type":"MultiPoint","coordinates":[]}
46	44	1.09.2019	1.52419E+16	{"type":"MultiPoint","coordinates":[]}
47	45	1.10.2019	1.49584E+16	{"type":"MultiPoint","coordinates":[]}
48	46	1.11.2019	1.454E+15	{"type":"MultiPoint","coordinates":[]}
49	47	1.12.2019	1.40385E+16	{"type":"MultiPoint","coordinates":[]}
50	48	1.01.2020	1.37635E+16	{"type":"MultiPoint","coordinates":[]}
51	49	1.02.2020	1.39245E+16	{"type":"MultiPoint","coordinates":[]}
52	50	1.03.2020	1.43999E+16	{"type":"MultiPoint","coordinates":[]}
53	51	1.04.2020	1.47451E+16	{"type":"MultiPoint","coordinates":[]}
54	52	1.05.2020	1.50566E+15	{"type":"MultiPoint","coordinates":[]}
55	53	1.06.2020	1.52645E+16	{"type":"MultiPoint","coordinates":[]}
56	54	1.07.2020	1.551E+16	{"type":"MultiPoint","coordinates":[]}
57	55	1.08.2020	1.55196E+16	{"type":"MultiPoint","coordinates":[]}
58	56	1.09.2020	1.54003E+16	{"type":"MultiPoint","coordinates":[]}
59	57	1.10.2020	1.5028E+16	{"type":"MultiPoint","coordinates":[]}
60	58	1.11.2020	1.43364E+16	{"type":"MultiPoint","coordinates":[]}
61	59	1.12.2020	1.40783E+16	{"type":"MultiPoint","coordinates":[]}

Turkey_Monthly_Mean_Temperature | Sayfa1

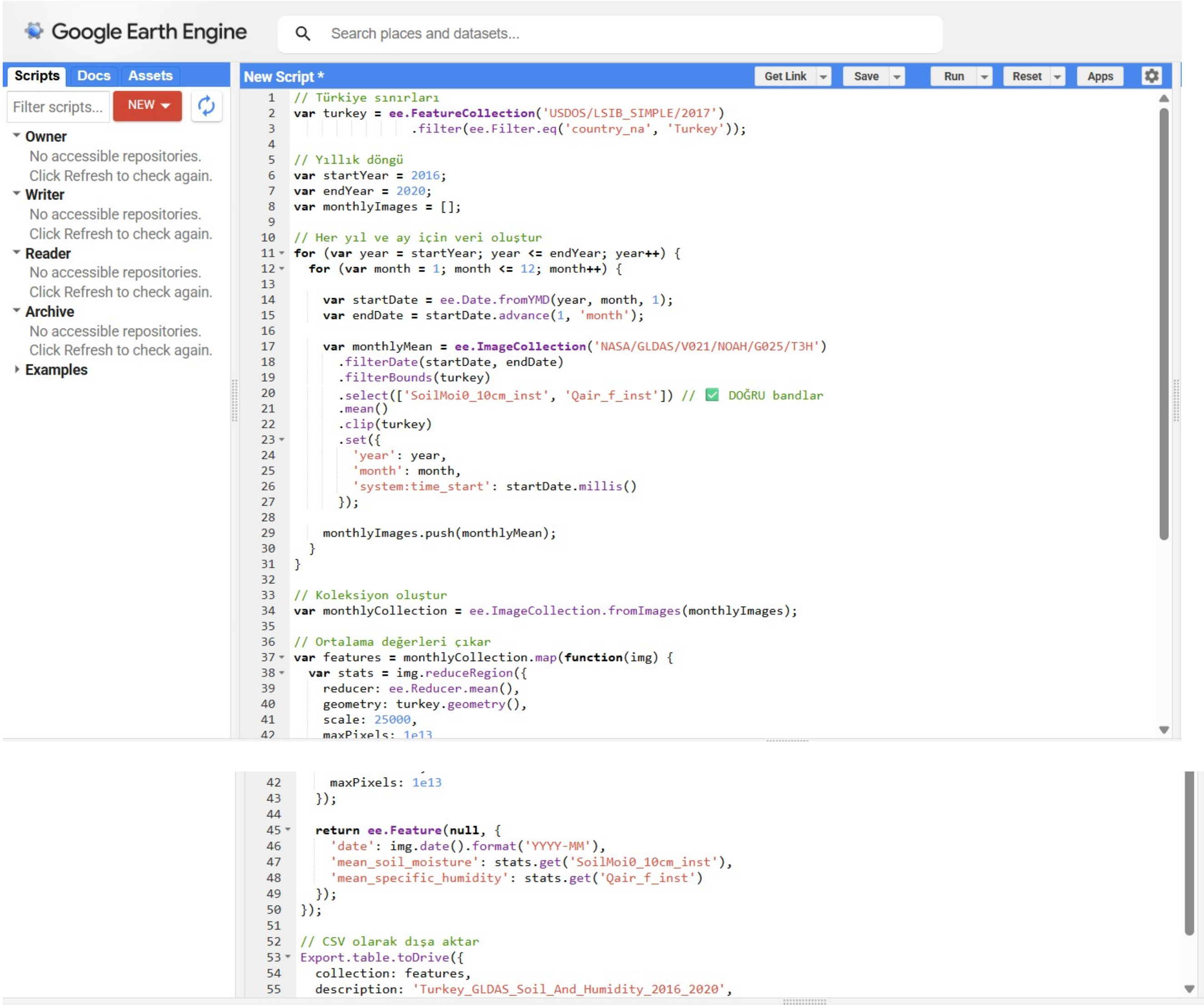
(Figure 4.2 – Screenshot of the downloaded MOD11A1 CSV data)

5. GLDAS Soil Moisture and Atmospheric Humidity (2016–2020, Turkey)

The Global Land Data Assimilation System (GLDAS) dataset includes parameters such as surface soil moisture and specific humidity.

- **Soil moisture** plays a critical role in determining the amount of water available for plant growth.
- **Specific humidity** represents the amount of water vapor in the atmosphere and is used to evaluate climatic conditions.

Using GLDAS satellite data, surface soil moisture and atmospheric specific humidity across Turkey for 2016–2020 were calculated and exported in CSV format.



(Figure 5.1 – JavaScript code for downloading GLDAS data)

- **GLDAS verilerinin CSV çıktısı**

	A	B	C	D	E
1	system:index	date	mean_soil_moisture	mean_specific_humidity	.geo
2	0	1.01.2016	3.27071E+15	3.34946E+15	{"type":"MultiPoint","coordinates":[]}
3	1	1.02.2016	3.34699E+15	4.48526E+15	{"type":"MultiPoint","coordinates":[]}
4	2	1.03.2016	3.06897E+16	4.21809E+16	{"type":"MultiPoint","coordinates":[]}
5	3	1.04.2016	2.62293E+15	5.04352E+15	{"type":"MultiPoint","coordinates":[]}
6	4	1.05.2016	2.58718E+16	6.77522E+15	{"type":"MultiPoint","coordinates":[]}
7	5	1.06.2016	2.20606E+16	8.15956E+15	{"type":"MultiPoint","coordinates":[]}
8	6	1.07.2016	1.67871E+16	8.53663E+15	{"type":"MultiPoint","coordinates":[]}
9	7	1.08.2016	1.46303E+16	8.92812E+15	{"type":"MultiPoint","coordinates":[]}
10	8	1.09.2016	1.64626E+16	6.74581E+15	{"type":"MultiPoint","coordinates":[]}
11	9	1.10.2016	1.67867E+15	5.40521E+15	{"type":"MultiPoint","coordinates":[]}
12	10	1.11.2016	2.07117E+15	3.68323E+15	{"type":"MultiPoint","coordinates":[]}
13	11	1.12.2016	2.82572E+16	2.88712E+16	{"type":"MultiPoint","coordinates":[]}
14	12	1.01.2017	3.57909E+15	2.82993E+16	{"type":"MultiPoint","coordinates":[]}
15	13	1.02.2017	3.41254E+15	3.27609E+15	{"type":"MultiPoint","coordinates":[]}
16	14	1.03.2017	3.1783E+16	4.34786E+15	{"type":"MultiPoint","coordinates":[]}
17	15	1.04.2017	2.8773E+15	4.76563E+15	{"type":"MultiPoint","coordinates":[]}
18	16	1.05.2017	2.6233E+16	6.64142E+15	{"type":"MultiPoint","coordinates":[]}
19	17	1.06.2017	2.2421E+16	7.9604E+15	{"type":"MultiPoint","coordinates":[]}
20	18	1.07.2017	1.56518E+16	8.16333E+15	{"type":"MultiPoint","coordinates":[]}
21	19	1.08.2017	1.48397E+16	9.1354E+15	{"type":"MultiPoint","coordinates":[]}
22	20	1.09.2017	1.36411E+16	6.47008E+15	{"type":"MultiPoint","coordinates":[]}
23	21	1.10.2017	1.75397E+14	5.1139E+16	{"type":"MultiPoint","coordinates":[]}
24	22	1.11.2017	2.2941E+15	4.43096E+14	{"type":"MultiPoint","coordinates":[]}
25	23	1.12.2017	2.71198E+16	4.01353E+15	{"type":"MultiPoint","coordinates":[]}
26	24	1.01.2018	3.09089E+15	3.61621E+15	{"type":"MultiPoint","coordinates":[]}

< > Turkey_GLDAS_Soil_And_Humidity_ Sayfa1 +

37	35	1.12.2018	3.04746E+16	4.16098E+15	{"type":"MultiPoint","coordinates":[]}
38	36	1.01.2019	3.60741E+15	3.56339E+16	{"type":"MultiPoint","coordinates":[]}
39	37	1.02.2019	3.34228E+15	3.63291E+16	{"type":"MultiPoint","coordinates":[]}
40	38	1.03.2019	3.13201E+16	4.04175E+15	{"type":"MultiPoint","coordinates":[]}
41	39	1.04.2019	3.0004E+16	5.10179E+15	{"type":"MultiPoint","coordinates":[]}
42	40	1.05.2019	2.49726E+16	6.61157E+15	{"type":"MultiPoint","coordinates":[]}
43	41	1.06.2019	2.20157E+15	9.08198E+14	{"type":"MultiPoint","coordinates":[]}
44	42	1.07.2019	1.76826E+16	8.23121E+15	{"type":"MultiPoint","coordinates":[]}
45	43	1.08.2019	1.47086E+16	8.47154E+15	{"type":"MultiPoint","coordinates":[]}
46	44	1.09.2019	1.56043E+16	6.90183E+15	{"type":"MultiPoint","coordinates":[]}
47	45	1.10.2019	1.73895E+16	6.28319E+15	{"type":"MultiPoint","coordinates":[]}
48	46	1.11.2019	2.01812E+15	4.42377E+15	{"type":"MultiPoint","coordinates":[]}
49	47	1.12.2019	2.73119E+16	4.14065E+15	{"type":"MultiPoint","coordinates":[]}
50	48	1.01.2020	3.17618E+15	3.19345E+15	{"type":"MultiPoint","coordinates":[]}
51	49	1.02.2020	3.37838E+15	3.67985E+15	{"type":"MultiPoint","coordinates":[]}
52	50	1.03.2020	3.19017E+15	4.74637E+15	{"type":"MultiPoint","coordinates":[]}
53	51	1.04.2020	2.85341E+14	5.14283E+14	{"type":"MultiPoint","coordinates":[]}
54	52	1.05.2020	2.57721E+14	6.46669E+15	{"type":"MultiPoint","coordinates":[]}
55	53	1.06.2020	2.22528E+16	7.8953E+15	{"type":"MultiPoint","coordinates":[]}
56	54	1.07.2020	1.62196E+16	8.98335E+14	{"type":"MultiPoint","coordinates":[]}
57	55	1.08.2020	1.46225E+14	7.58995E+15	{"type":"MultiPoint","coordinates":[]}
58	56	1.09.2020	1.44897E+16	7.54479E+15	{"type":"MultiPoint","coordinates":[]}
59	57	1.10.2020	1.71235E+15	5.70028E+15	{"type":"MultiPoint","coordinates":[]}
60	58	1.11.2020	2.11926E+14	4.43277E+15	{"type":"MultiPoint","coordinates":[]}
61	59	1.12.2020	2.42292E+14	4.23288E+15	{"type":"MultiPoint","coordinates":[]}
62					

< > Turkey_GLDAS_Soil_And_Humidity_ Sayfa1 +

(Figure 5.2 – Screenshot of the downloaded GLDAS CSV data)

6. MODIS MCD12Q1 – Global Land Cover Type (Annual, 500m Resolution)

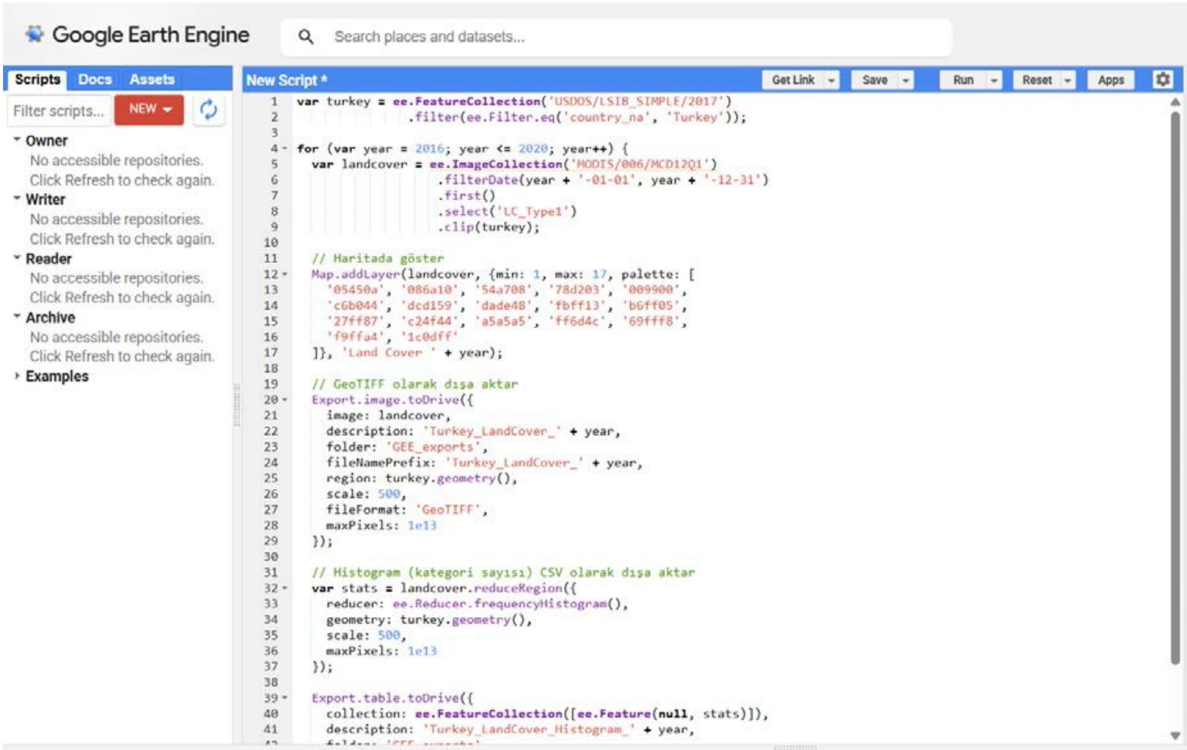
Land Use Data (MODIS MCD12Q1, 2016-2020)

The MCD12Q1 product of the MODIS satellite provides annual classifications of global land cover types. Each pixel is assigned to categories such as agriculture, forest, urban, or water bodies.

MODIS LC_Type1 Class Descriptions:

Code	Land Cover Type
0	No Data
1	Evergreen Needleleaf Forest
2	Evergreen Broadleaf Forest
3	Deciduous Needleleaf Forest
4	Deciduous Broadleaf Forest
5	Mixed Forests
6	Closed Shrublands
7	Open Shrublands
8	Woody Savannas
9	Savannas
10	Grasslands
11	Permanent Wetlands
12	Croplands
13	Urban and Built-Up
14	Cropland/Natural Vegetation Mosaic
15	Snow and Ice
16	Barren or Sparsely Vegetated
17	Water Bodies

Using MODIS MCD12Q1 data, land cover classifications across Turkey for 2016-2020 were extracted and exported in CSV format.



(Figure 6.1 – JavaScript code for downloading MODIS MCD12Q1 data)

- **MODIS MCD12Q1 verilerinin CSV çıktısı**

	A	B	C	D
1	system:index	dominant_land_cover	year	.geo
2	0	1.00033E+16	20160	{"type":"MultiPoint","coordinates":[]}
3	1	1.00044E+16	20170	{"type":"MultiPoint","coordinates":[]}
4	2	1.00047E+16	20180	{"type":"MultiPoint","coordinates":[]}
5	3	1.00034E+16	20190	{"type":"MultiPoint","coordinates":[]}
6	4	1.00049E+16	20200	{"type":"MultiPoint","coordinates":[]}
7				

(Figure 6.2 – Screenshot of the downloaded MODIS MCD12Q1 CSV data)

Implementation of the Linear Regression Model

After collecting all datasets, the following Python codes were used for data preparation and analysis with the linear regression machine learning model.

NDVI Regresyon Analizi.py ×

NDVI Regresyon Analizi.py > ...

```python
# Gerekli kütüphanelerin Eklenmesi
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.preprocessing import OneHotEncoder

# 1. Veri Setlerinin Yüklenmesi
df_ndvi = pd.read_csv('Turkey_Monthly_Mean_NDVI_Sentinel2_2016_2020.csv')
df_temp = pd.read_csv('Turkey_Monthly_Mean_Temperature_2016_2020.csv')
df_vv = pd.read_csv('Turkey_Monthly_Sentinel1_VV_2016_2020.csv')
df_precip = pd.read_csv('Turkey_Monthly_Mean_Precip_2016_2020.csv')
df_soil_humidity = pd.read_csv('Turkey_GLDAS_Soil_And_Humidity_2016_2020.csv')

# 2. Verilerin Birleştirilmesi
df_temp_vv = pd.merge(df_temp, df_vv, on=['system:index', 'date'])
df_temp_vv_precip = pd.merge(df_temp_vv, df_precip, on=['system:index', 'date'])
df_full = pd.merge(
    df_temp_vv_precip,
    df_soil_humidity[['system:index', 'date', 'mean_soil_moisture', 'mean_specific_humidity']],
    on=['system:index', 'date']
)
df_full = df_full.drop(columns=[col for col in df_full.columns if '.geo' in col], errors='ignore')
df = pd.merge(df_ndvi, df_full, on=['system:index', 'date'])
```


(Figures 7.1–7.3 – Linear regression analysis codes for NDVI)

NDVI Regresyon Analizi.py > ...

```python
df = df.drop(columns=[col for col in df.columns if '.geo' in col], errors='ignore')

# 3. Arazi Kullanım Verisinin Eklenmesi
df_landcover = pd.read_csv("Turkey_Yearly_LandCover_2016_2020.csv")
df['year'] = df['date'].str.slice(0, 4).astype(int)
df = df.merge(df_landcover[['year', 'dominant_land_cover']], on='year', how='left')

# 4. Kategorik Verinin Sayısal Dönüşümü
encoder = OneHotEncoder(sparse_output=False, handle_unknown='ignore')
landcover_encoded = encoder.fit_transform(df[['dominant_land_cover']])
landcover_encoded_df = pd.DataFrame(landcover_encoded, columns=encoder.get_feature_names_out(['dominan
df = pd.concat([df.reset_index(drop=True), landcover_encoded_df], axis=1)
df.drop(columns=['dominant_land_cover'], inplace=True)

# 5. Model Girdileri ve Hedef Değişken
X = df[['mean_temp_K', 'mean_VV_backscatter', 'mean_precip_mm',
        'mean_soil_moisture', 'mean_specific_humidity'] + list(landcover_encoded_df.columns)]
y = df['mean_NDVI']

# 6. Veri Setinin Eğitim ve Test Olarak Ayrılması
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# 7. Lineer Regresyon Modeli
model = LinearRegression()
model.fit(X_train, y_train)

```



```python
# 8. Tahmin ve Hata Hesaplaması
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

result = pd.DataFrame({
    'Gerçek NDVI': y_test,
    'Tahmin NDVI': y_pred
})

# 9. Hata Sütunlarının Eklenmesi
df['error'] = abs(df['mean_NDVI'] - model.predict(X))
df['squared_error'] = (df['mean_NDVI'] - model.predict(X))**2

# 10. Eksik Verilerin Temizlenmesi
print("Eksik Veri Sayısı:\n", df.isna().sum())
df = df.dropna()

# 11. Model Performans Sonuçları
print("-"*50)
print(f"Mean Squared Error: {mse:.6f}")
print(f"R² Score: {r2:.6f}")
print("-"*50)
print("Gerçek ve Tahmin Edilen NDVI Değerleri:")
print(result.head())

```


(Figures 7.1–7.3 – Linear regression analysis codes for NDVI)

```python
# 12. Model Katsayıları  Katsayıları
coef_df = pd.DataFrame({
    'Değişken': X.columns,
    'Katsayı': model.coef_
}).sort_values(by='Katsayı', key=abs, ascending=False)
print(coef_df)

# 13. Korelasyon Matrisi
plt.figure(figsize=(10, 6))
sns.heatmap(df.corr(numeric_only=True), annot=True, cmap='coolwarm')
plt.title('Korelasyon Matrisi')
plt.tight_layout()
plt.show()
# 14. Zaman Serisi Grafiği
df['tarih'] = pd.to_datetime(df['date'])
plt.figure(figsize=(12, 5))
sns.lineplot(data=df, x='tarih', y='mean_NDVI', label='Gerçek NDVI')
sns.lineplot(data=df, x='tarih', y=model.predict(X), label='Tahmin NDVI')
plt.title('NDVI Zaman Serisi')
plt.xlabel('Tarih')
plt.ylabel('NDVI')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```



```python
# 15. Gerçek vs Tahmin NDVI Dağılım Grafiği
plt.figure(figsize=(6, 6))
plt.scatter(df['mean_NDVI'], model.predict(X), alpha=0.5, label='Veri Noktaları')
plt.plot([df['mean_NDVI'].min(), df['mean_NDVI'].max()],
         [df['mean_NDVI'].min(), df['mean_NDVI'].max()],
         color='red', linestyle='--', label='y = x')
plt.xlabel('Gerçek NDVI')
plt.ylabel('Tahmin NDVI')
plt.title('Gerçek vs Tahmin NDVI')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

(Figures 7.1–7.3 – Linear regression analysis codes for NDVI)

Results of the Linear Regression Model

- **The outputs of the model are presented below.**

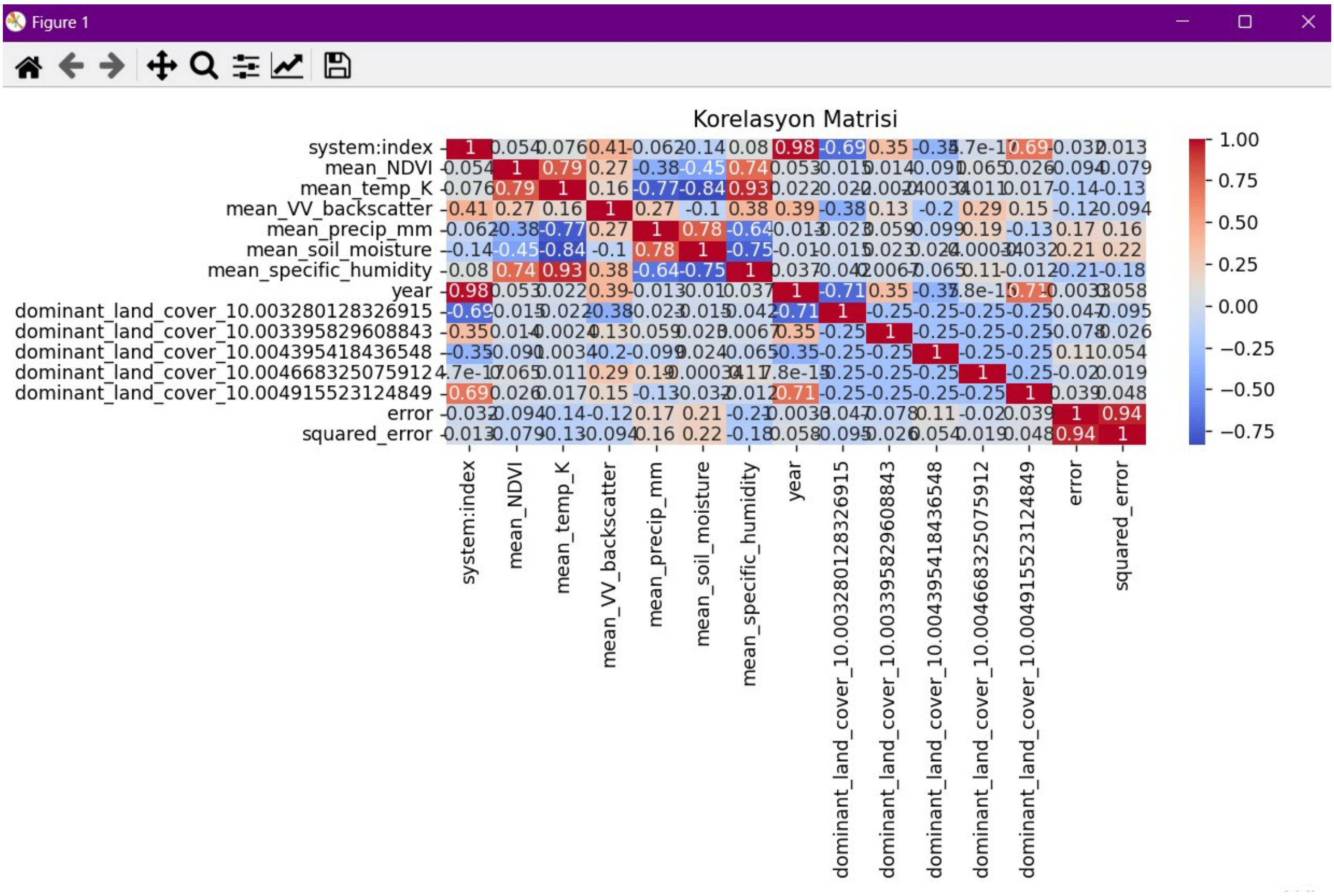
```
TERMINAL

Eksik Veri Sayısı:

mean_temp_K                              0
mean_VV_backscatter                      0
mean_precip_mm                           0
mean_soil_moisture                       0
mean_specific_humidity                   0
dominant_land_cover_10.003280128326915   0
dominant_land_cover_10.003395829608843   0
dominant_land_cover_10.004395418436548   0
dominant_land_cover_10.004668325075912   0
dominant_land_cover_10.004915523124849   0
mean_NDVI                                0
--------------------------------------------------
Mean Squared Error: 0.001281
R² Score: 0.755325
--------------------------------------------------
Gerçek ve Tahmin Edilen NDVI Değerleri:
    Gerçek NDVI  Tahmin NDVI
0      0.178865     0.148189
5      0.330533     0.314218
36     0.118222     0.195774
45     0.234713     0.226960
13     0.107672     0.123987
--------------------------------------------------
                                 Değişken    Katsayı
4                  mean_specific_humidity -15.075654
2                          mean_precip_mm   0.028117
1                     mean_VV_backscatter   0.016326
9  dominant_land_cover_10.004915523124849   0.011768
7  dominant_land_cover_10.004395418436548  -0.009695
3                      mean_soil_moisture   0.006549
6  dominant_land_cover_10.003395829608843  -0.002666
5  dominant_land_cover_10.003280128326915   0.001186
8  dominant_land_cover_10.004668325075912  -0.000592
0                             mean_temp_K   0.000225
```

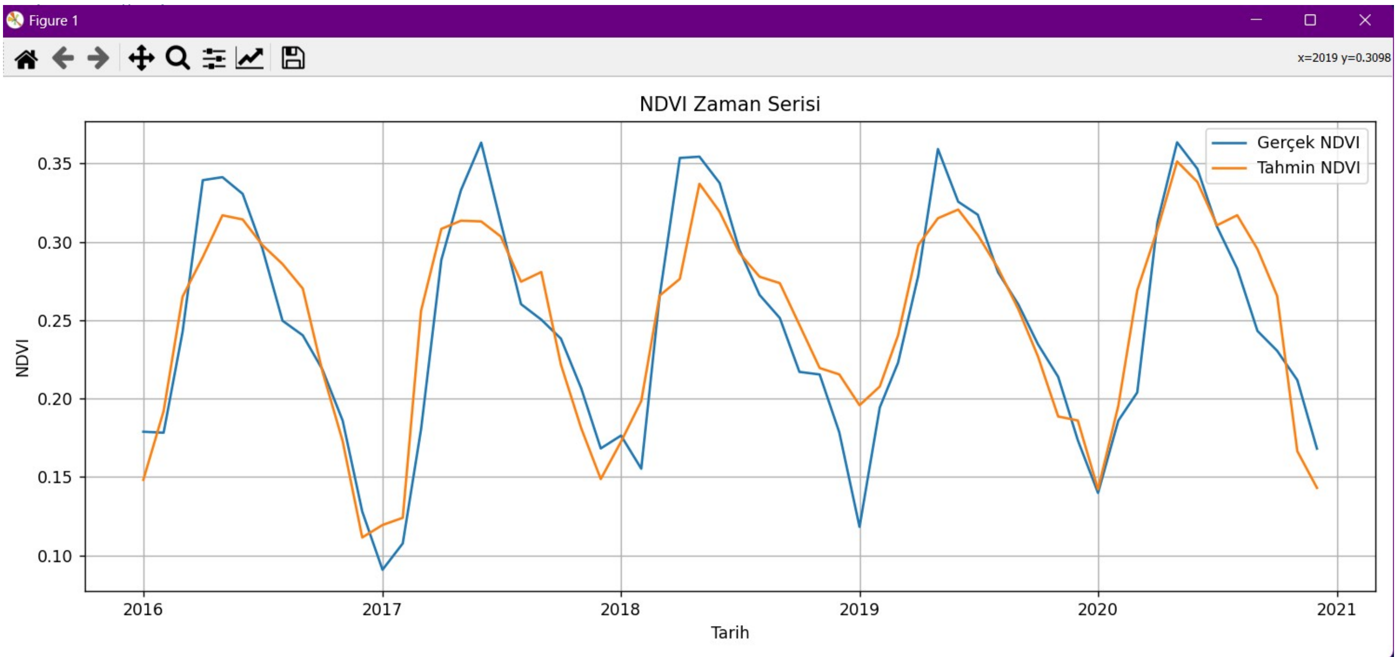
(Figure 8.1 – Output of the NDVI regression analysis)

Correlation Matrix



(Figure 8.2 – Correlation matrix of the NDVI regression analysis)

Actual and Predicted NDVI



(Figures 8.3 & 8.4 – Graphs of actual and predicted NDVI values from the regression analysis)

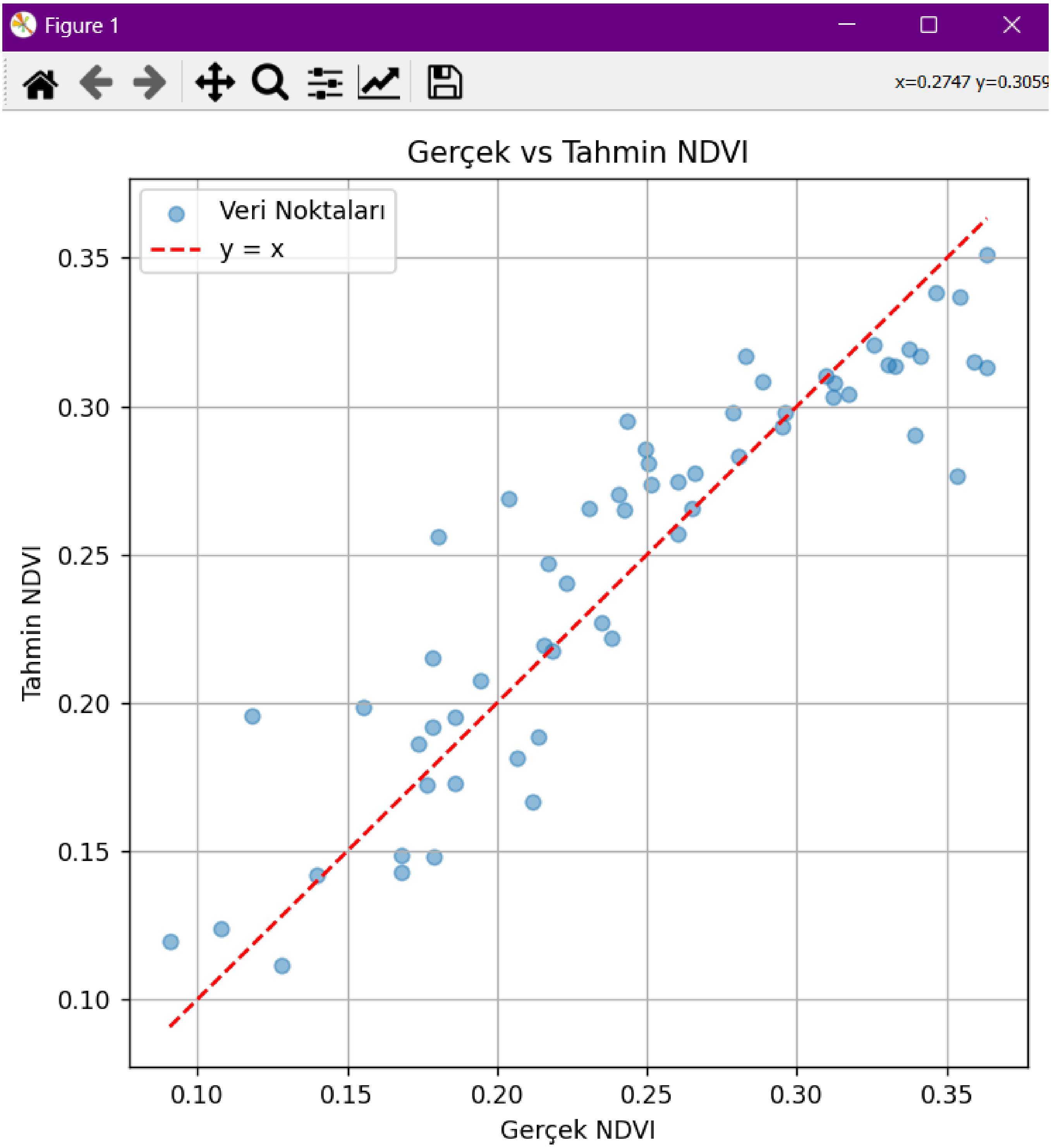


Figure 1
x=0.2747 y=0.3059
Gerçek vs Tahmin NDVI
Veri Noktaları
y = x
0.35
0.30
0.25
0.20
0.15
0.10
Tahmin NDVI
0.10
0.15
0.20
0.25
0.30
0.35
Gerçek NDVI

Project Results and Future Work

The linear regression-based machine learning model developed in this project demonstrated satisfactory performance in predicting NDVI (Normalized Difference Vegetation Index) values. The overall performance of the model was evaluated using the coefficient of determination (R^2) and mean squared error (MSE), and the results indicate that the model successfully captured the fundamental relationships between climate variables, land cover, and vegetation health.
Several strategies can be considered to further enhance the model's performance. Variables showing high correlation in the correlation matrix (e.g., mean_temperature and mean_humidity) could be more effectively utilized, while variables with low correlation might be excluded to improve model accuracy.

In future studies, extending the analysis period to cover a longer timeframe would allow for a more comprehensive assessment of long-term climate change impacts. Additionally, exploring non-linear models could better capture complex relationships. Techniques such as PolynomialFeatures, Ridge, or Lasso regularization could be applied to assess the model's generalization capabilities.

Moreover, developing separate models for different crop types or regions could further improve prediction accuracy. For this purpose, high-quality, crop-specific datasets are critical, and institutional databases such as the Turkish Ministry of Agriculture and Forestry's “Plant Protection and Products Database” could serve as valuable resources.

Overall, this study demonstrates the effective integration of **remote sensing, satellite imagery, and machine learning methods** to analyze the relationship between climate data and vegetation health. By combining multiple satellite-derived datasets and supporting analyses with visualizations, the project provides a foundation for decision support systems in sustainable agriculture. Future applications of similar methodologies over longer time periods and in different regions could further test and validate the model's applicability.