Argomento: Clustering e analisi dei pattern degli ecosistemi

Corso: Machine Learning per le applicazioni ambientali

Obiettivi di apprendimento

Al termine di questa unità, gli studenti saranno in grado di:

- Applicare tecniche di clustering per identificare pattern e raggruppamenti nei dataset ambientali.
- Valutare la qualità dei cluster mediante metriche di validazione interne ed esterne.
- Interpretare il significato ecologico dei risultati dei modelli non supervisionati.
- Utilizzare i risultati del clustering per sostenere la conservazione e la gestione degli ecosistemi.

Introduzione

Il clustering è un approccio fondamentale dell'apprendimento non supervisionato che organizza le osservazioni in gruppi in base alla loro somiglianza, senza richiedere etichette predefinite. Nell'analisi dei dati ambientali è particolarmente utile, perché gli ecosistemi presentano spesso strutture complesse e multidimensionali, nelle quali i confini tra le classi non sono evidenti. Individuando raggruppamenti naturali nei dati, il clustering supporta attività quali la valutazione della biodiversità, l'analisi della composizione chimica dei suoli e l'interpretazione delle osservazioni da telerilevamento, anche in presenza di rumore e pattern irregolari comuni nelle misurazioni ecologiche [4,6,8]. Il clustering non supervisionato applicato ai dati di produttività da telerilevamento è stato associato a pattern di biodiversità su larga scala [2].

Clustering K-Means

K-Means suddivide il dataset in k cluster assegnando iterativamente i punti al centroide più vicino e aggiornando la posizione dei centroidi per minimizzare la varianza interna ai cluster. È uno degli algoritmi di clustering più diffusi grazie alla semplicità e all'efficienza computazionale.

Vantaggi:

- Rapido e scalabile su dataset di grandi dimensioni.
- Facile da implementare e interpretare.

Limiti:

- Richiede di specificare in anticipo il numero di cluster.k
- È sensibile all'inizializzazione e ai valori anomali.
- Funziona meglio con cluster sferici e di dimensioni simili.

Applicazioni ambientali: individuazione di pattern di inquinamento nelle misure di qualità dell'acqua [9].

Un limite importante dell'algoritmo K-Means è la necessità di specificare in anticipo il numero di cluster (k). Nelle applicazioni ambientali, il numero reale di regimi ecosistemici raramente è noto a priori e una scelta inadeguata di k può produrre rappresentazioni eccessivamente semplificate o frammentate della struttura ecologica.

Per affrontare questo problema, si utilizzano comunemente tecniche di validazione interna per orientare la scelta di k. Tali metodi valutano la qualità del clustering in base alla compattezza e alla separazione dei cluster, senza ricorrere a etichette esterne. Tra gli approcci più diffusi figurano il metodo del gomito (Elbow), che esamina la riduzione della varianza interna ai cluster all'aumentare di k, e l'analisi della silhouette, che valuta quanto ogni punto sia coerente con il cluster assegnato rispetto ai cluster vicini.

In pratica, la scelta finale di k bilancia spesso i risultati quantitativi della validazione con l'interpretabilità ecologica, assicurando che i cluster individuati rappresentino regimi ambientali significativi e non semplici partizioni matematiche.

Clustering gerarchico

Il clustering gerarchico costruisce una gerarchia di cluster, spesso rappresentata mediante un dendrogramma, che mostra livelli annidati di somiglianza tra le osservazioni. Può essere agglomerativo (dal basso verso l'alto) oppure divisivo (dall'alto verso il basso). È particolarmente utile negli studi ecologici che richiedono di comprendere relazioni a più scale.

Vantaggi:

- Non richiede di specificare a priori il numero di cluster.
- Produce dendrogrammi che mostrano strutture ecologiche a più scale.
- Consente diversi criteri di collegamento (single, complete, average).

Limiti:

- È computazionalmente oneroso per dataset molto grandi.

- È sensibile al rumore e alla scala delle variabili.
- I risultati possono essere difficili da interpretare se i cluster non sono ben separati.

Applicazioni ambientali: raggruppamento dei bacini idrografici in base ai regimi di portata e precipitazione [5].

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

DBSCAN raggruppa i punti che si trovano in regioni densamente popolate e identifica come rumore quelli situati in aree a bassa densità. A differenza di K-Means, non richiede di specificare in anticipo il numero di cluster e può individuare cluster di forma arbitraria.

Vantaggi:

- Rileva cluster di forma irregolare.
- Identifica automaticamente rumore e valori anomali.
- Non richiede di stabilire in anticipo il numero di cluster.

Limiti:

- Le prestazioni dipendono fortemente dalla scelta dei parametri (epsilon, minPts).
- Gestisce con difficoltà cluster con densità differenti.
- È meno scalabile di K-Means su dataset molto grandi.

Applicazioni ambientali: individuazione di aree a elevata concentrazione di inquinanti nelle reti di sensori per la qualità dell'aria o dell'acqua [3, 7].

Metriche di valutazione del clustering

A differenza della classificazione supervisionata, nella maggior parte dei dataset ambientali il clustering non dispone di etichette di riferimento. La valutazione della qualità dei cluster si basa quindi su indici di validazione interni o esterni, che quantificano compattezza, separazione oppure concordanza con raggruppamenti definiti da esperti.

Indice di silhouette

- Misura quanto ogni punto è simile agli elementi del proprio cluster rispetto a quelli degli altri cluster.
- Assume valori tra -1 e 1; valori più alti indicano cluster compatti e ben separati.
- È utile per scegliere il numero di cluster k in K-Means.

Indice di Davies–Bouldin

- Rapporto tra la dispersione interna ai cluster e la separazione tra cluster.
- Valori più bassi indicano un clustering migliore.
- È efficiente dal punto di vista computazionale e applicabile a grandi dataset ambientali.

Indice di Calinski–Harabasz

- Confronta la varianza tra cluster con quella interna ai cluster.
- Valori più alti corrispondono a cluster meglio definiti.
- È spesso utilizzato in applicazioni ecologiche con dati multivariati.

Indice di Rand corretto (ARI)

- Metrica di validazione esterna che confronta i risultati del clustering con dati etichettati da esperti o con regioni ecologiche note.
- È corretto per l'accordo casuale: 1 indica concordanza perfetta e 0 una partizione equivalente al caso.
- È utile per confrontare i cluster con mappe di ecoregioni o unità di conservazione già definite.

La validazione dei cluster contribuisce a verificare che le strutture individuate non siano artefatti dell'algoritmo, ma riflettano pattern ecologici significativi. Negli studi ambientali, combinare indici interni (per esempio silhouette e Davies–Bouldin) con una validazione esterna rispetto alle classificazioni degli esperti fornisce elementi più solidi a sostegno dell'interpretazione ecologica.

Caso di studio: individuazione di regimi ecosistemici mediante clustering non supervisionato di dati ambientali

Obiettivo

Questo caso di studio mostra come utilizzare tecniche di apprendimento non supervisionato per individuare raggruppamenti naturali e regimi ecologici nei dati ambientali senza etichette predefinite. L'obiettivo è identificare regioni con caratteristiche ambientali simili e interpretare tali cluster come regimi ecosistemici o ambientali distinti. La scoperta di questi pattern è particolarmente utile nell'analisi ecologica, dove i confini naturali sono spesso graduali, sfumati o sconosciuti.

Librerie utilizzate

Il caso di studio è implementato in Python e utilizza librerie open source ampiamente diffuse per l'analisi dei dati, il machine learning e la visualizzazione:

- **NumPy**
  Utilizzata per operazioni numeriche, manipolazione di array e campionamento casuale.
- **Pandas**
  Utilizzata per caricare, gestire e preprocessare dataset ambientali tabellari.
- **Scikit-learn (sklearn).**
  Utilizzata per:
  - preprocessing dei dati (StandardScaler),
  - algoritmi di clustering (K-Means),
  - riduzione della dimensionalità (PCA),
  - metriche di validazione interna dei cluster (indice di silhouette, indice di Davies–Bouldin).
- **SciPy**
  Utilizzata per il clustering gerarchico e la visualizzazione di dendrogrammi.
- **Matplotlib**
  Utilizzata per visualizzare i risultati del clustering, grafici di dispersione, proiezioni PCA e dendrogrammi.

Questa scelta favorisce la trasparenza e rende il procedimento facilmente eseguibile in ambienti standard quali Jupyter Notebook o Google Colab.

Fonte dei dati

L'analisi si basa sul dataset Forest Cover Type (Covertype) [1], disponibile liberamente presso l'UCI Machine Learning Repository. Il dataset contiene misure ambientali e topografiche raccolte in aree boschive della Roosevelt National Forest, negli Stati Uniti. Sebbene il dataset includa un'etichetta relativa al tipo di copertura forestale, questa informazione viene volutamente esclusa dal caso di studio per preservare il carattere non supervisionato dell'analisi.

Descrizione dei dati

Ogni osservazione rappresenta una posizione geografica descritta da variabili ambientali quali:

- Altitudine
- Pendenza
- Esposizione
- Distanze orizzontali e verticali dai corsi d'acqua
- Distanza dalle strade
- Indici di ombreggiamento del terreno

Nel loro insieme, queste variabili descrivono il terreno, i gradienti di umidità e l'accessibilità, fattori importanti per la differenziazione degli ecosistemi.

Preprocessing dei dati

Per preparare il dataset al clustering si eseguono i seguenti passaggi:

- Selezione delle sole variabili ambientali e topografiche continue
- Rimozione dell'etichetta del tipo di copertura forestale
- Standardizzazione delle variabili a media zero e varianza unitaria, per garantire un contributo equilibrato ai metodi di clustering basati sulle distanze

```python
import pandas as pd
from sklearn.preprocessing import StandardScaler

# Load dataset (CSV version of Covertype)
df = pd.read_csv("covtype.csv")

# Drop the cover type label (last column)
X = df.iloc[:, :-1]

# Standardize features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

pd.DataFrame(X_scaled, columns=X.columns).head()
```

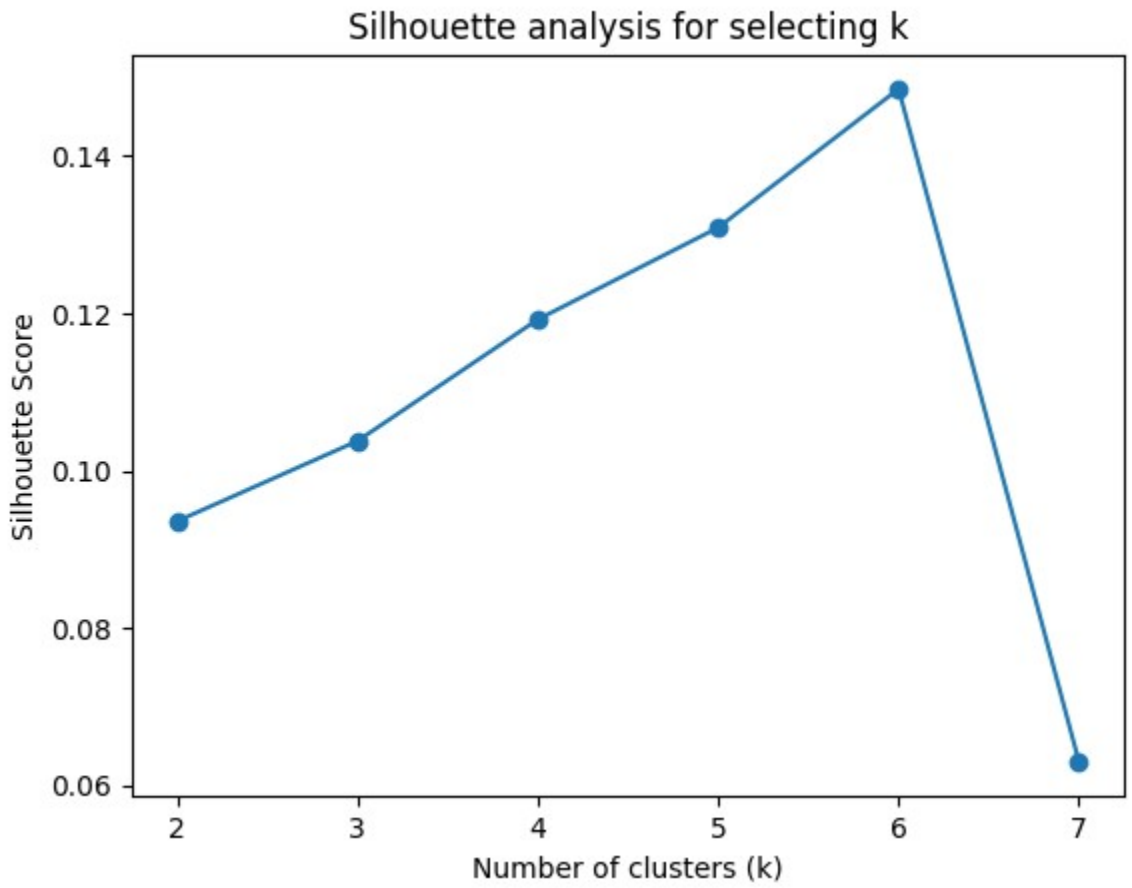
Metodi di clustering

Si applicano due tecniche di clustering complementari:

Clustering K-Means

K-Means viene utilizzato per suddividere il dataset in un numero predefinito di cluster in base alla somiglianza ambientale. Prima di applicarlo, si esamina il numero di cluster mediante tecniche di validazione interna. L'analisi della silhouette permette di valutare la qualità del clustering per diversi valori di k. Queste analisi supportano la scelta di un numero di cluster che bilanci separazione, compattezza e interpretabilità ecologica.

```python
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
import matplotlib.pyplot as plt

# Range of k values to test
K_range = range(2, 8)

silhouette_scores = []

for k in K_range:
    print(f"Computing Silhouette Score for k = {k} ...")

    kmeans = KMeans(n_clusters=k, random_state=42)
    labels = kmeans.fit_predict(X_scaled)

    score = silhouette_score(X_scaled, labels)
    silhouette_scores.append(score)

# Plot results
plt.figure()
plt.plot(K_range, silhouette_scores, marker='o')
plt.xlabel("Number of clusters (k)")
plt.ylabel("Silhouette Score")
plt.title("Silhouette analysis for selecting k (full dataset)")
plt.show()
```



L’analisi della silhouette ha evidenziato punteggi più elevati per k = 6, suggerendo una migliore separazione dei cluster dal punto di vista puramente statistico. Tuttavia, nelle applicazioni ambientali, il numero ottimale di cluster non dipende soltanto dalle metriche di validazione interna. Gli ecosistemi sono caratterizzati da gradienti continui piuttosto che da classi nettamente separate; un numero crescente di cluster può quindi frammentare regimi ecologicamente coerenti in unità più piccole e meno interpretabili.

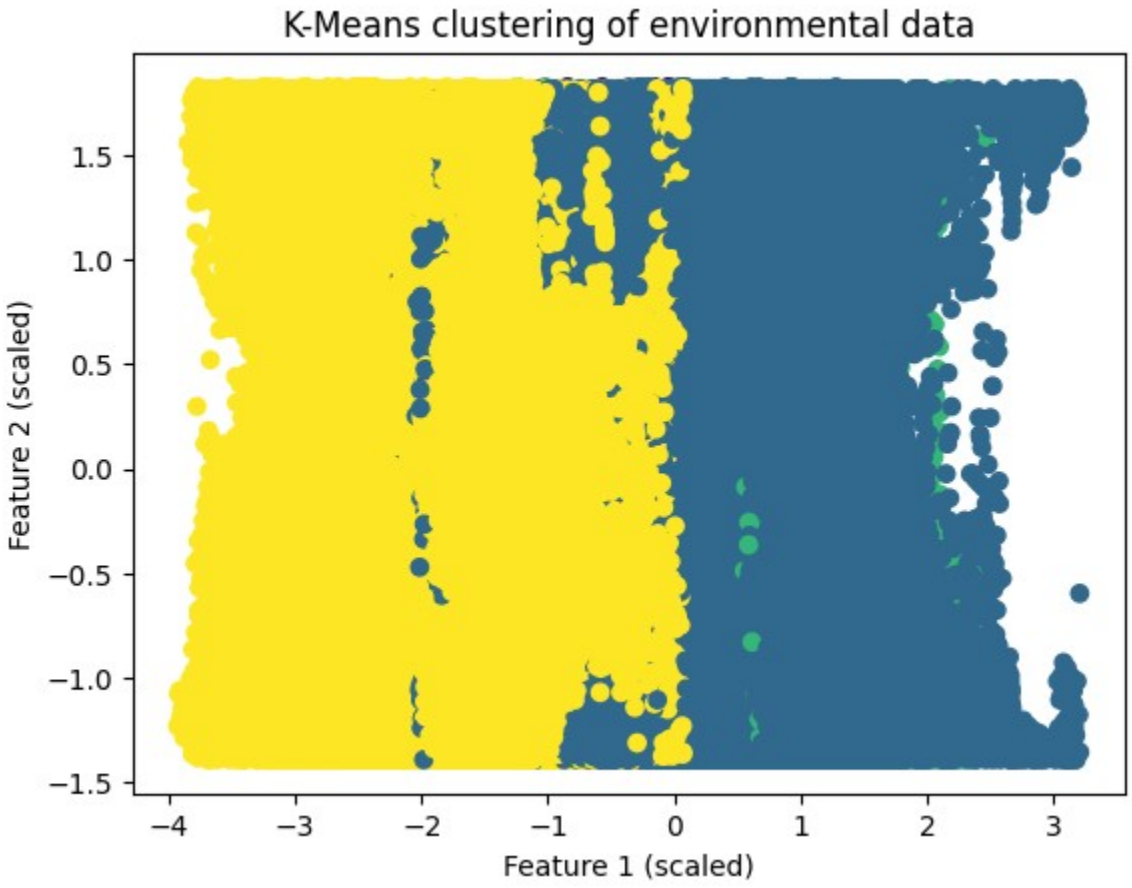
Si è pertanto scelto un numero inferiore di cluster (k = 4), per bilanciare la qualità statistica del clustering con l’interpretabilità ecologica e la chiarezza didattica. Questa scelta consente di individuare regimi ecosistemici ampi e significativi, coerenti con i principali gradienti ambientali, evitando una suddivisione eccessiva che potrebbe ostacolare la comprensione e l’applicazione pratica.

```python
from sklearn.cluster import KMeans

# Apply K-Means
kmeans = KMeans(n_clusters=4, random_state=42)
clusters = kmeans.fit_predict(X_scaled)

# Add cluster labels to dataframe
df["Cluster"] = clusters

df[["Cluster"]].value_counts()
```

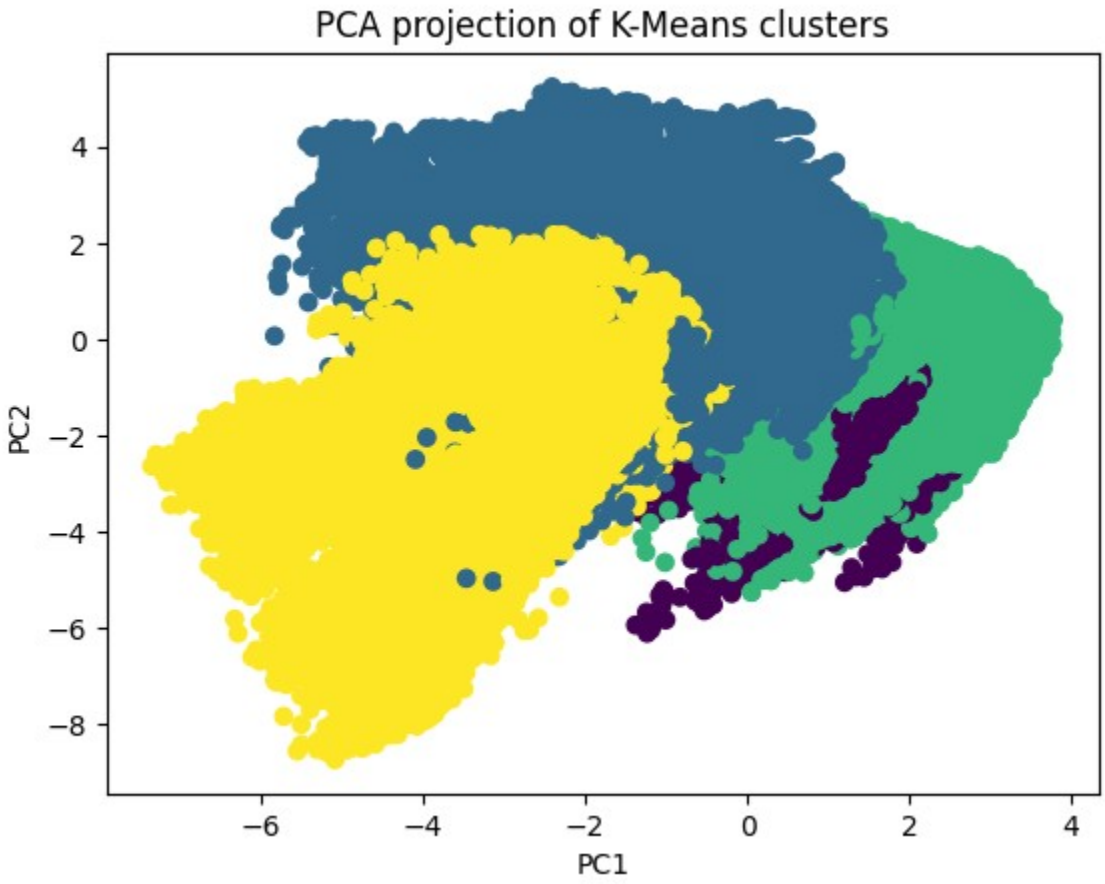
✓ 1.9s

```
Cluster
1          236975
2          228173
3           84547
0           31317
Name: count, dtype: int64
```

```python
import matplotlib.pyplot as plt

plt.figure()
plt.scatter(X_scaled[:, 0], X_scaled[:, 1], c=clusters)
plt.xlabel("Feature 1 (scaled)")
plt.ylabel("Feature 2 (scaled)")
plt.title("K-Means clustering of environmental data")
plt.show()
```



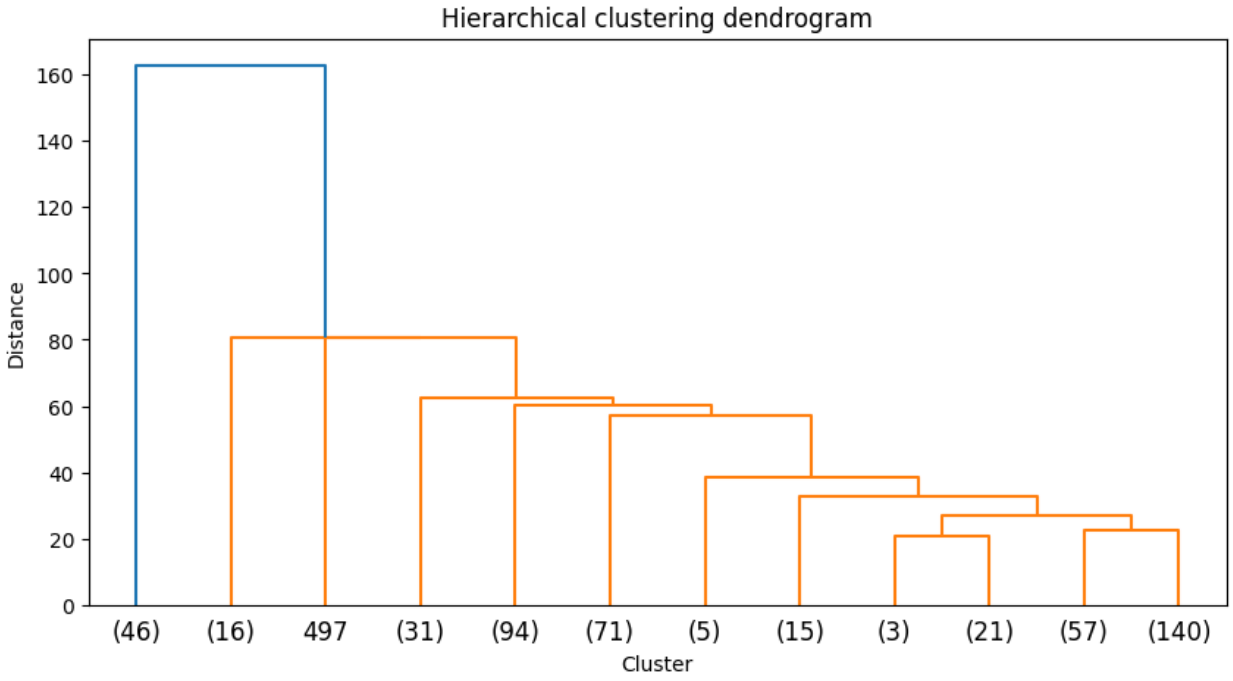

```python
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

# Apply PCA to reduce to 2 dimensions
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)

plt.figure()
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=clusters)
plt.xlabel("PC1")
plt.ylabel("PC2")
plt.title("PCA projection of K-Means clusters")
plt.show()
```



Clustering gerarchico

Il clustering gerarchico agglomerativo viene applicato per esplorare le relazioni annidate tra le località di campionamento. I dendrogrammi visualizzano strutture ecologiche a più scale e mostrano come i pattern locali si aggreghino in raggruppamenti ecosistemici più ampi.

```python
from scipy.cluster.hierarchy import linkage, dendrogram

# Use a subset for visualization
X_subset = X_scaled[:500]

linked = linkage(X_subset, method="ward")

plt.figure(figsize=(10, 5))
dendrogram(linked, truncate_mode="lastp", p=12)
plt.title("Hierarchical clustering dendrogram")
plt.xlabel("Cluster")
plt.ylabel("Distance")
plt.show()
```



Valutazione dei cluster

La qualità dei cluster viene valutata mediante indici di validazione interna:

- Indice di silhouette
- Indice di Davies–Bouldin

Queste metriche aiutano a verificare che i cluster individuati siano compatti, ben separati e non semplici artefatti di una partizione casuale.

```
from sklearn.metrics import silhouette_score, davies_bouldin_score

sil_score = silhouette_score(X_scaled, clusters)
db_score = davies_bouldin_score(X_scaled, clusters)

sil_score, db_score
```

Il calcolo di metriche di validazione interna come l'indice di silhouette e l'indice di Davies–Bouldin può essere oneroso su dataset ambientali di grandi dimensioni, poiché tali metriche si basano su calcoli delle distanze tra coppie di osservazioni. Date le dimensioni del dataset utilizzato in questo caso di studio, la qualità del clustering è stata quindi valutata su un sottoinsieme casuale rappresentativo dei dati. Questo approccio mantiene le caratteristiche essenziali della struttura dei dati e rende il calcolo sostenibile; è una pratica comune nelle analisi di clustering su larga scala. Le metriche ottenute offrono indicazioni comparative attendibili sulla qualità dei cluster senza richiedere calcoli esaustivi sull'intero dataset.

```
import numpy as np
from sklearn.metrics import silhouette_score, davies_bouldin_score

# Subsample for faster evaluation
np.random.seed(42)
sample_idx = np.random.choice(len(X_scaled), size=5000, replace=False)

X_sample = X_scaled[sample_idx]
clusters_sample = clusters[sample_idx]

sil_score = silhouette_score(X_sample, clusters_sample)
db_score = davies_bouldin_score(X_sample, clusters_sample)

sil_score, db_score
```

Il calcolo di metriche di validazione interna come l'indice di silhouette e l'indice di Davies–Bouldin può essere oneroso su dataset ambientali di grandi dimensioni, poiché tali metriche si basano su calcoli delle distanze tra coppie di osservazioni. Date le dimensioni del dataset utilizzato in questo caso di studio, la qualità del clustering è stata quindi valutata su un

sottoinsieme casuale rappresentativo dei dati. Questo approccio mantiene le caratteristiche essenziali della struttura dei dati e rende il calcolo sostenibile; è una pratica comune nelle analisi di clustering su larga scala. Le metriche ottenute offrono indicazioni comparative attendibili sulla qualità dei cluster senza richiedere calcoli esaustivi sull'intero dataset.

```
import numpy as np
from sklearn.metrics import silhouette_score, davies_bouldin_score

# Subsample for faster evaluation
np.random.seed(42)
sample_idx = np.random.choice(len(X_scaled), size=5000, replace=False)

X_sample = X_scaled[sample_idx]
clusters_sample = clusters[sample_idx]

sil_score = silhouette_score(X_sample, clusters_sample)
db_score = davies_bouldin_score(X_sample, clusters_sample)

sil_score, db_score
```

Interpretazione e significato ecologico

I cluster risultanti possono essere interpretati come regimi ecosistemici distinti, per esempio:

- Ambienti forestali a bassa e ad alta quota
- Terreni umidi e terreni asciutti
- Regioni montane accessibili e regioni remote

Confrontando i cluster con gradienti ambientali noti, gli studenti possono associare ciascun gruppo a condizioni ecologiche caratteristiche anziché a classi predefinite di copertura del suolo.

Rilevanza per la gestione degli ecosistemi

I cluster individuati forniscono una base oggettiva per:

- Classificare e mappare gli ecosistemi
- Individuare regioni ambientalmente simili per pianificare interventi di conservazione

- Supportare decisioni sulla gestione dell'uso del suolo in aree con pochi dati o scarsamente cartografate

Questo caso di studio mostra come il clustering integri le conoscenze degli esperti, facendo emergere strutture ecologiche basate sui dati che non coincidono necessariamente con i confini amministrativi.

Esercitazioni

- Scaricare il dataset Forest Cover Type (Covertype) dall'UCI Machine Learning Repository [1].
- Esplorare il dataset e svolgere un'analisi descrittiva iniziale delle variabili ambientali e topografiche.
- Selezionare un sottoinsieme di variabili continue rilevanti (per esempio altitudine, pendenza e distanze idrologiche) e motivare la scelta.
- Standardizzare le variabili selezionate ed esaminare l'effetto della scala sul clustering basato sulle distanze.
- Utilizzare tecniche di validazione interna (metodo del gomito e analisi della silhouette) per esplorare diversi valori del numero di cluster (k).
- Applicare K-Means con il valore di k scelto e analizzare la struttura dei cluster ottenuti.
- Eseguire il clustering gerarchico su un sottoinsieme ridotto dei dati e visualizzare le relazioni a più scale mediante un dendrogramma.
- Confrontare i risultati ottenuti con K-Means e con i metodi gerarchici, evidenziando somiglianze e differenze.
- Interpretare i cluster in termini di gradienti ambientali dominanti, anziché di classi predefinite di copertura del suolo.
- Riflettere sui limiti del clustering nell'analisi degli ecosistemi e discutere come i risultati possano integrare le classificazioni degli esperti e le decisioni di gestione.

Valutazione

1. Qual è lo scopo principale delle tecniche di clustering nell'analisi dei pattern degli ecosistemi?
    A. Prevedere variabili ambientali future
    B. Assegnare le osservazioni a classi di rischio predefinite
    C. Individuare raggruppamenti e pattern naturali in dati ambientali privi di etichette
    D. Ridurre la dimensionalità esclusivamente per la visualizzazione

2. Quale limite caratterizza maggiormente l'algoritmo K-Means nei dataset ambientali?
    A. L'impossibilità di gestire grandi dataset
    B. La necessità di specificare in anticipo il numero di cluster (k)
    C. L'impossibilità di utilizzare variabili numeriche
    D. L'impossibilità di individuare cluster compatti

3. Quale metodo di clustering è più adatto a individuare aree di inquinamento di forma irregolare, identificando anche il rumore?
    A. K-Means
    B. Clustering gerarchico
    C. DBSCAN
    D. Regressione logistica

4. Perché nell'analisi di clustering si utilizzano spesso metriche di validazione interna come l'indice di silhouette?
    A. Perché il clustering dispone sempre di etichette di riferimento
    B. Per prevedere cambiamenti futuri degli ecosistemi
    C. Per valutare compattezza e separazione dei cluster senza dati etichettati
    D. Per ottimizzare classificatori supervisionati

5. In che modo i risultati del clustering possono supportare le decisioni di gestione e conservazione degli ecosistemi?
    A. Sostituendo le osservazioni ecologiche sul campo
    B. Fornendo previsioni esatte dei cambiamenti ecosistemici
    C. Rivelando gruppi funzionali e gradienti ecologici utili alla gestione
    D. Eliminando l'incertezza dai dati ambientali

Fonti aggiuntive

- Machine Learning - A First Course for Engineers and Scientists, Andreas Lindholm, Niklas Wahlström, Fredrik Lindsten, and Thomas B. Schön

- Deep Learning with Python, Second Edition, François Chollet
- Applications of Deep Neural Networks with Keras, Jeff Heaton
- Understanding Machine Learning: From Theory to Algorithms, Shai Shalev-Shwartz and Shai Ben-David

Bibliografia

[1] Blackard, J. (1998). Covertype [Dataset]. UCI Machine Learning Repository. https://doi.org/10.24432/C50K5N.

[2] Coops, N. C., Kearney, S. P., Bolton, D. K., & Radeloff, V. C. (2018). Remotely-sensed productivity clusters capture global biodiversity patterns. Scientific Reports, 8, 16261.

[3] Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD) (pp. 226–231).

[4] Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. ACM Computing Surveys, 31 (3), 264–323.

[5] Jehn, F. U., Bestian, K., Breuer, L., Kraft, P., & Houska, T. (2020). Using hydrological and climatic catchment clusters to explore drivers of catchment behavior. Hydrology and Earth System Sciences, 24(3), 1081–1100.

[6] Legendre, P., & Legendre, L. (2012). Numerical Ecology (3rd ed.). Elsevier.

[7] Shetty, C., Seema, S., Sowmya, B. J., Nandalike, R, Supreeth, S., Dayananda, P., Rohith, S., Ranjan, R., & Goud, V. (2024). A Machine Learning Approach for Environmental Assessment on Air Quality and Mitigation Strategy. Journal of Engineering, 2024(1), 2893021.

[8] Tan, P.-N., Steinbach, M., Karpatne, A., & Kumar, V. (2018). Introduction to Data Mining (2nd ed.). Pearson.

[9] Zou, K. (2022). The Dynamic Detection of Water Quality Monitoring and Pollution Prevention and Control. In Journal of Physics: Conference Series (Vol. 2152, No. 1, p. 012017). IOP Publishing.