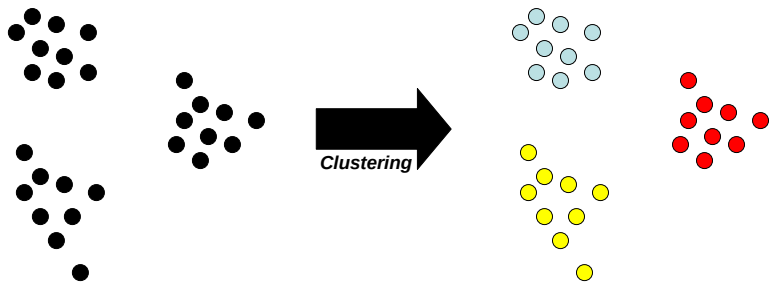
Clustering

Iuri Frosio
frosio@dsi.unimi.it

Approfondimenti in A.K. Jan, M. N. Murty, P. J. Flynn, “Data clustering: a review”, ACM Computing Surveys, Vol. 31, No. 3, September 1999, ref. pp. 265-290, disponibile in http://citeseer.ist.psu.edu/jain99data.html

Clustering

- Intuitivamente…



Il clustering per…

… Confermare ipotesi sui dati (es. “E’ possibile identificare tre diversi tipi di clima in Italia: mediterraneo, continentale, alpino…”);

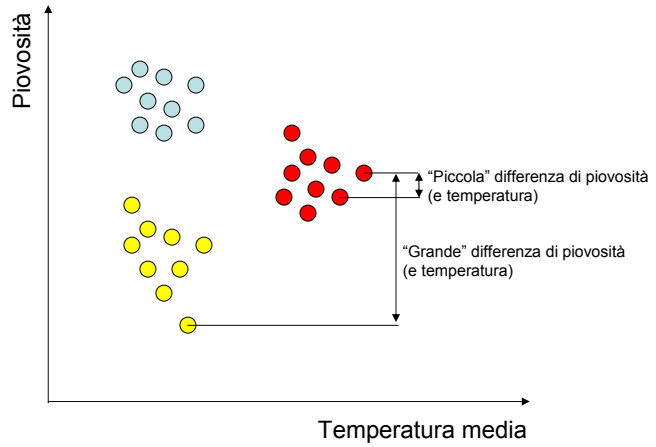
… Esplorare lo spazio dei dati (es. “Quanti tipi diversi di clima sono presenti in Italia?”);

… Semplificare l’interpretazione dei dati (“Il clima di ogni città d’Italia è approssimativamente mediterraneo, continentale o alpino.”).

Cluster analysis

- Cluster analysis: organizzazione di una collezione di dati (pattern, vettori) in cluster, basata sulla similarità.
- I pattern appartenenti ad un cluster valido sono più simili l’uno con l’altro rispetto ai pattern appartenenti ad un cluster differente.

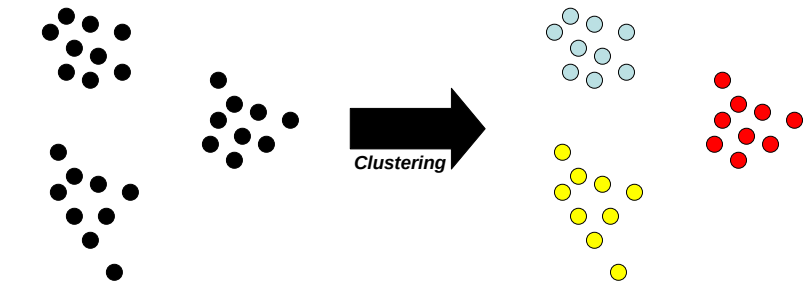
Cluster analysis



Clustering vs. Discriminant analysis

- Si definisce clustering un procedimento **non supervisionato** di **classificazione di pattern non precedentemente etichettati** in cluster.
- Nel caso in cui i pattern siano stati precedentemente etichettati, si parla di discriminant analysis (determinazione della classe di appartenenza di un nuovo pattern, data in partenza una serie di pattern già etichettati).

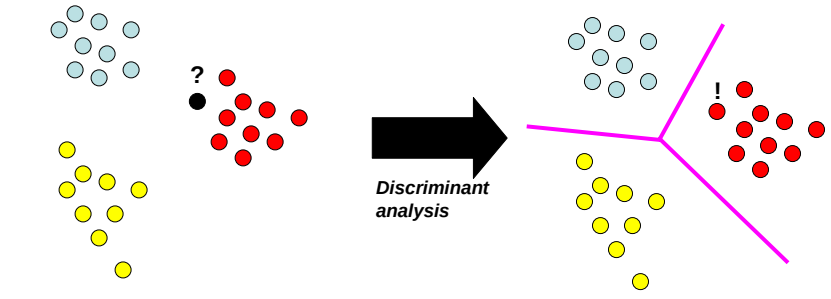
Clustering vs. Discriminant analysis



Dati non etichettati

Dati etichettati

Clustering vs. Discriminant analysis



Dati etichettati + nuovo dato

Nuovo dato etichettato

Passi del clustering

1) Rappresentazione dei pattern;
2) Definizione di una misura di prossimità dei pattern;
3) Clustering propriamente detto;
4) Data abstraction (se necessario);
5) Validazione dell'output (se necessario).

Rappresentazione dei pattern

- Numero di classi, numero di pattern disponibili;
- Numero, tipo e scala delle **feature** (caratteristiche) utilizzate dall'algoritmo di clustering.

Roma : [17°; 500mm] Pattern

Milano : [13°; 900mm]

Feature 1 Feature 2

Rappresentazione dei pattern

- Feature selection: identificazione delle feature più significative per la descrizione dei pattern.

  esempio:
  Roma: [17°; 500mm; ~~1.500.000 ab.~~]

- Feature extraction: trasformazione delle feature per creare nuove, significative feature;

  esempio:
  Milano: [13°; 900mm; 255 giorni sole; 100 giorni pioggia]
  oppure
  Milano: [13°; 900mm / 100 giorni pioggia; 255 giorni sole]

Prossimità dei pattern

- Definizione di una **misura di distanza tra due patterns**;

  esempio:
  Distanza euclidea…
  dist (Roma, Milano) = …
  dist ([17°; 500mm], [13°; 900mm]) = …
  = … Distanza euclidea? = …
  = $((17-13)^2+(500-900)^2)^{1/2}$ = 400.02 ~ 400

Normalizzazione feature

Att.ne!

dist (Roma, Milano) = …

dist ([17°; 500mm], [13°; 900mm]) = …

= … Distanza euclidea? = …

= $((17-13)^2+(500-900)^2)^{½}$ = 400.02 ~ 400

La distanza tra le due città in termini di gradi è insignificante nel nostro calcolo… **E' necessario normalizzare i dati**!

Es.

T_{Max}= 20° T_{Min}= 5° → $T_{Norm} = (T - T_{Min})/(T_{Max} - T_{Min})$

P_{Max}= 1000mm P_{Min}= 0mm → $P_{Norm} = (P - P_{Min})/(P_{Max} - P_{Min})$

$Roma_{Norm}$ = [0.8 0.5]

$Milano_{Norm}$ = [0.53 0.9]

$dist(Roma_{Norm}, Milano_{Norm}) = ((0.8-0.53)^2+(0.5-0.9)^2)^{½}$ = 0.4826

Altre funzioni di distanza

- Distanza euclidea:
  $dist(x,y)=[\Sigma_{k=1..d}(x_k-y_k)^2]^{1/2}$
- Minkowski:
  $dist(x,y)=[\Sigma_{k=1..d}(x_k-y_k)^p]^{1/p}$
- Mahalanobis:
  $dist(x,y)= (x_k-y_k)S^{-1}(x_k-y_k)$
- Context dependent:
  dist(x,y)= f(x, y, context) *

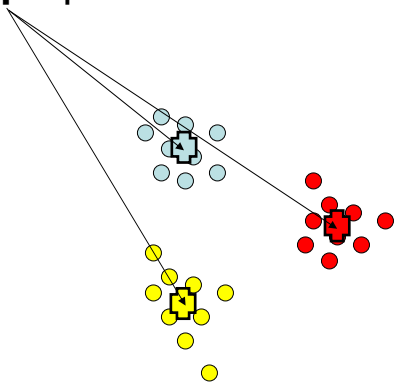
*Approfondimenti in A.K. Jan, M. N. Murty, P. J. Flynn, "Data clustering: a review", ACM Computing Surveys, Vol. 31, No. 3, September 1999, ref. pp. 272-273.

Clustering

- Hard vs. fuzzy clustering;
- Hierarchical vs. partitional clustering;
- Agglomerative vs. divisive;
- Deterministic vs. stochastic…

Data abstraction

- **Prototipi** per la descrizione di ciascuna classe.



Validazione dell'output

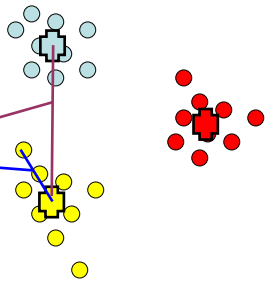
- Valutazione qualitativa (es. operatore umano) della classificazione;
- Valutazione quantitativa della classificazione;

  es. minimo di:

  $k_1 \cdot \Sigma_j \Sigma_i [dist(x_i^j, prot_j)] / N_j -$

  $k_2 \cdot \Sigma_j \Sigma_h [dist(prot_j, prot_h)]$

  La valutazione quantitativa è più semplice se ci si basa su metodi statistici.



Clustering: definizioni

- **Pattern**: un singolo dato $x = [x_1, x_2, \ldots x_d]$;
- **Feature**: ogni componente $x_1, x_2, \ldots x_d$;
- **d**: dimensione dello spazio delle feature;
- **Classe**: in generale, un processo che governa la generazione dei pattern di un determinato cluster (in particolare, la ddp dei dati di un cluster);
- **Hard vs. fuzzy clustering**: ogni pattern è etichettato con una sola label nel primo caso o da un grado di appartenenza ad un cluster nel secondo caso;
- **Funzione di distanza**: una metrica (o quasi metrica) nello spazio delle feature, usata per quantificare la similarità tra due pattern.

Altre funzioni di distanza

- Distanza euclidea:
  $dist(x,y)=[\Sigma_{k=1..d}(x_k-y_k)^2]^{1/2}$
- Minkowski:
  $dist(x,y)=[\Sigma_{k=1..d}(x_k-y_k)^p]^{1/p}$
- Mahalanobis:
  $dist(x,y)= (x_k-y_k)S^{-1}(x_k-y_k)$
- Context dependent:
  $dist(x,y)= f(x, y, context)$ *

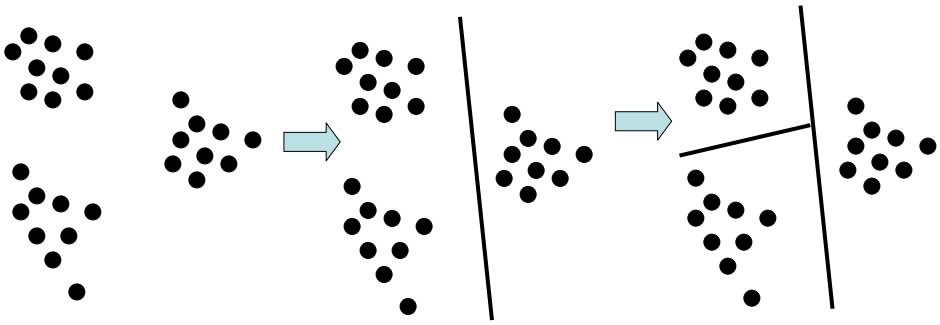
*Approfondimenti in A.K. Jan, M. N. Murty, P. J. Flynn, “Data clustering: a review”, ACM Computing Surveys, Vol. 31, No. 3, September 1999, ref. pp. 272-273.

Tecniche per il clustering

- **Hierarchical vs. partitional**: Gli algoritmi gerarchici producono una serie di partizioni al contrario degli algoritmi partitional, che lavorano sempre con una sola partizione.

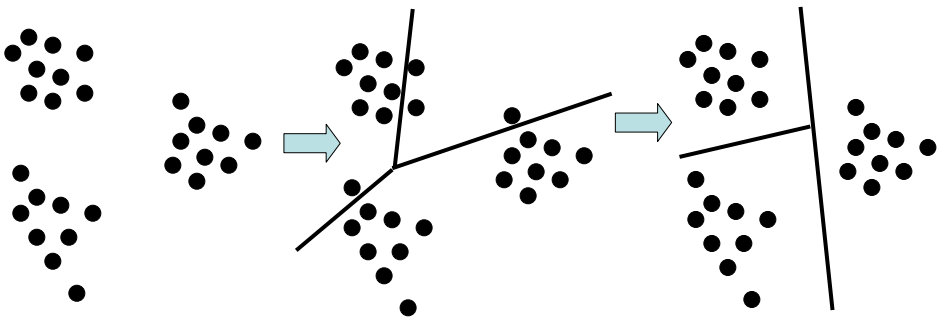
Tecniche per il clustering

Hierarchical:



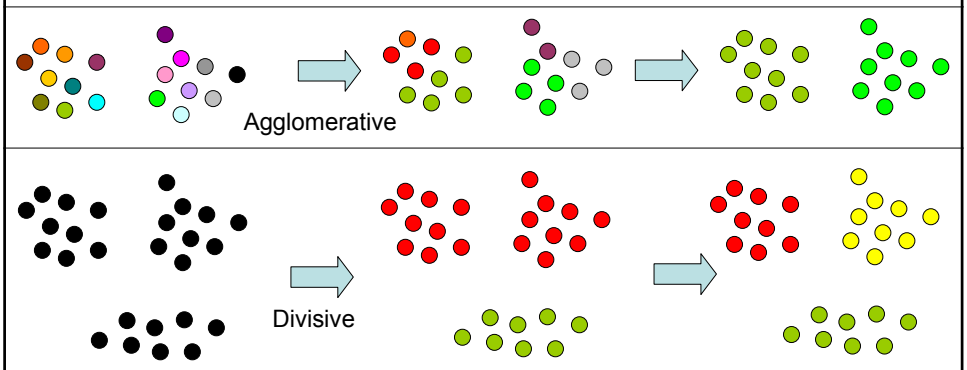
Tecniche per il clustering

Partitional:



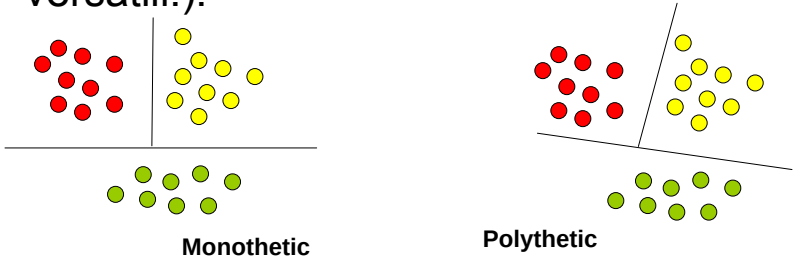
Tecniche per il clustering

- **Agglomerative vs. divisive**: i dati vengono accorpati oppure divisi nel processo di clustering.



Tecniche per il clustering

- **Monothetic vs. polythetic**: si considera una sola feature (monothetic) oppure tutte le feature (polythetic) per volta per effettuare il clustering… La maggior parte degli algoritmi sono di tipo polythetic (più versatili!).



Tecniche per il clustering

- **Hard vs. fuzzy**: label vs. grado di appartenenza. Un algoritmo fuzzy può sempre essere defuzzyficato (classificazione dell'i-esimo pattern nella classe con grado di appartenenza più alto).

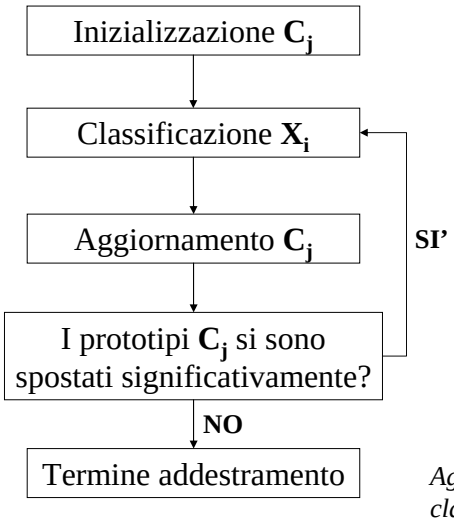
Tecniche per il clustering

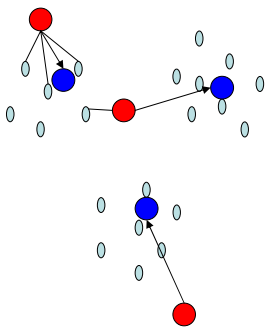
- **Deterministic vs. stochastic**: introduzione di elementi random nell'algoritmo di clusterizzazione (es. GA) che rendono l'output dell'algoritmo non predicibile sulla base dei dati di partenza (stochastic).

K-means (partitional): framework

- Siano $\mathbf{X_1}$, …, $\mathbf{X_D}$ i dati di addestramento (per semplicità, definiti in R^2);
- siano $\mathbf{C_1}$, …, $\mathbf{C_K}$ i *prototipi* di K classi, definiti anch'essi in R^2; ogni *prototipo* identifica il baricentro della classe corrispondente;
- lo schema di classificazione adottato sia il seguente: "$\mathbf{X_i}$ appartiene a $\mathbf{C_j}$ se e solo se $\mathbf{C_j}$ è il *prototipo* più vicino a $\mathbf{X_i}$ (distanza euclidea)**";**
- l'algoritmo di addestramento permette di determinare le posizioni dei *prototipi* $\mathbf{C_j}$ mediante successive approssimazioni.

K-means: addestramento





Aggiornamento C_j: baricentro degli X_i classificati da C_j.

K-means: limiti

- Partitional, polythetic, hard, deterministic;
- Veloce, semplice da implementare;
- Trova un minimo locale della funzione $f=\Sigma_j\Sigma_i[dist(x_i,prot_j)] / N_j$;
- Il risultato dipende dall'inizializzazione!
- Possono essere usati altri metodi (es. GA) per inizializzare K-means… es. GA per la minimizzazione di f, effettuano una ricerca globale, ma sono lenti!

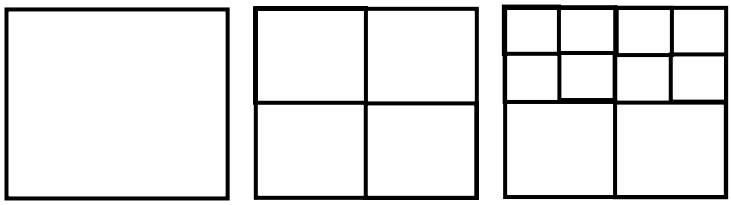
K-Means e EM

- K-Means come versione "limite" di EM (assegnazione "hard" delle responsabilità delle diverse componenti per ciascun dato);*
- Expectation → assegnazione (hard) dei dati ai cluster;
- Maximization → aggiornamento dei prototipi.

*Approfondimenti in Cristopher M. Bishop, Pattern Recognition and Machine Learning, Capitolo 9.1, (K-means vs. mixture models, EM).

Algoritmi hierarchical: QTD

- Quad Tree Decomposition;
- Suddivisione gerarchica dello spazio delle feature, mediante splitting dei cluster;
- Criterio di splitting (~distanza tra cluster).



Algoritmi hierarchical: QTD

- Clusterizzazione immagini RGB, 512x512;
- Pattern: pixel (x,y);
- Feature: canali R, G, B.
- Distanza tra due pattern (non euclidea):
  dist (p1, p2) =
  dist ([R1 G1 B1], [R2 G2 B2]) =
  max (|R1-R2|, |G1-G2|, |B1-B2|).

Algoritmi hierarchical: QTD

p1 = [0 100 250]
p2 = [50 100 200]
p3 = [255 150 50]

dist (p1, p2) = dist ([R1 G1 B1], [R2 G2 B2]) =
max (|R1-R2|, |G1-G2|, |B1-B2|) = max([50 0 50]) = 50.

dist (p2, p3) = 205.

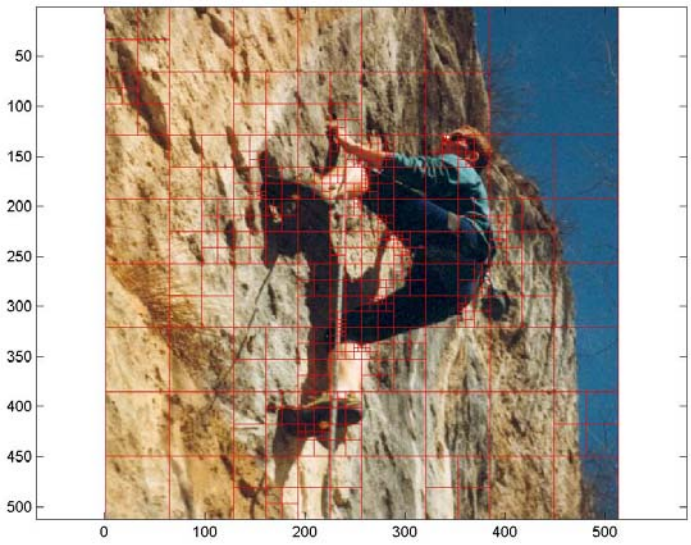
dist (p3, p1) = 255.

Algoritmi hierarchical: QTD

Criterio di splitting: se due pixel all'interno dello stesso cluster distano più di una determinata soglia, il cluster viene diviso in 4 cluster.

Esempio applicazione: segmentazione immagini, compressione immagini, analisi locale frequenze immagini…

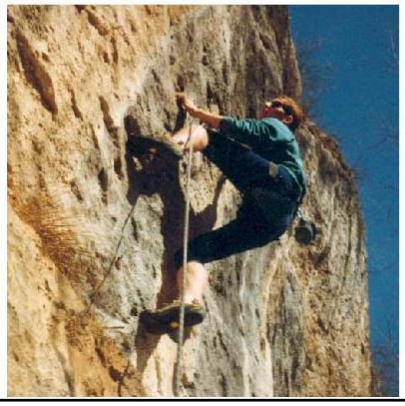
QTD: Risultati

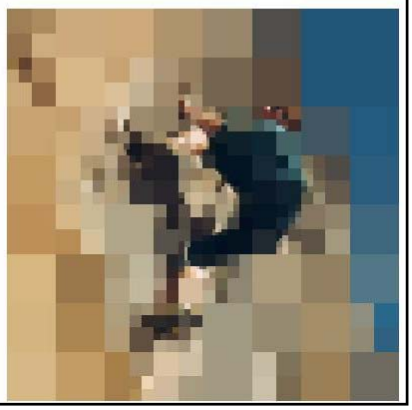


QTD: Risultati

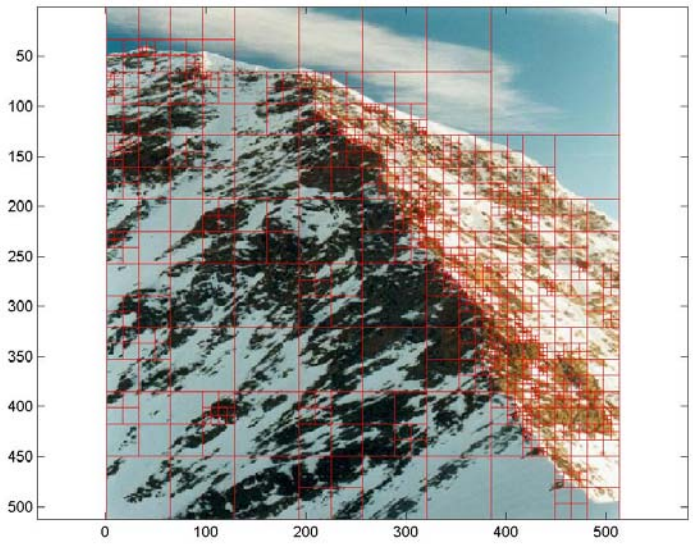
Original

Clusterized





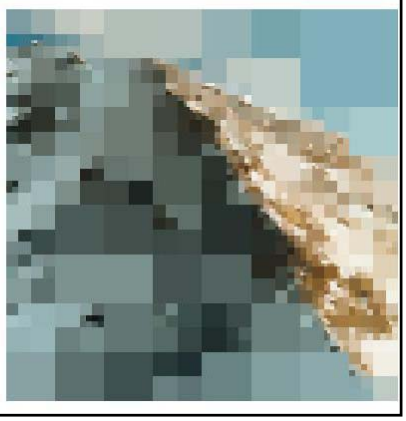
QTD: Risultati



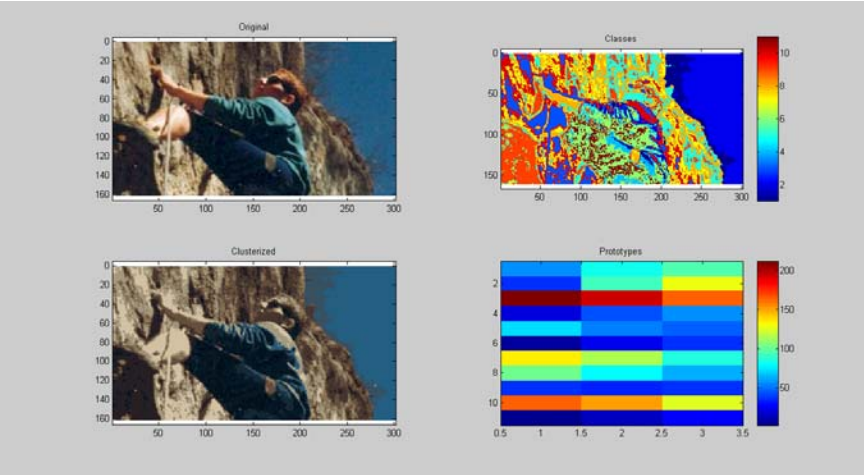
QTD: Risultati

Original

Clusterized



K-Means per immagine RGB



M-Files

- M-files (disponibili sul sito http://homes.dsi.unimi.it/~frosio/) e documenti allegati:
  - KMeans;
  - KMeans per clusterizzazione immagini RGB;
  - QTD Clustering.