

51

**GEODETSKI DAN**

BRDO PRI KRANJU

15. IN 16. NOVEMBER

2023



# UMETNA INTELIGENCA V GIS

Alen Mangafič

Alen.Mangafic@gis.si

VEĀRAŠEŽNO MODELIRANJE PROSTORA



# Vsebina

- Osnovni koncepti
- GIS & UI
- Pasti pri uporabi UI (v GIS)
- Sklep



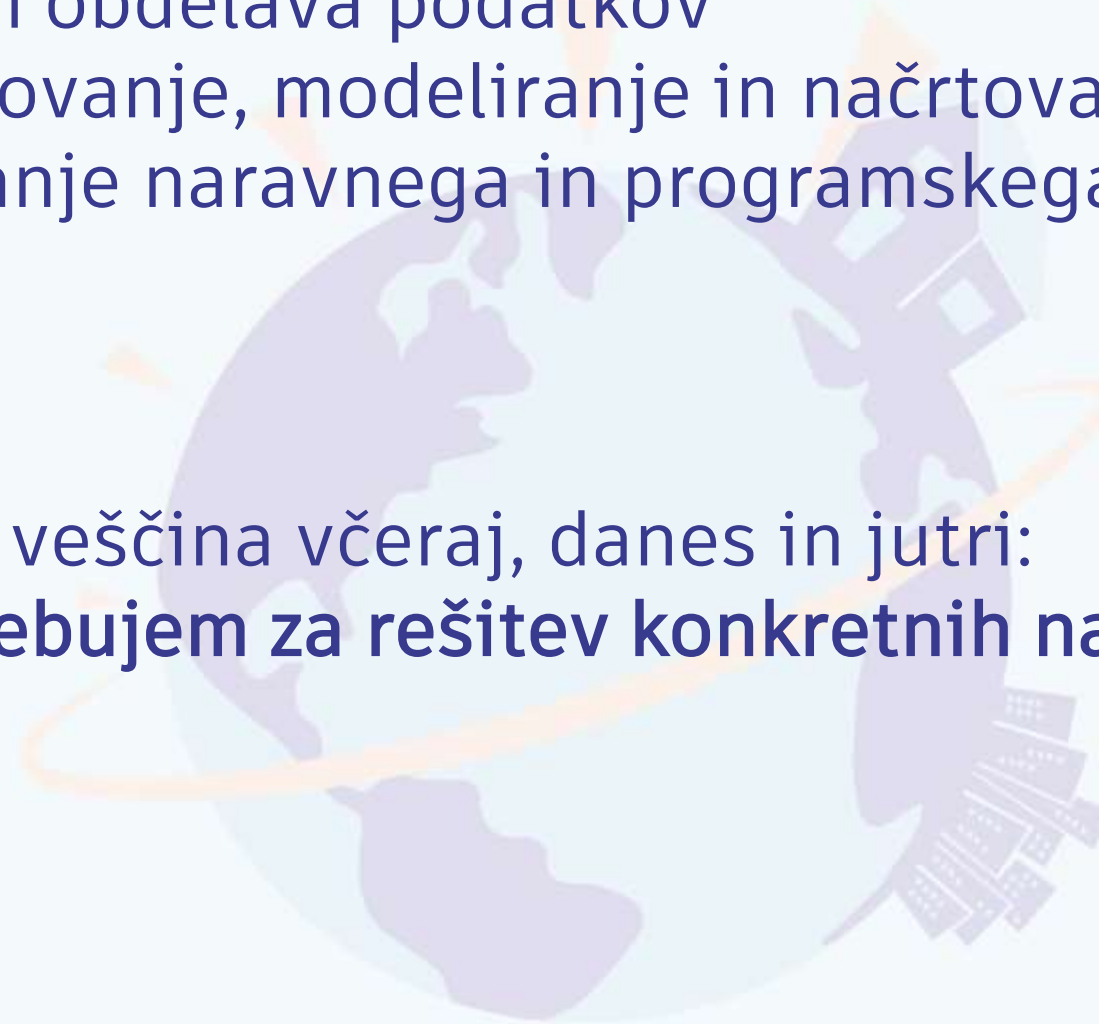
# Osnovni koncepti - definicija

- UI je interdisciplinarna veda, ki proučuje naprave, sposobne posnemati človekovo razmišljanje in vključuje npr. sklepanje, planiranje, učenje, znanje, komuniciranje, percepcijo (Terminološki slovar avtomatike, 2018)
- UI je zmožnost stroja, da izkazuje človeške lastnosti, kot so mišljenje, učenje, načrtovanje in kreativnost (Evropska komisija, 2021)
- Sinteza: mišljenje, učenje, percepcija, načrtovanje, komuniciranje, kreativnost



# GIS integracija

- Analiza in obdelava podatkov
- Napovedovanje, modeliranje in načrtovanje
- Procesiranje naravnega in programskega jezika
  
- Osnovna veščina včeraj, danes in jutri:  
**Kaj potrebujem za rešitev konkretnih nalog?**



# Analiza in obdelava podatkov – računalniški vid

- Interpretacija
  - Kvalitativna interpretacija (npr. semantična segmentacija stavb po tipologiji)
  - Kvantitativna interpretacija (npr. ocena suše v vinogradu)
- Detekcija objektov
  - Štetje objektov (npr. segmentacija primerkov stavb)
- Zaznavanje sprememb
  - Zaznavanje krčenja gozdov



# Analiza in obdelava podatkov – kontrola podatkov

- Samodejna validacija in čiščenje podatkov
  - Iskanje manjkajočih ali napačnih vrednosti (npr. napačna imena cest, napačna koordinata)
- Kontrola in urejanje posnetkov
  - Zaznavanje artefaktov in težav (npr. oblačnost)
  - Odstranjevanje artefaktov (npr. odstranjevanje meglic)
  - Interpolacija in združevanje podatkov (npr. sintetična generacija podatkov, združevanje podatkov iz različnih senzorjev)



# Napovedovanje, modeliranje in načrtovanje

- Aplikacije v naravi in okolju
  - Napoved tveganja naravnih nesreč
  - Napoved širjenja požarov
  - Napovedi širjenja epidemij
  - Modeliranje kakovosti vode ...
- Aplikacije v urbanem okolju
  - Prometne napovedi
  - Načrtovanje sosesk
  - Povpraševanje po električni energiji
  - Napoved cen nepremičnin ...
- Energetika in energetski viri
  - Razporejanje energetskih virov
  - Napoved vetrne in sončne energije
  - ...



# Procesiranje naravnega in programskega jezika

- Geokodiranje dokumentov
  - Pripis prostorske komponente besedilom
  - Pridobivanje prostorskih podatkov (npr. QChatGPT)
- Kontekstualna analiza in sinteza
  - Pisanje poročil in opis dokumentov na osnovi konteksta (npr. ocena kakovosti, ocena ustreznosti)
- Podpora pri programiranju
  - Optimizacija kode (npr. pospeševanje algoritmov)
  - Posodobitev in prevod kode (npr. posodobitev skript iz Python 2 v Python 3)
  - Avtomatizirana kartografija (npr. izdelava spletne karte na osnovi pisnega opisa)





# Primer – spletna karta

- Izdelava spletne karte v 60 s
- Recept: ChatGPT in Python



Default (GPT-3.5)



Ciao! Lepo te prosim, da mi poveš seznam največjih mest v Sloveniji in število njihovih prebivalcev



Seveda! Tukaj je seznam nekaterih največjih mest v Sloveniji in njihovo približno število prebivalcev na podlagi podatkov, ki so bili na voljo do leta 2021:



1. Ljubljana - približno 280.000 prebivalcev
2. Maribor - približno 110.000 prebivalcev
3. Celje - približno 38.000 prebivalcev
4. Kranj - približno 38.000 prebivalcev
5. Velenje - približno 25.000 prebivalcev
6. Novo mesto - približno 23.000 prebivalcev
7. Koper - približno 25.000 prebivalcev
8. Nova Gorica - približno 13.000 prebivalcev
9. Ptuj - približno 23.000 prebivalcev
10. Domžale - približno 12.000 prebivalcev

Omenjeni podatki so približni in se lahko od takrat spremenili. Za najnovejše informacije o prebivalstvu mest v Sloveniji svetujem, da preverite uradne vire ali statistične podatke.



Preuredi ta odgovor v tabelo



Seveda, tukaj je seznam največjih mest v Sloveniji z njihovim približnim številom prebivalcev v obliki tabele:



Mesto	Število prebivalcev (približno)
Ljubljana	280.000
Maribor	110.000
Celje	38.000
Kranj	38.000
Velenje	25.000
Novo mesto	23.000
Koper	25.000
Nova Gorica	13.000
Ptuj	23.000
Domžale	12.000

Omenjeni podatki so približni in se lahko od takrat spremenili. Za najnovejše informacije o prebivalstvu mest v Sloveniji svetujem, da preverite uradne vire ali statistične podatke.



Dodaj mi še geografske koordinate v stolpec tabele (WGS84)



Seveda, tukaj je posodobljena tabela z dodanim stolpcem za geografske koordinate v WGS84 za vsako mesto:



Mesto	Število prebivalcev (približno)	Geografske koordinate (WGS84)
Ljubljana	280.000	46.0569° N, 14.5058° E
Maribor	110.000	46.5547° N, 15.6466° E
Celje	38.000	46.2344° N, 15.2606° E
Kranj	38.000	46.2383° N, 14.3559° E
Velenje	25.000	46.3581° N, 15.1100° E
Novo mesto	23.000	45.8029° N, 15.1684° E
Koper	25.000	45.5469° N, 13.7302° E
Nova Gorica	13.000	45.9614° N, 13.6432° E
Ptuj	23.000	46.4196° N, 15.8700° E
Domžale	12.000	46.1389° N, 14.5933° E

Geografske koordinate so podane v obliki stopinj (°) za zemljepisno širino (N za severno) in zemljepisno dolžino (E za vzhodno) ter so približne vrednosti. Za natančnejše geografske koordinate lahko uporabite geografske storitve ali uradne vire.



Kartiraj lokacije teh mest v Pythonu z uporabo knjižnic Geopandas in Leaflet. Prilagodi kodo za interaktivno odpiranje v Jupyter notebooku. Omogoči možnost povečave in prikaži mesta v modri barvi. Dodaj tudi vrstico za izvoz prikaza kot spletne karte z uporabo knjižnice Folium. Na karti prikaži imena mest in v oklepaju dodaj naključna imena glasbenikov iz 60-ih let, pri čemer se imena ne smejo ponavljati



Da ustvarimo interaktivno karto mest v Pythonu s pomočjo knjižnic Geopandas, Leaflet in Folium, morate najprej namestiti te knjižnice. Upoštevajte, da za delo z Geopandas potrebujete tudi oblikovne datoteke, ki vsebujejo podatke o mejah mest. Tukaj je primer, kako to storiti:



1. Namestite knjižnice:

bash

Copy code

```
pip install geopandas folium
```

1. Naložite oblikovne datoteke za slovenske meje. Te datoteke morate dobiti iz uradnih virov ali drugih zanesljivih virov:

python

Copy code

```
import geopandas as gpd
```

```
# Naložite oblikovne datoteke za slovenske meje
```

```
shapefile_path = 'path_to_shapefile.shp' # Nadomestite z dejansko p
```

```
mesta.gdf = gpd.read_file(shapefile_path)
```

Regenerate



1. Naložite oblikovne datoteke za slovenske meje. Te datoteke morate dobiti iz uradnih virov ali drugih zanesljivih virov:

python

Copy code

```
import geopandas as gpd

# Naložite oblikovne datoteke za slovenske meje
shapefile_path = 'path_to_shapefile.shp' # Nadomestite z dejansko potjo do
mesta_gdf = gpd.read_file(shapefile_path)
```

1. Dodajte geografske koordinate mest v dataframe in naključna imena glasbenikov:

python

Copy code

```
import random

# Dodajte geografske koordinate mest
mesta_gdf['latitude'] = [46.0569, 46.5547, 46.2344, 46.2383, 46.3581, 45.802]
mesta_gdf['longitude'] = [14.5058, 15.6466, 15.2606, 14.3559, 15.1100, 15.16]

# Naključna imena glasbenikov
glasbeniki = ["Elvis Presley", "The Beatles", "Bob Dylan", "The Rolling Stones"]
mesta_gdf['glasbenik'] = [random.choice(glasbeniki) for _ in range(len(mesta_gdf))]
```



1. Uporabite knjižnico Folium za prikaz interaktivne karte z označenimi mesti:

```
python Copy code  
  
import folium  
  
# Središče zemljevida  
center_lat, center_lon = 46.15, 14.995 # Približna središčna točka Slovenij  
  
# Ustvarite zemljevid z uporabo knjižnice Folium  
mesta_map = folium.Map(location=[center_lat, center_lon], zoom_start=7)  
  
# Dodajte markerje za vsako mesto  
for idx, row in mesta_gdf.iterrows():  
    folium.Marker(  
        location=[row['latitude'], row['longitude']],  
        popup=f"{row['Mesto']} ({row['glasbenik']})",  
        icon=folium.Icon(color='blue')  
    ).add_to(mesta_map)  
  
# Prikažite zemljevid v Jupyter notebooku  
mesta_map
```

1. Za izvoz prikaza kot spletne karte uporabite funkcijo `save` iz knjižnice Folium:

```
python Copy code  
  
mesta_map.save('slovenian_cities_map.html')
```



```
In [2]: import geopandas as gpd
import folium
import random

# Ustvari geodataski okvir z mestnimi koordinatami
mesta = {
    'Mesto': ['Ljubljana', 'Maribor', 'Celje', 'Kranj', 'Velenje', 'Novo mesto', 'Koper', 'Ptuj'],
    'Število prebivalcev (2021)': [282000, 110000, 37000, 38000, 24000, 36000, 25000, 24000],
    'Latitude': [46.0569, 46.5547, 46.2308, 46.2382, 46.3592, 45.8020, 45.5469, 46.4201],
    'Longitude': [14.5058, 15.6466, 15.2647, 14.3559, 15.1182, 15.1674, 13.7294, 15.8709]
}

# Pretvori podatke v GeoDataFrame
gdf = gpd.GeoDataFrame(mesta, geometry=gpd.points_from_xy(mesta['Longitude'], mesta['Latitude']))

# Naključna imena pevcev iz 60-ih let
pevci_60 = ['Elvis Presley', 'The Beatles', 'Frank Sinatra', 'Aretha Franklin', 'The Rolling Stones', 'Jim Morrison',

# Naključno izberite imena pevcev brez ponavljanja
nakljucna_imena_pevcev = random.sample(pevci_60, len(gdf))

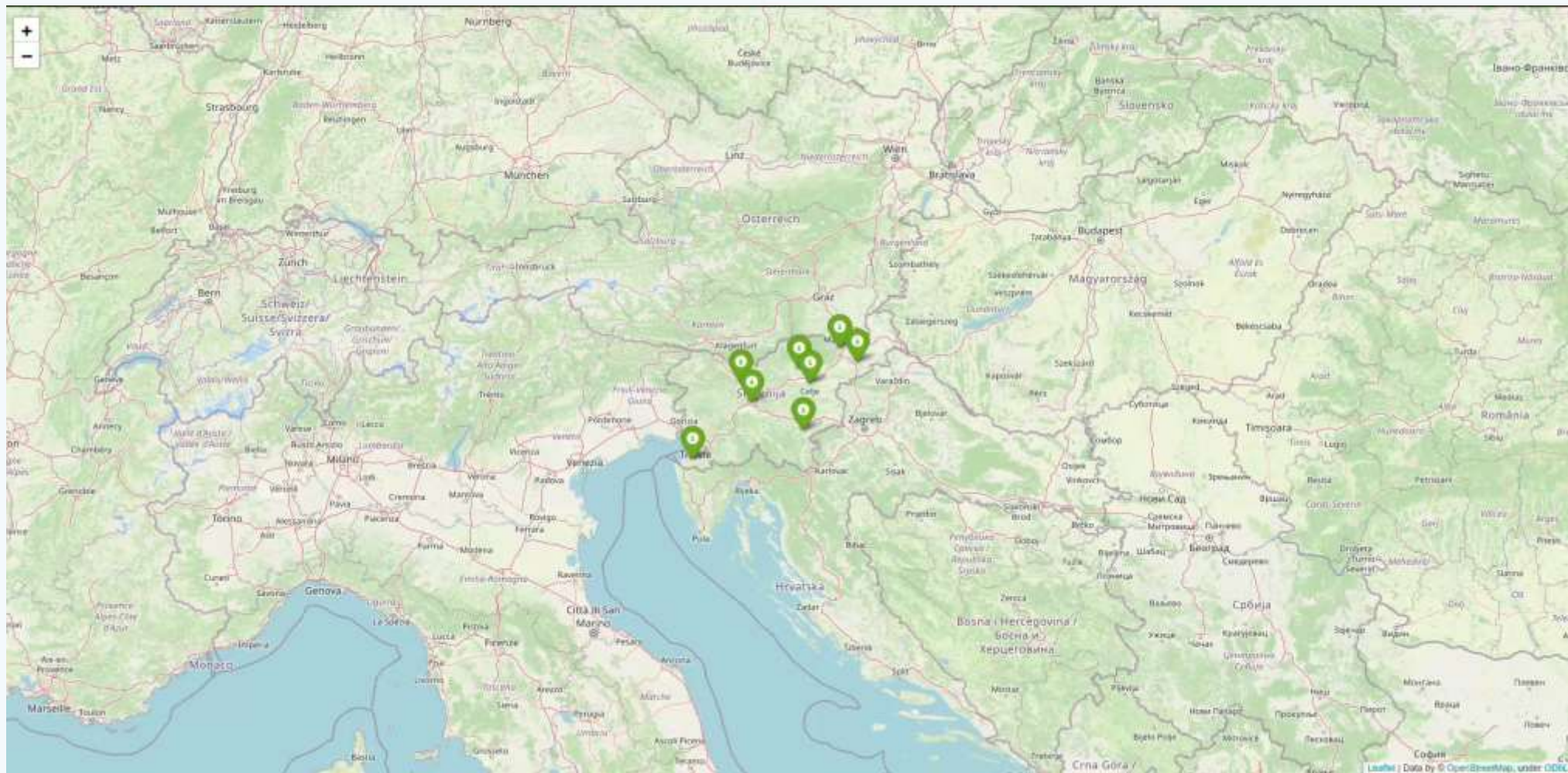
# Ustvari osnovno karto
m = folium.Map(location=[46.1512, 14.9955], zoom_start=7)

# Dodaj markerje za vsako mesto z imenom mesta in naključnim imenom pevca
for index, row in gdf.iterrows():
    ime_mesta = row['Mesto']
    nakljucno_ime_pevca = nakljucna_imena_pevcev[index]
    popup_text = f'<strong>{ime_mesta}</strong><br>Pevčev ime: {nakljucno_ime_pevca}'

    folium.Marker(
        location=(row['Latitude'], row['Longitude']),
        popup=folium.Popup(popup_text, max_width=300),
        icon=folium.Icon(color='green')
    ).add_to(m)

# Shranite zemljevid v HTML datoteko
m.save('zemljevid.html')
```





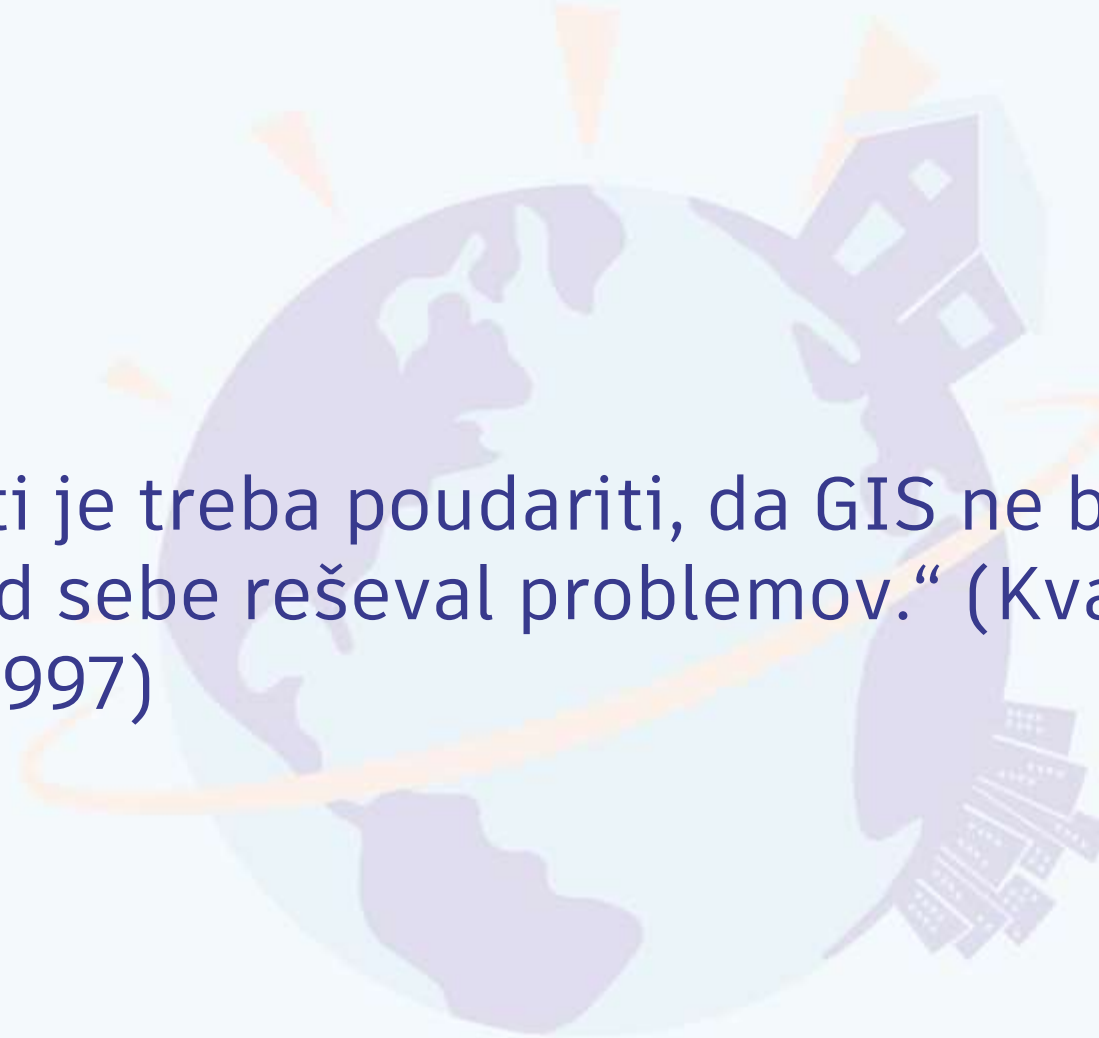
# Pasti pri uporabi umetne inteligence (v GIS)

- Pravilnost poizvedb (preverjanje)
- Pristranskost orodja
- Etični okvir
- Pravni okvir
- Intelektualna lastnina (plagiatorstvo, citiranje orodij)



# Premisa?

„Hkrati je treba poudariti, da GIS ne bo nikoli sam od sebe reševal problemov.“ (Kvamme in sod., 1997)



HVALA ZA POZORNOST!  
SLEDIJO

**VPRAŠANJA/QUESTIONS**

