

# GeoAI Kullanım Dokumanı

Kurulum, Konfigurasyon, Çalıştırma ve Saha Operasyonu (Genisletilmiş L<sup>A</sup>T<sub>E</sub>X Surumu)

GeoAI

19 Subat 2026

## Özet

Bu dokuman, GeoAI boru hattını (pipeline) kurulumdan üretim kosularına kadar uçtan uca anlatır. Demo (sentetik veri) ile hızlı doğrulama, gerçek veri ile proje çalıştırma (`run_project.py`), ve programatik API (`GeoAIPipeline`) kullanım senaryoları kapsanır. Ayrıca çıktı dosyalarının yorumu, hata giderme, operasyon öncesi kontrol listesi ve parametre seçim rehberi sunulur.

## İçindekiler

<b>1</b>	<b>Hızlı Başlangıç</b>	<b>3</b>
1.1	En kısa yol: demo . . . . .	3
1.2	Gerçek veri ile koku . . . . .	3
1.3	Programatik API . . . . .	3
<b>2</b>	<b>Kurulum</b>	<b>3</b>
2.1	Gereksinimler . . . . .	3
2.2	Proje yapısı . . . . .	4
<b>3</b>	<b>Çalışma Mantığı: Girdi–Çıktı ve Skorlar</b>	<b>4</b>
<b>4</b>	<b>Demo Modu</b>	<b>4</b>
4.1	Katman yoksa otomatik demo fallback . . . . .	4
<b>5</b>	<b>Gerçek Veri ile Çalışma: <code>run_project.py</code></b>	<b>4</b>
5.1	Temel konfigürasyon değişkenleri . . . . .	4
5.2	Örnek akış . . . . .	5
<b>6</b>	<b>Veri Hazırlama Kuralları</b>	<b>5</b>
6.1	Grid katmanları . . . . .	5
6.2	Kuyu CSV formatı . . . . .	5
6.3	CRS/extent uyumu için minimum kontrol . . . . .	5
<b>7</b>	<b>API Referansı: <code>GeoAIPipeline</code></b>	<b>6</b>
<b>8</b>	<b>Çıktıları Yorumlama</b>	<b>6</b>
8.1	Haritalar . . . . .	6
8.2	Rapor metrikleri . . . . .	6
<b>9</b>	<b>Saha Operasyonu için Önerilen İş Akışı</b>	<b>6</b>
<b>10</b>	<b>Hata Giderme</b>	<b>7</b>

11 Uretim Ortami Onerileri	7
12 Komut Ozeti	7
A Ayrintili Konfigurasyon Rehberi	7
B Cikti Dosya Sozlugu	8
C Operasyon Oncesi Kontrol Listesi	8
D Sik Sorulan Sorular	8

# 1 Hizli Baslangic

## 1.1 En kısa yol: demo

Kurulumun dogru calistigini dogrulamak icin demo kosusu onerilir:

```
python demo.py
```

Demo, sentetik grid katmanlari ve sentetik kuyu etiketleri uretir ve ciktilari varsayilan olarak output/ altina yazar.

## 1.2 Gercek veri ile kosu

Gercek veriyle calistirmek icin:

```
python run_project.py
```

run\_project.py icinde KATMANLAR bos ise sistem (ayar aciksa) otomatik olarak demo fallback calistirabilir (bkz. Bolum 4.1).

## 1.3 Programatik API

Python icinden:

```
from geoai.pipeline import GeoAIPipeline

pipe = GeoAIPipeline(project_name="Saha_A")
pipe.add_layer("data/magnetic.grd", name="magnetic")
pipe.add_layer("data/gravity.grd", name="gravity")
pipe.add_wells("data/wells_mineral.csv", target_type="mineral")
results = pipe.run(target_types=["mineral"], target_nx=300, target_ny=300)
```

# 2 Kurulum

## 2.1 Gereksinimler

GeoAI, Python tabanlidir. Tipik bagimliliklar:

Paket	Amac
numpy, scipy, pandas	Numerik hesaplama ve veri islemleri
scikit-learn	Modelleme, CV ve metrikler
matplotlib	Harita/cikti gorselleri
rasterio (onerilir)	GeoTIFF koordinatli okuma
Pillow	TIFF fallback okuma

Kurulum:

```
python -m pip install -r requirements.txt
# veya hizli kurulum:
python setup_geoai.py
```

## 2.2 Proje yapisi

Tipik dizin agaci:

```
geoai/  
  pipeline.py  
  io/loaders.py  
  core/preprocessor.py  
  models/prospectivity.py  
  viz/maps.py  
  utils/reporting.py  
demo.py  
run_project.py
```

## 3 Calisma Mantigi: Girdi–Cikti ve Skorlar

GeoAI, her piksel konumu  $\mathbf{s} = (x, y)$  icin bir ozellik vektörü  $\mathbf{x}(\mathbf{s}) \in \mathbb{R}^{n_f}$  olusturur. Denetimli senaryoda hedef olasiligi:

$$P(\mathbf{s}) \approx \mathbb{P}(y(\mathbf{s}) = 1 \mid \mathbf{x}(\mathbf{s})).$$

Model cesitliliinden turetilen belirsizlik (or. ensemble dagilimi):

$$U(\mathbf{s}) = \sqrt{\sum_{m=1}^M w_m (\hat{p}_m(\mathbf{s}) - P(\mathbf{s}))^2}, \quad \sum_{m=1}^M w_m = 1.$$

Operasyonel siralama icin risk-duzeltilmis skor:

$$S(\mathbf{s}) = \text{clip}(P(\mathbf{s}) - \lambda U(\mathbf{s}), 0, 1),$$

burada  $\lambda$  (or. 0.3) belirsizlik cezasini ayarlar.

## 4 Demo Modu

Demo, kurulum dogrulaması ve uctan uca test icin tasarlanmistir.

```
python demo.py
```

### 4.1 Katman yoksa otomatik demo fallback

`run_project.py` icinde katman listesi bos ise, belirli bir bayrak ile sistem demo fallback calistirabilir. Bu davranisin amaci, “bos konfig” durumunda bile boru hattinin smoke-test yapılabilmesidir.

## 5 Gercek Veri ile Calisma: `run_project.py`

### 5.1 Temel konfigurasyon degiskenleri

Degisken	Anlam	Tipik deger
PROJE_ADI	Rapor/cikti adlarına yansir	Saha_A_2026
CIKTI_KLASORU	PNG/CSV/TXT cikis dizini	output/
HEDEF_TIPLERI	Problem tipi	['mineral']
KATMANLAR	Katman adi $\rightarrow$ dosya yolu	{...}
KUYU_DOSYALARI	Hedef tipi $\rightarrow$ kuyu CSV	{...}
GRID_NX, GRID_NY	Referans grid cozunurlugu	300, 300

## 5.2 Ornek akis

```
from geoai.pipeline import GeoAIPipeline

pipe = GeoAIPipeline(project_name="Saha_A")
pipe.add_layer("data/magnetic.grd", name="magnetic")
pipe.add_layer("data/gravity.grd", name="gravity")
pipe.add_layer("data/resistivity.tif", name="resistivity")

pipe.add_wells("data/wells_mineral.csv", target_type="mineral")

results = pipe.run(
    target_types=["mineral"],
    target_nx=300,
    target_ny=300,
    n_targets=15,
    min_prob=0.35,
    cv_folds=5,
    output_dir="output/",
    show_plots=False,
    save_plots=True,
)
```

## 6 Veri Hazirlama Kurallari

### 6.1 Grid katmanlari

Pratik kurallar:

- Tum katmanlari mumkunse ayni CRS'e (koordinat referans sistemi) getir.
- Extent (kapsam) benzer olmalı; asiri farkli extent co-registration sirasinda bilgi kaybi veya NaN artisi dogurur.
- GeoTIFF icin **rasterio** kullanimi onerilir (koordinat sadakati).
- CSV/XYZ verilerinde X, Y, Z kolonlari acik ve sayisal olmalidir.
- Cok yuksek NaN orani, hem feature kalitesini hem de egitim ornek sayisini dusurur.

### 6.2 Kuyu CSV formati

Beklenen minimal kolonlar: X, Y, LABEL. Istege bagli: TARGET\_TYPE, DEPTH, NOTES.

```
X,Y,LABEL,TARGET_TYPE,DEPTH,NOTES
312000,4012000,1,mineral,150,cevherli zon
320000,4018000,0,mineral,,steril
```

LABEL yalnızca {0,1} olmalıdır. Gecersiz degerlerde yukleme asamasinda hata uretilmesi beklenir.

### 6.3 CRS/extent uyumu icin minimum kontrol

GIS tarafinda hizli kontrol:

1. Katmanlarin CRS'ini dogrula (EPSG kodu).
2. Extent'leri overlay ederek kuyu noktalarinin katmanlarin icine dustugunu kontrol et.
3. Buyuk kaymalar varsa once reprojection/warp uygula.

## 7 API Referansi: GeoAIPipeline

Metod	Ne yapar	Not
<code>add_layer(path, ...)</code>	Diskten katman ekler	Format otomatik algılanır
<code>add_layer_array(grid, x, y, name)</code>	Numpy ile katman ekler	Test/demo için hızlı
<code>add_wells(path, target_type)</code>	Kuyu CSV yukler	Kolonlar otomatik bulunabilir
<code>add_wells_dataframe(df, target_type)</code>	DataFrame ile kuyu ekler	<code>x,y,label</code> zorunlu
<code>run(...)</code>	Tam boru hattı	Sonuçları <code>dict</code> dondurur
<code>run_demo(...)</code>	Sentetik veriyle demo	Smoke test

## 8 Çıktıları Yorumlama

### 8.1 Haritalar

- **Prospectivity haritası:** sıcak renkler yüksek  $P(s)$  değerini temsil eder.
- **Belirsizlik haritası:** yüksek  $U(s)$  karar kalitesinin düşük olabileceğini gösterir.
- **Güven skoru:**  $S(s) = P(s) - \lambda U(s)$  ile daha risk-averse bir sıralama sağlar.

### 8.2 Rapor metrikleri

- **ROC-AUC:** 0.5 rastgele, 1.0 mükemmel ayırma.
- **PR-AUC:** pozitif sınıf az olduğunda daha anlamlı.
- **Brier skoru:** daha düşük değer daha iyi olasılık kalibrasyonu.
- **Optimal threshold:** sınıf karar eşikini otomatik optimize eder (örn. F1 maksimize).

## 9 Saha Operasyonu için Önerilen İş Akışı

Önerilen pratik sıra:

1. Tüm jeofizik katmanları tek CRS'e getir.
2. Kuyu etiketlerini (0/1) saha logları ile doğrula.
3. Küçük grid ile hızlı deneme yap (örn.  $200 \times 200$ ).
4. Metrikleri incele; gerekiyorsa katman/feature setini revize et.
5. Nihai kosuyu daha yüksek çözünürlükte al (örn.  $500 \times 500$ ).
6. Top hedef listesini jeoloji ekibi ile birleştir ve sondaj planı çıkar.

## 10 Hata Giderme

Belirti	Muhtemel sebep	Cozum
KATMANLAR bos hatasi	Konfig bos	Demo fallback acik birak veya katman yolunu gir
Hedef cikmiyor	Kuyu yok / az pozitif	Kuyu setini artır; gecici unsupervised ciktiyi kullan
GeoTIFF koordinati bozuk	rasterio yok / CRS sorunlu	rasterio kur; katmani GIS ile kontrol et
CV grafikleri bos	Metrik anahtar uyumsuzlugu	Guncel isimleri kontrol et (auc_roc/auc_pr)
Kuyular kenara yigiliyor	CRS/extent uyumsuz	Reprojection/warp uygula; extent'i eslestir

## 11 Uretim Ortami Onerileri

- Kosulari **timestamp** ve **run-id** ile arsilve.
- Veri seti ve model konfigurasyonunu birlikte versiyonla (orn. Git + cikti manifest).
- Buyuk gridler icin batch boyutlarini ve CPU/RAM kullanimini izle.
- Cikti CSV/TXT dosyalarini QA kontrolunden gecirerek sahaya aktar.

## 12 Komut Ozeti

```
# Demo
python demo.py

# Gercek veri (KATMANLAR bossa otomatik demo fallback olabilir)
python run_project.py

# Regresyon testleri (varsa)
python -m unittest tests.test_regressions -v
```

## A Ayrıntili Konfigurasyon Rehberi

Aynı kod tabanı ile farklı hedef tipleri için ayrı konfigürasyon kullanılabilir. Başlangıç değerleri:

Senaryo	HEDEF_TIPLERI	Grid	min_prob
Maden arama	['mineral']	400×400	0.35
Yeraltı suyu	['groundwater']	300×300	0.30
Coklu hedef	['mineral','groundwater']	350×350	(opsiyonel)

## B Cikti Dosya Sozlugu

Dosya	Icerik	Tipik kullanim
input_layers.png	Tum giris katmanlari	Hizalama/NaN kontrolu
prospectivity_<tip>.png	$P$ , $U$ ve skor panelleri	Hedef bolge secimi
report_<tip>.png	Hedef siralama + metrikler	Toplantı sunumu/QA
multi_target_comparison.png	Tipler arasi karsilastirma	Kaynak dagitimi
*_targets_*.csv	Koordinatli hedef listesi	GIS/CAD aktarimi
*_report_*.txt	Metrik ve hedef ozet metni	Arsiv/izlenebilirlik

## C Operasyon Oncesi Kontrol Listesi

- run\_project.py icindeki dosya yollari dogru mu?
- Kuyu CSV'sinde x,y,label kolonlari dogru ve 0/1 etiketli mi?
- output/ dizininde yazma izni var mi?
- Grid cozunurlugu ( $nx$ ,  $ny$ ) donanim kapasitesine uygun mu?
- Son kosu metrikleri (PR-AUC, Brier) kabul esiginde mi?

## D Sik Sorulan Sorular

Soru	Kisa cevap
Kuyu verisi yoksa ne olur?	Pipeline, unsupervised (anomaly tabanlı) fallback ile harita uretir.
Neden PR-AUC onemli?	Pozitifler az oldugunda ROC-AUC tek basina yaniltici olabilir.
Neden hedef sayisi sinirli?	Bagli bileşen + minimum mesafe filtreleri tekrarları azaltir.
Koordinatlar neden kaymis?	CRS/extent tutarsizligi co-registration kalitesini bozar.