Business Intelligence per i Big Data

Esercitazione di laboratorio N. 6

Gli obiettivi dell'esercitazione sono:

- **applicare i principali algoritmi di clustering disponibili in RapidMiner per segmentare gli utenti della campagna in funzione delle loro caratteristiche anagrafiche e lavorative e le news in base alla similarità dei termini in esse contenuti.**

Dati strutturati

Il dataset denominato UsersSmall (UsersSmall.xls) è disponibile sul sito del corso (http://dbdmg.polito.it/wordpress/teaching/business-intelligence/). Esso raccoglie dati anagrafici e lavorativi relativi a circa 300 persone contattate da un'azienda per proporgli l'iscrizione ad un loro servizio. Per tali utenti è noto se, dopo essere stati contattati, si sono iscritti al servizio proposto oppure no (valore del campo Response).

La lista completa degli attributi del dataset a disposizione (UsersSmall.xls) è riportata di seguito.

(1) Age
(2) Workclass
(3) Education record
(4) Marital status
(5) Occupation
(6) Relationship
(7) Race
(8) Sex
(9) Hours per week
(10) Native country
(11) Response.

Dati testuali

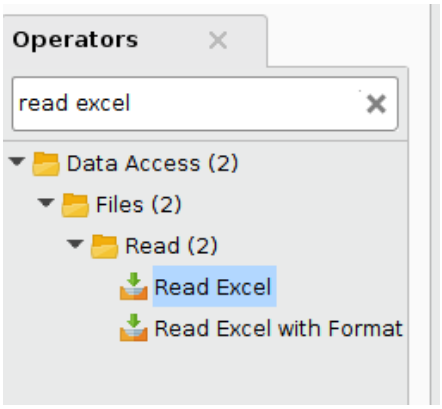
a) Il dataset denominati ObamaNews (ObamaNews.zip) è disponibile sul sito del corso (http://dbdmg.polito.it/wordpress/teaching/business-intelligence/). Esso contiene una collezione di news scaricate mediante il servizio Google News. La collezione rappresenta l'insieme delle prime 10 news (pagine contenenti notizie) restituite da Google News a fronte della specifica della parola chiave *Obama*.
b) Collezione di Tweet

Clustering di dati strutturati

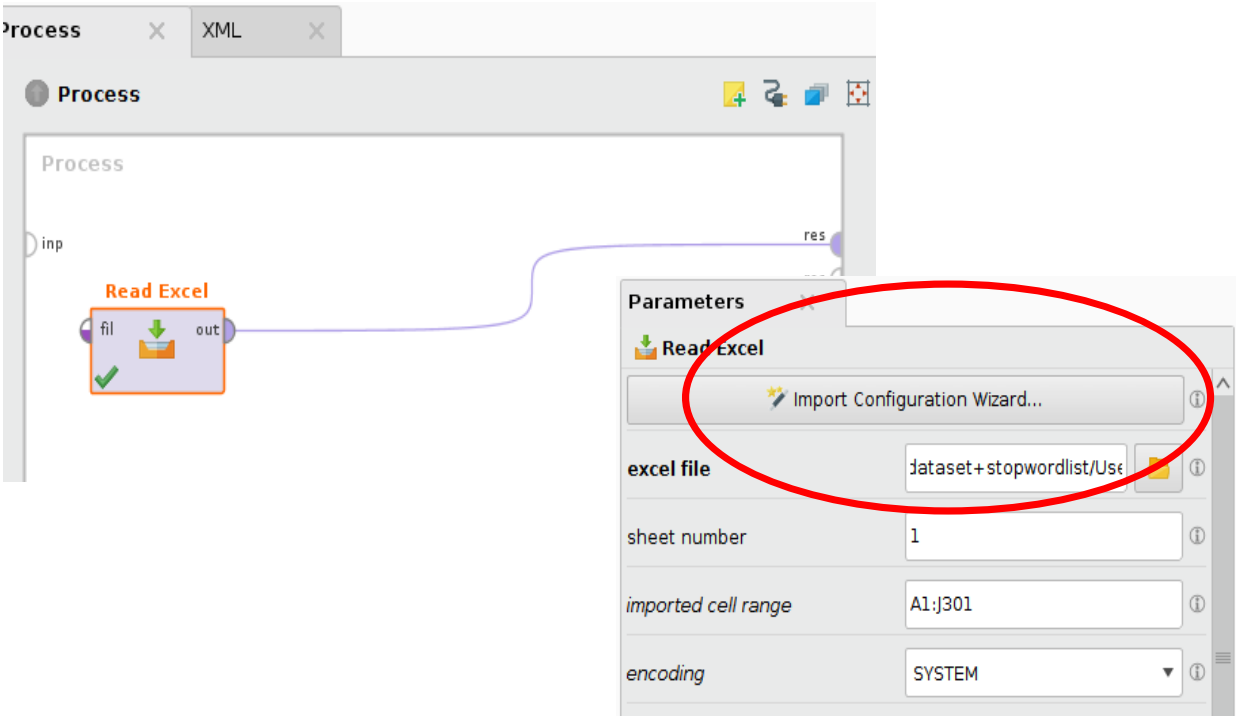
L'obiettivo dell'analisi è raggruppare le persone in gruppi omogenei, tali che persone appartenenti al medesimo gruppo abbiano caratteristiche simili mentre persone appartenenti a gruppi diversi siano dissimili. I gruppi possono rappresentare segmenti di clientela verso cui mirare specifiche promozioni o campagne pubblicitarie.

Obiettivo 1 – Valutazione soggettiva dei cluster generati

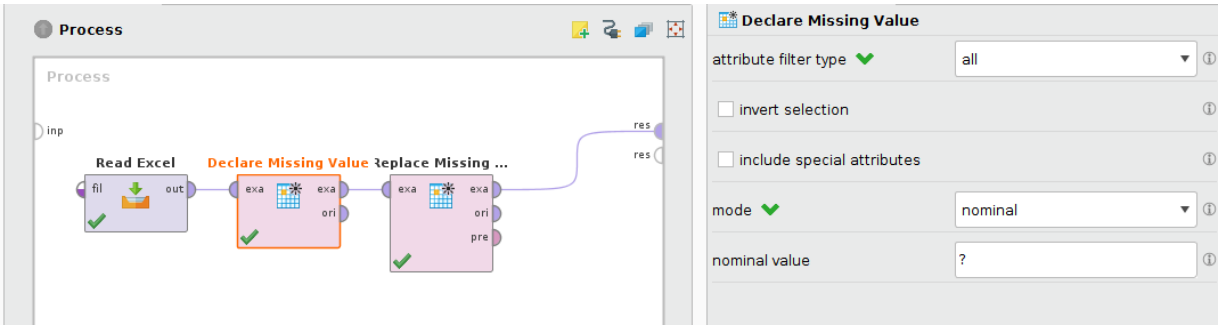
- Nel pannello **Operators** cercare l'operatore **Read Excel** e trascinarlo nello spazio di lavoro.



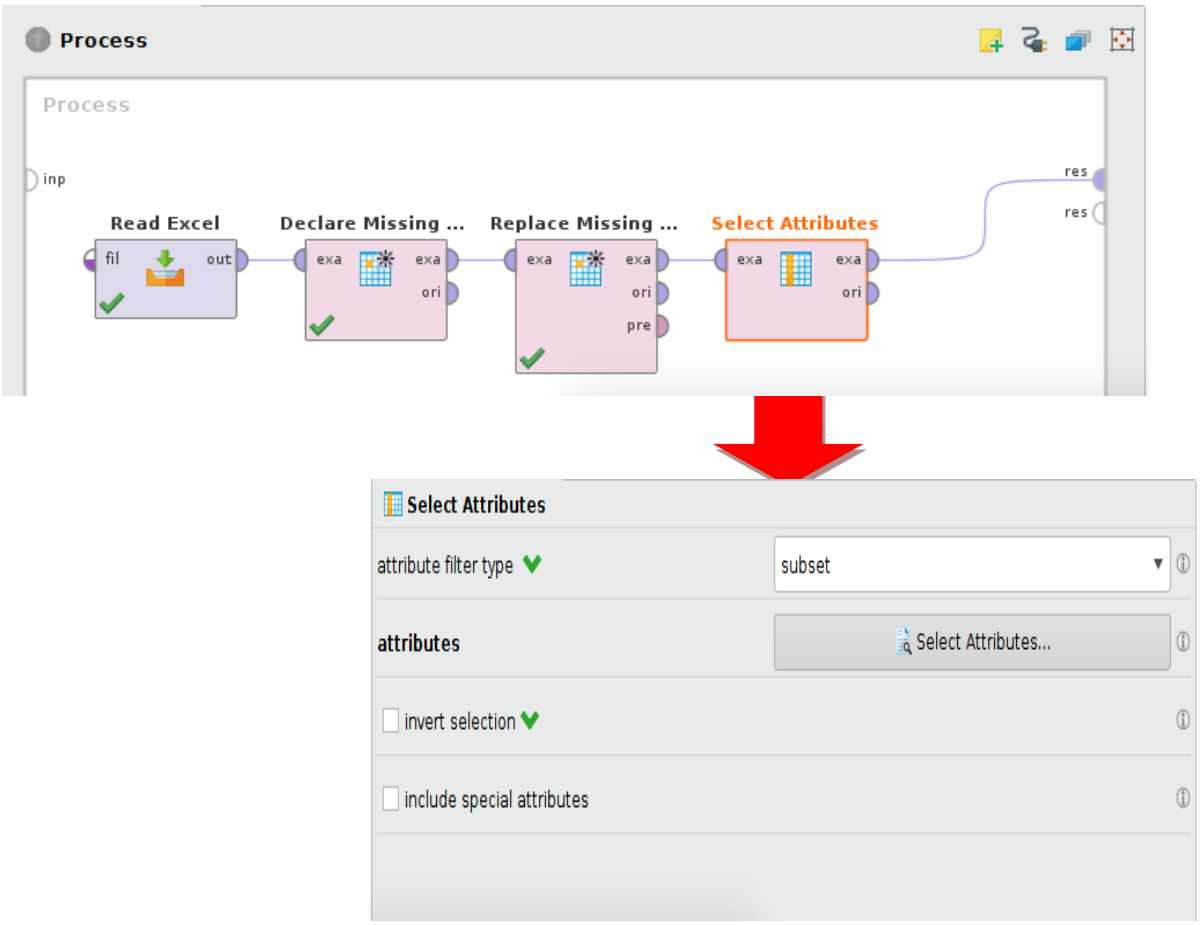
- Importare il dataset ***UsersSmall.xls*** utilizzando la procedura guidata ***Import Configuration Wizard***.

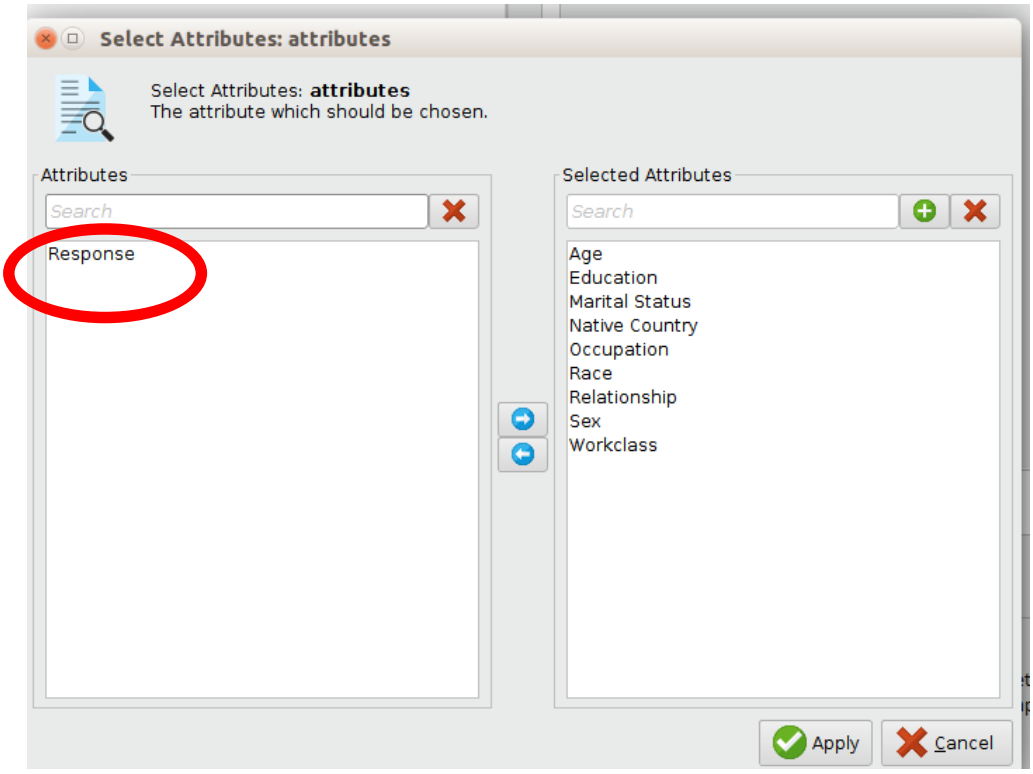


- Dichiarare per tutti gli attributi il '**?**' come valore NULL attraverso l'operatore **Declare Missing Value**.
- Sostituire i valori nulli dichiarati al punto precedente con il valore più frequente usando l'operatore **Replace Missing Values**.

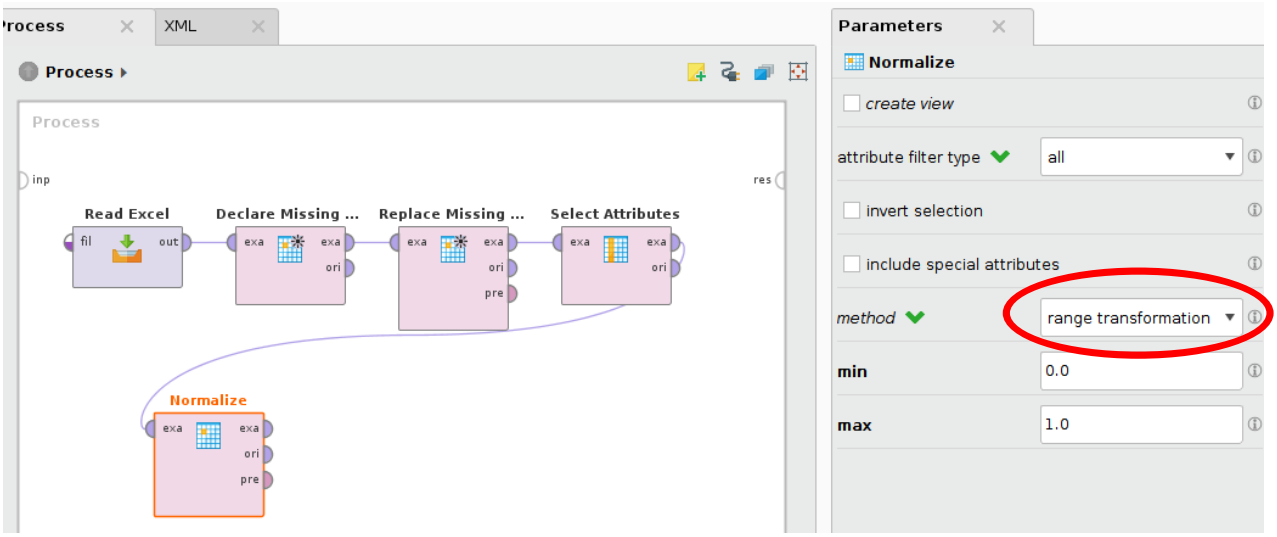


- Escludere l'attributo ***Response*** dall'analisi usando l'operatore ***Select Attributes***.

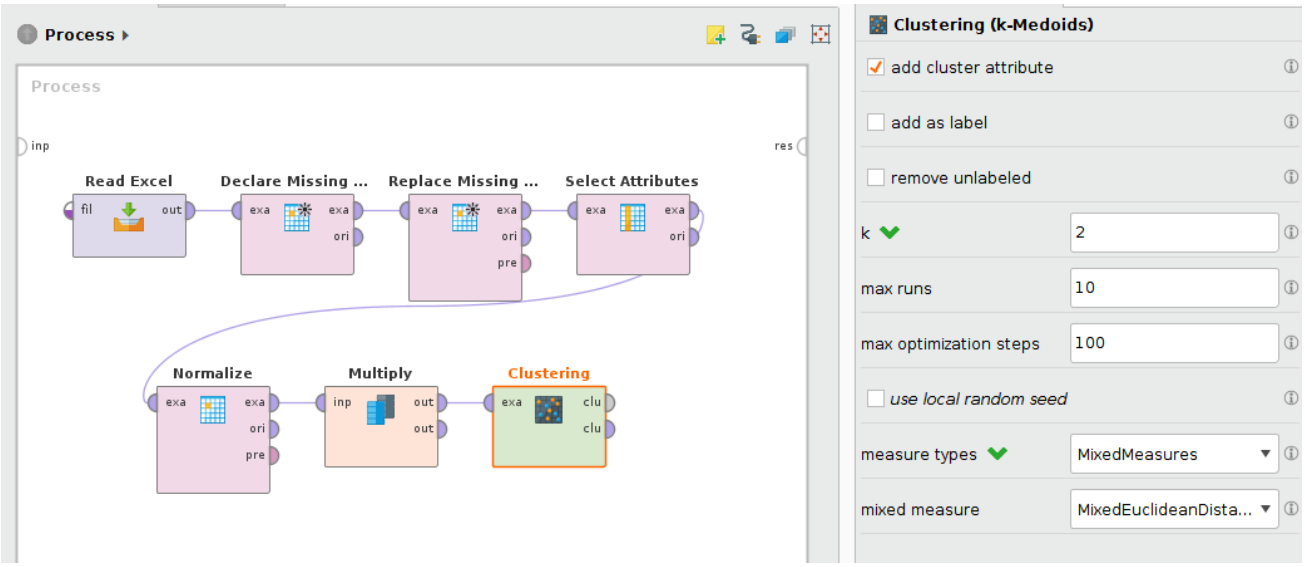




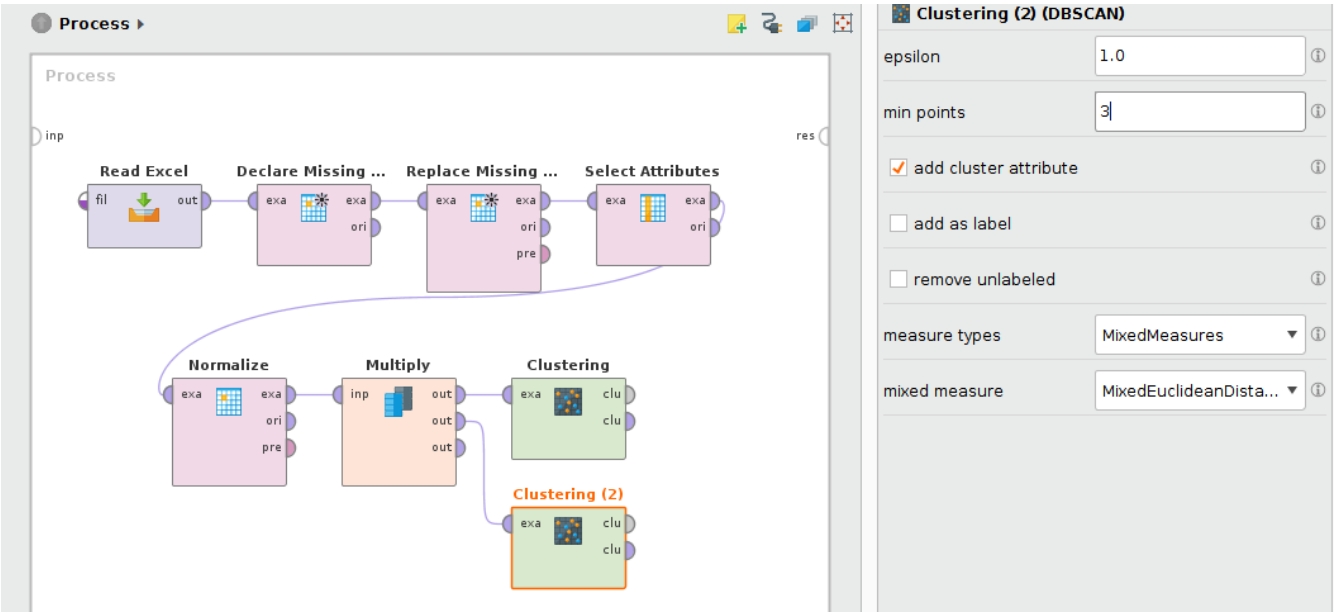
- **Normalizzare** i valori degli attributi numerici indicando come intervallo di valori [0-1] utilizzando l'operatore **Normalize**. L'unico attributo che verrà normalizzato è l'attributo età.



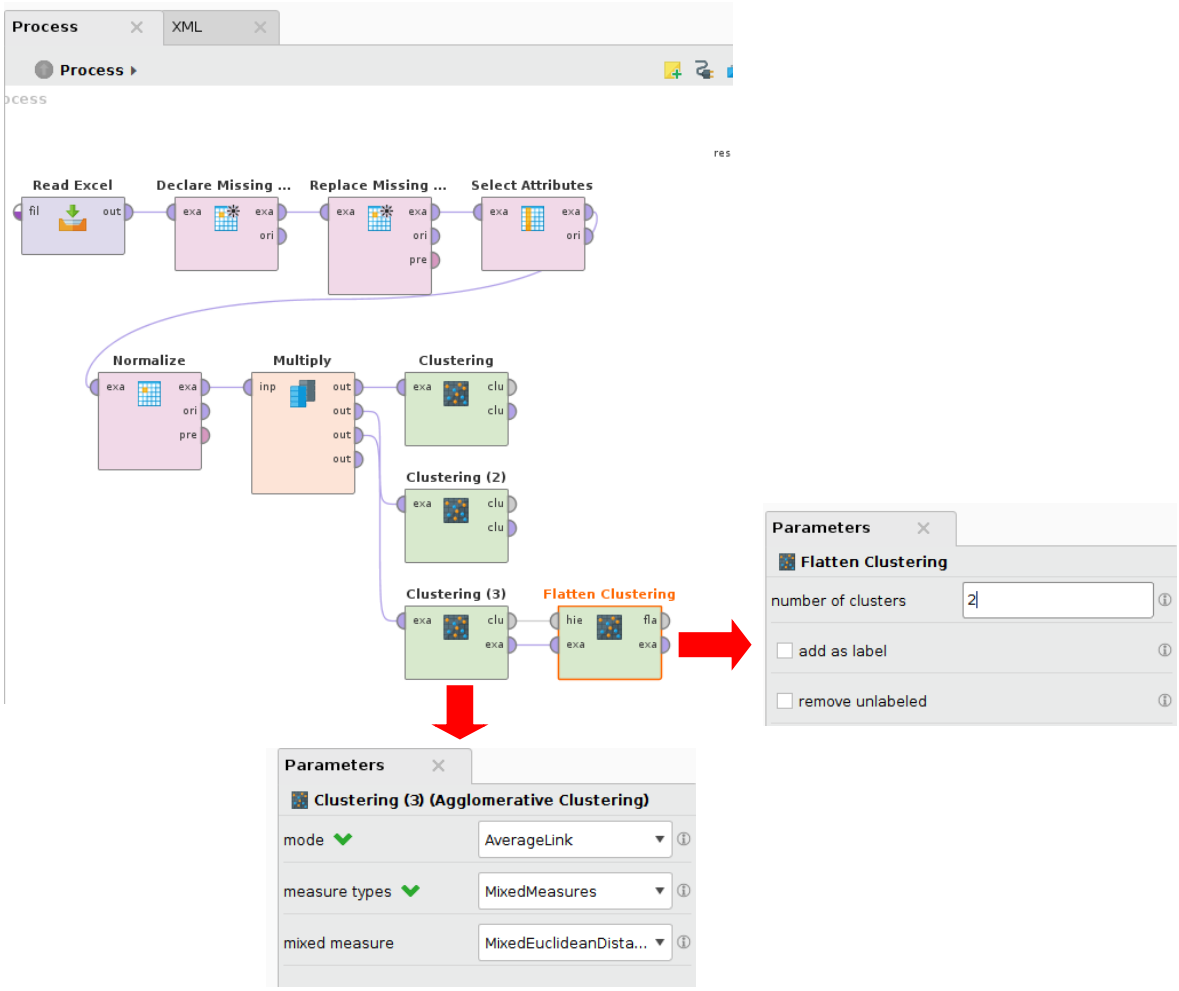
- Quando avete bisogno di utilizzare lo stesso input per diversi algoritmi, utilizzate l'operatore **Multiply**. Nei prossimi step verranno comparati diversi algoritmi di clustering.
- Applicare l'algoritmo di clustering ***k-Medoids*** (quali differenze ha rispetto al K-means?) settando a K=2 il numero di cluster. Esegui il processo e analizza i risultati. Come sono distribuiti i due cluster trovati?



- Applicare l'algoritmo di clustering DBScan settando min points = 3. Esegui il processo e analizza i risultati.

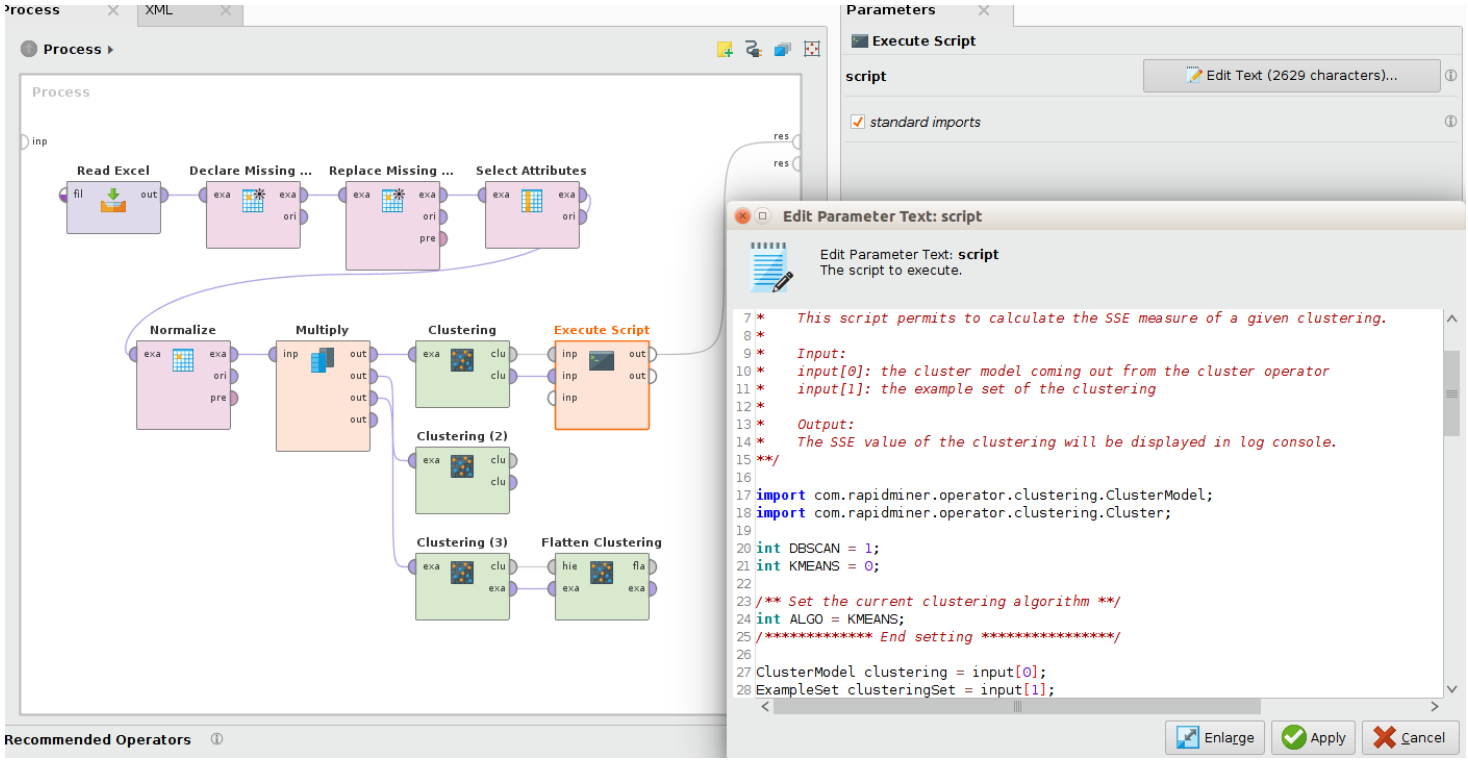


- Applicare l'algoritmo di clustering *Agglomerative (Agglomerative Clustering)*. Selezionare due cluster dal risultato dell'algoritmo di clustering *Agglomerative* utilizzando l'operatore *Flatten Clustering*. Esegui il processo e confrontare il risultato ottenuto con quello prodotto dall'algoritmo *k-Medoids* (numero di cluster k=2) svolto precedentemente. Come sono distribuiti gli elementi per cluster?

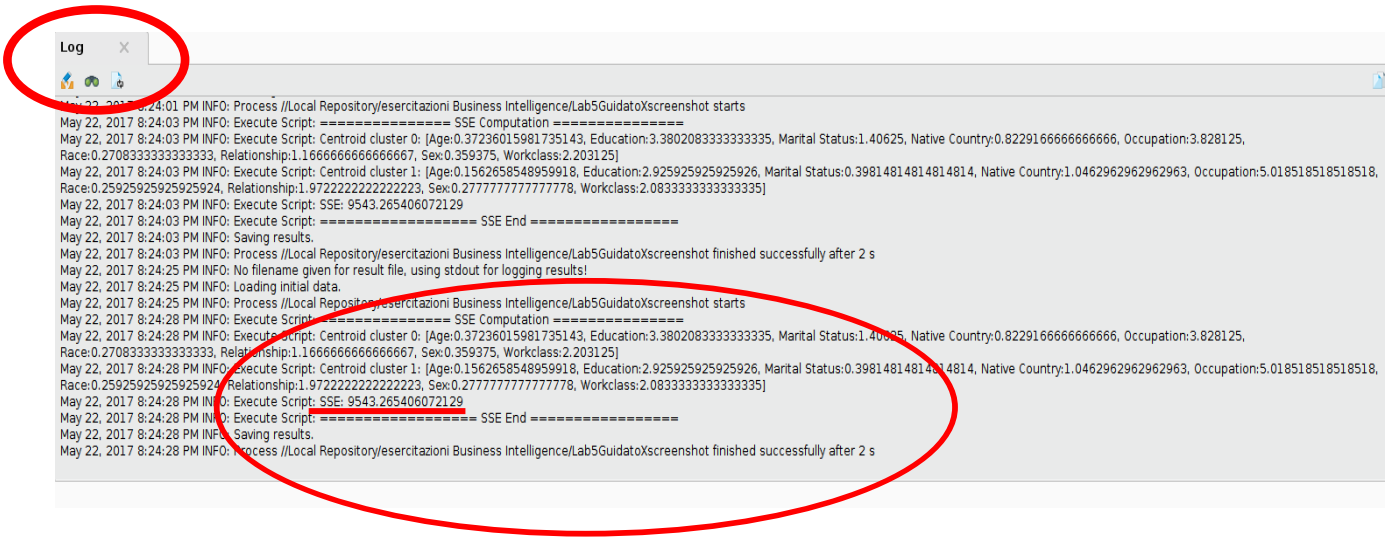


Obiettivo 2 – Valutazione oggettiva dei cluster generati per gli algoritmi K-Medoids e DBSCAN

- Calcolare l'**SSE** (Sum of Squared Errors) dei cluster generate con l'algoritmo *K-Means*. Per il calcolo usare lo script **sse.script** disponibile sul sito del corso; copia e incolla il codice dello script all'interno dell'opportuna textbox dell'operatore ***Execute Script***. *(NB. Settare all'interno del codice il corretto algortitmo (KMEANS o DBSCAN)).*

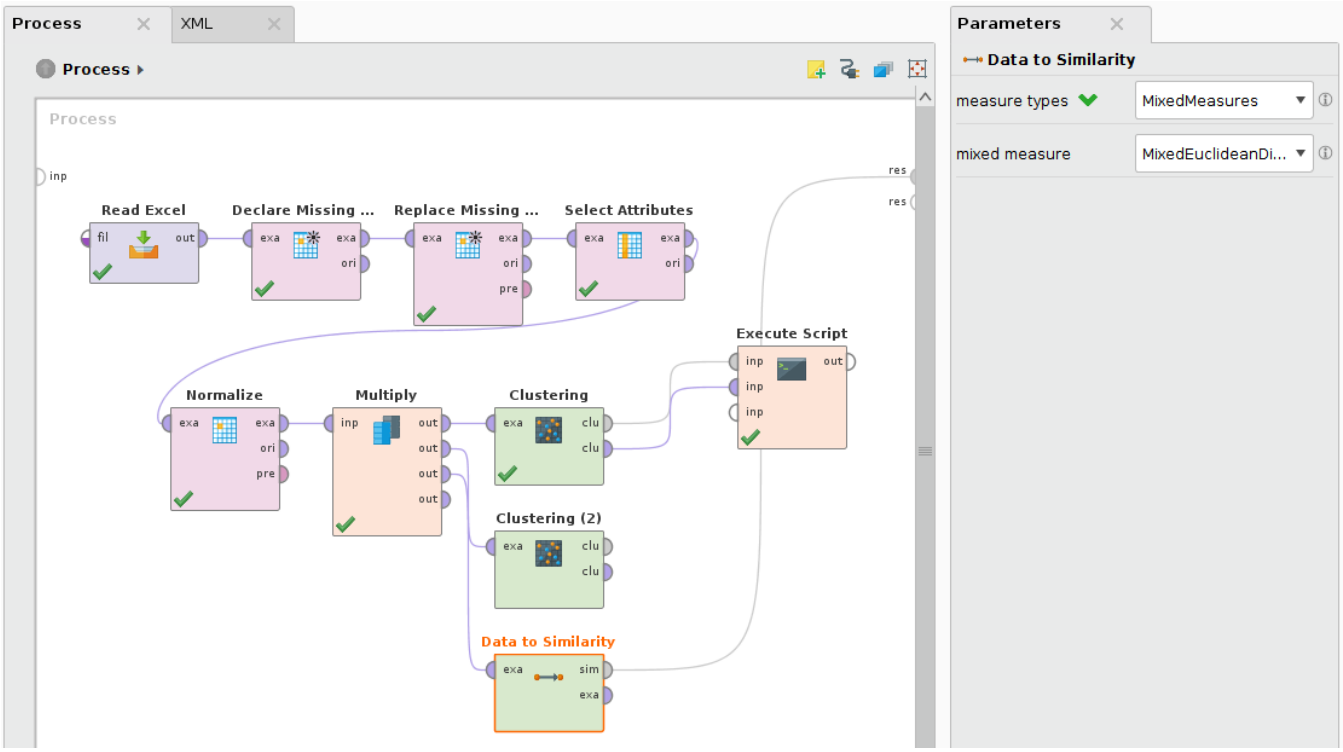


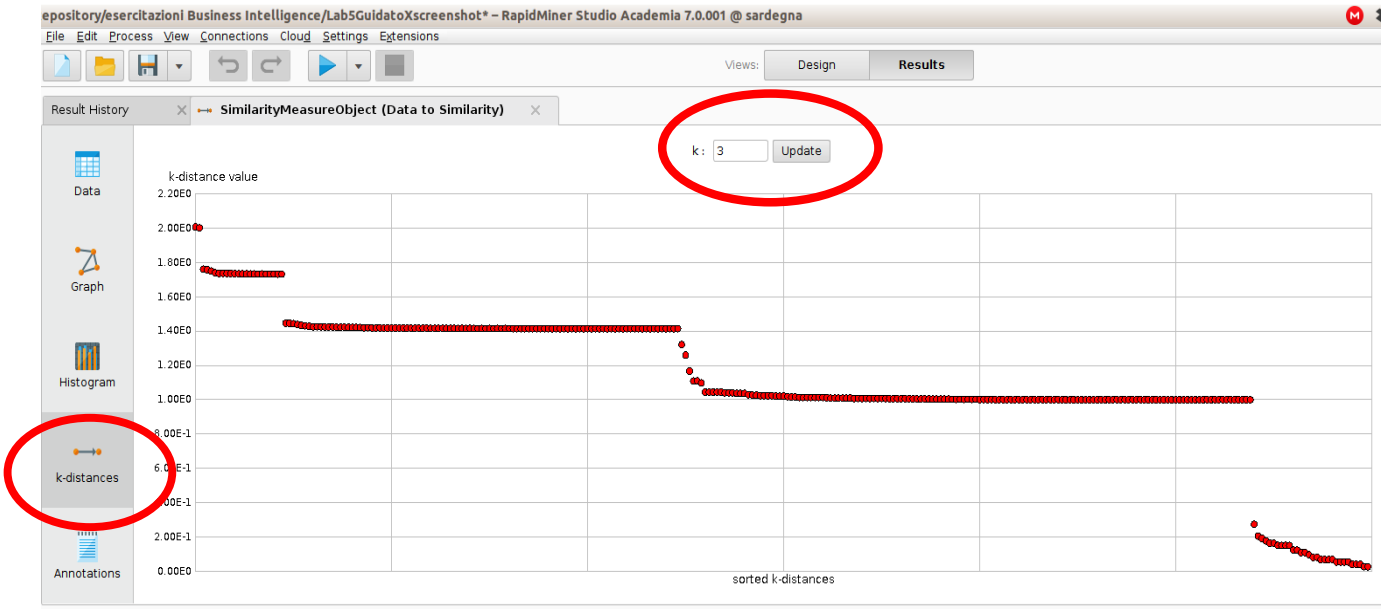
- Eseguite il programma. Il valore di SSE viene riportato nel tab **Log** della pagina dei risultati.



- Rieseguire il processo di valutazione precedente per **differenti valori di K** per l'algoritmo **K-medoids**. Come scegliere il valore ottimale di K? (Suggerimento: utilizza il metodo del ginocchio o del gomito visto durante la spiegazione del clustering in classe).

- **Eliminare** il ramo del Clustering agglomerativo (o disabilitarlo con il tasto destro).
- Per l'algoritmo di clustering DBScan utilizzare il **k-distances plot** per settare correttamente i parametri. Come funziona il k-distances plot? Discutere la configurazione del parametro ***Epsilon*** e il valore di ***MinPts*** dell'algoritmo DBScan.



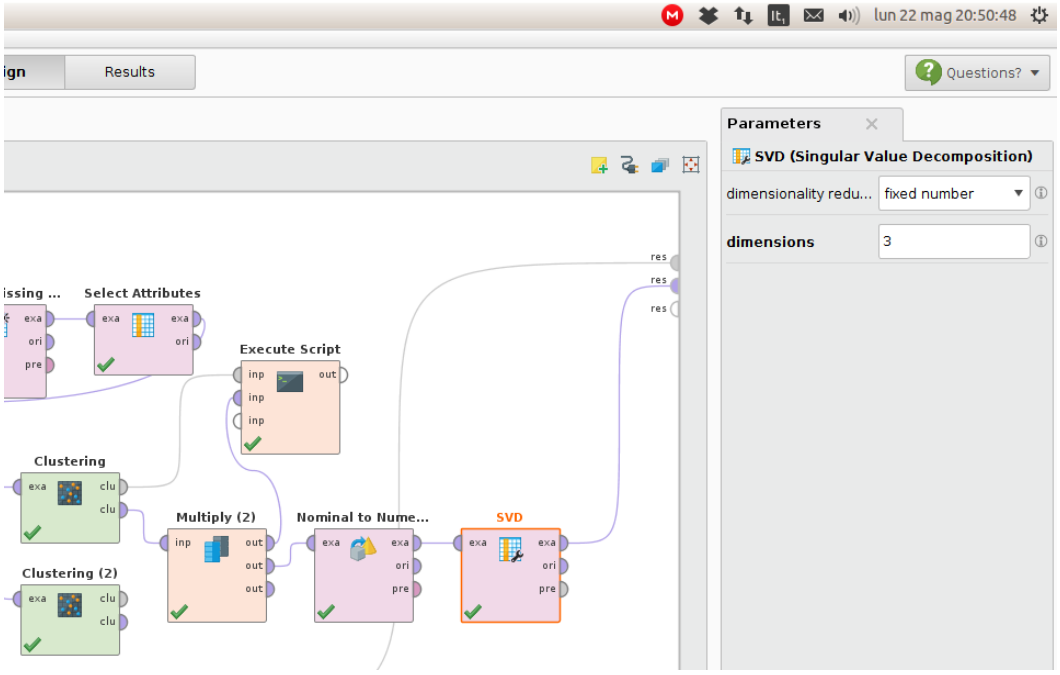


Suggerimento: Aggiorna i diversi valori di k nella pagina dei risultati del Data to Similarity. Quando noti che la curva non varia rispetto al valore precedente, taglia la curva nel punto in cui crea un gomito. Quel valore rappresenterà il raggio **Epsilon**, mentre il k settato rappresenterà il numero di **Min Points**.

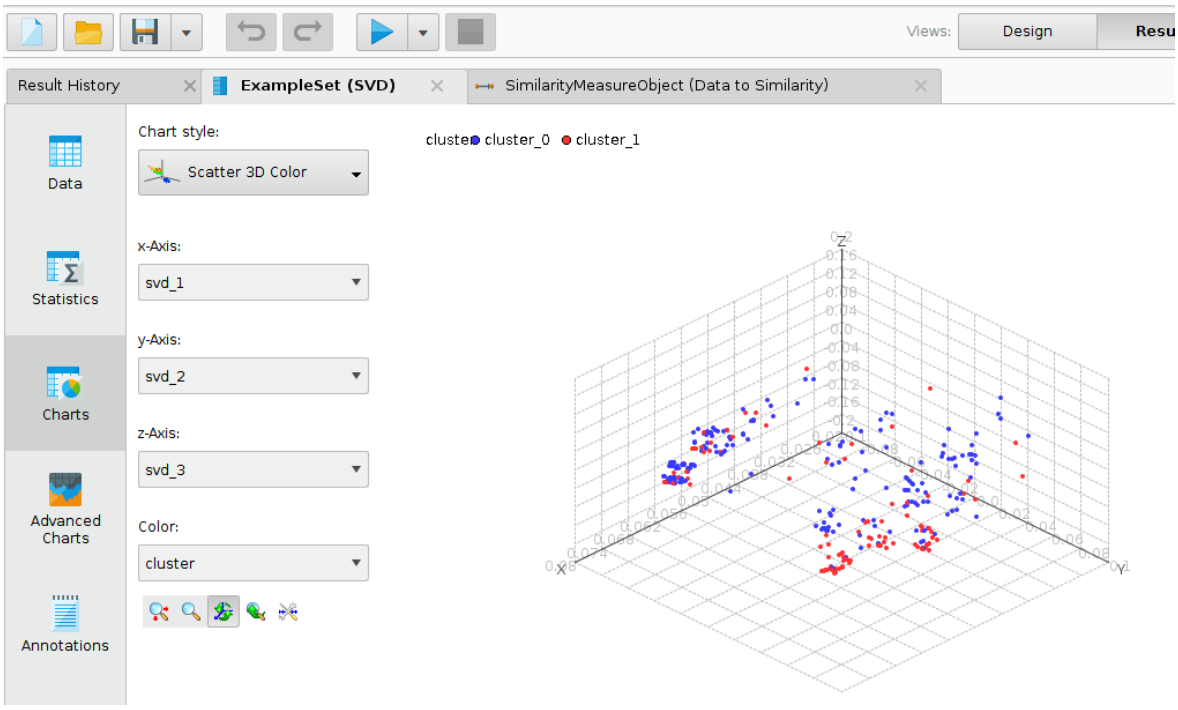
Obiettivo 3 – Visualizzazione/validazione del risultato di un processo di clustering tramite l'uso di tecniche di riduzione delle dimensioni dei dati - SVD (Singular Value Decomposition)

Analizzare la qualità del clustering generato mediante una tecnica di riduzione della dimensionalità dei dati, nota come ***Singular Value Decomposition*** (SVD). SVD permette di proiettare dei dati a N-dimensioni in uno spazio a K-dimensioni, con K scelto dall'utente e minore di N.

Applicate l'operatore *SVD (Singular Value Decomposition)* sul dataset generato dal processo di clustering realizzato al passo precedente. Eseguire il processo impostando K=3 e visualizzare su un grafico di tipo scatter i dati rispetto alle tre dimensioni individuate dall'operatore SVD. Usare l'attributo **cluster** come attributo per la **scelta dei colori dei punti**. I cluster sono ben definiti?



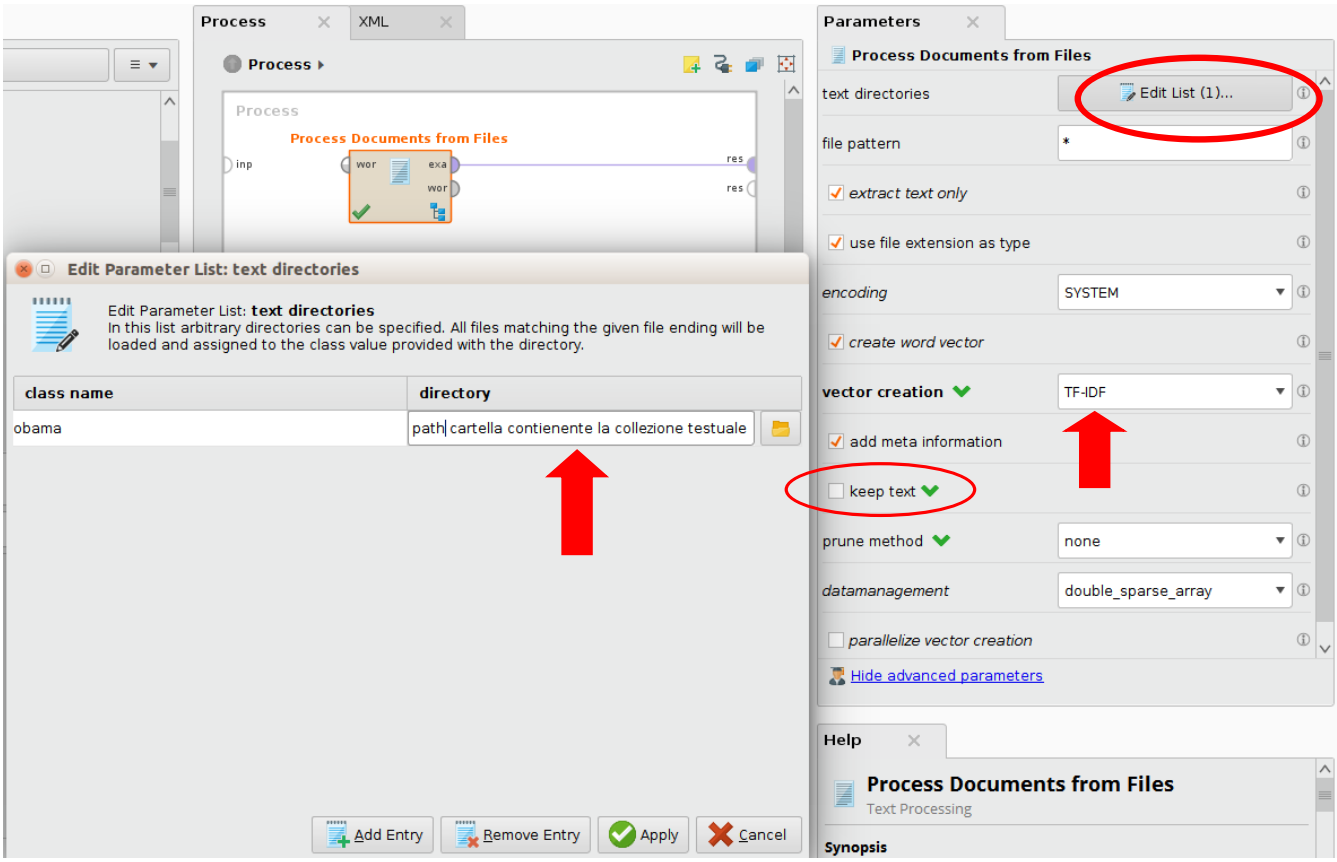
Moltiplica l'uscita del clustering K-medoids e utilizza l'operatore **Nominal to Numerical**. Collegalo all'operatore **SVD** e esegui il programma. Salva il processo sul tuo computer.



Clustering di dati testuali

a) Nella seconda parte di questa esercitazione prevede l'analisi attraverso l'algoritmo K-Means della collezione di frasi denominata *ObamaNews*. Scompatta la cartella presente sul sito del corso ed esegui i passi seguenti.

- Trasforma la collezione di documenti nella matrice **document*term**. Per fare questo passaggio utilizza l'operatore **Process Documents from Files**.



Il **Tf-IDF** (*Term frequency–Inverse Document Frequency*) è una funzione nota nel text mining utilizzata per misurare l'importanza di un termine rispetto ad una collezione di documenti. Il Tf-IDF aumenta **proporzionalmente** al numero di volte che il termine è contenuto nel documento, ma cresce in maniera **inversamente proporzionale** con la frequenza del termine all'interno della collezione. In questo modo si possono penalizzare le parole molto frequenti che non danno rilevanza alla collezione e dare più importanza ai termini che in generale sono poco frequenti ma più rilevanti per l'analisi.

$$\mathrm{tfidf}_{i,j} = \mathrm{tf}_{i,j} \times \log\left(\frac{\mathrm{N}}{\mathrm{df}_i}\right)$$

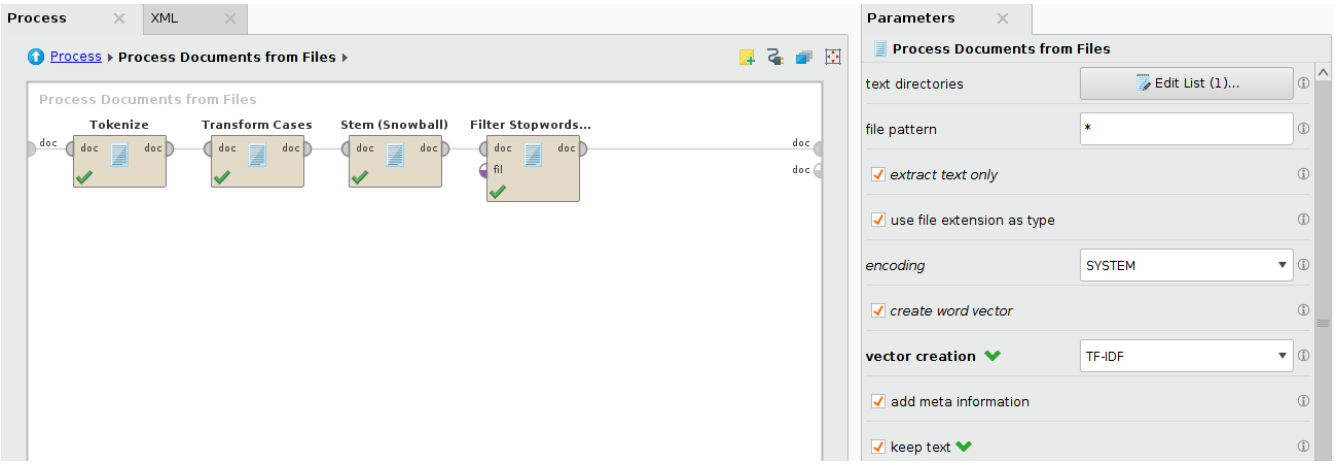
$\mathrm{tf}_{i,j}$ = total number of occurences of i in j

df_i = total number of documents (speeches) containing i

N = total number of documents (speeches)

Se volete avere l'informazione del testo all'interno dei risultati, spuntate la voce **Keep Text** nel pannello dei parametri dell'operatore **Process Documents from Files**.

- Applicare i passi di pre-processing visti nell'esercitazione precedente. Doppio click sull'operatore **Process Documents from Files**. Verrà aperto un sottoprocesso. Utilizzare i seguenti blocchi:

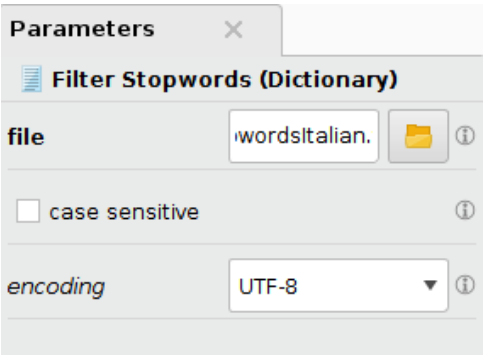


Il blocchetto **Tokenize**: splitta ogni documento della collezione Obama in un vettore di parole. L'ordine delle parole non sarà più rispettato. Secondo te ha importanza ai fini dell'analisi? (Settare il parametro non letters).

Il blocchetto **Transform Cases**: Trasforma il testo in maiuscolo o minuscolo.

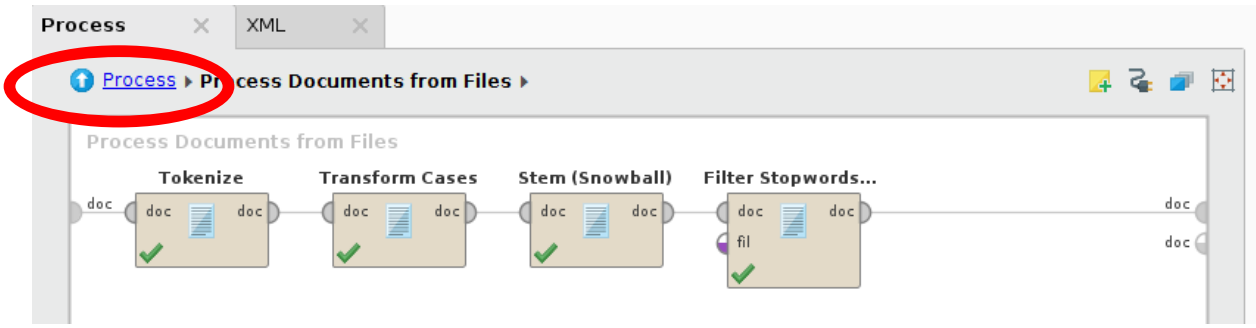
Il blocchetto **Stem (Snowball)**: Riduce le parole alla propria radice. La radice è quell'elemento linguistico irriducibile (non ulteriormente suddivisibile) che esprime il significato principale della parola. (Utilizzare la lingua italiana).

Il blocchetto **Filter Stopwords (Dictionary)**: Permette di eliminare le parole definite Stopword, parole che non hanno un particolare significato se isolate dal testo e quindi vengono ignorate dai programmi. Sono parole poco significative perché possono essere usate spesso all' interno delle frasi. Ad esempio articoli, congiunzioni e preposizioni non caratterizzano il significato di un testo, possono essere eliminate a monte di una analisi text mining. Carica il file **stopwordsItalian.txt** presente sul sito del corso.

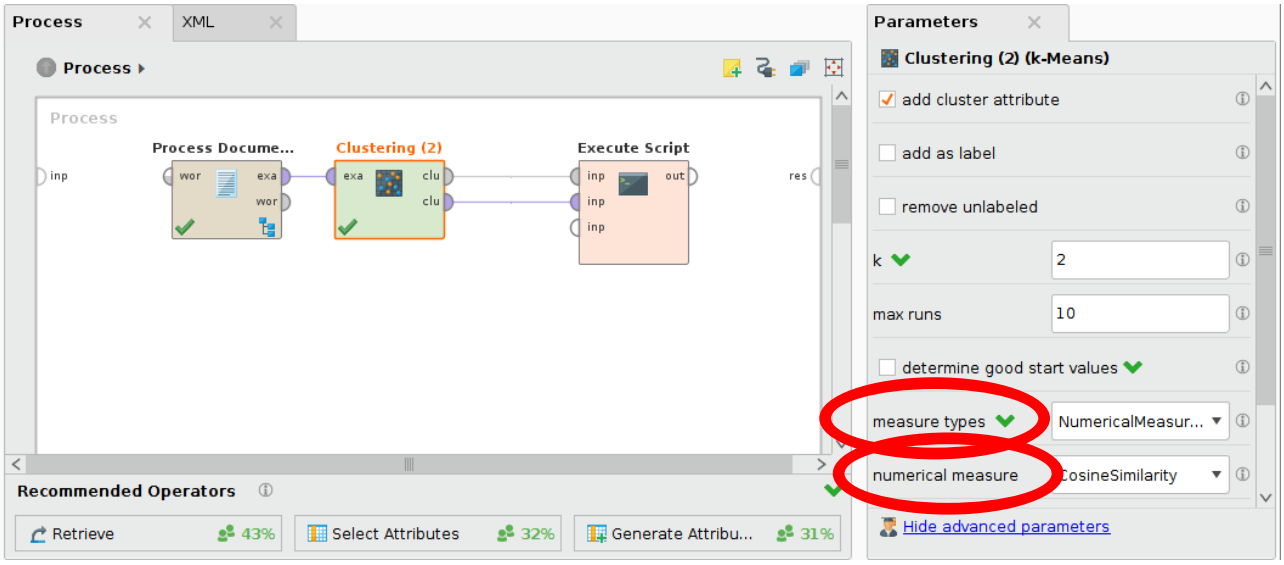


Utilizzare la codifica UTF-8 per il file delle stopword.

Torna al processo iniziale cliccando sulla freccia blu sotto la voce Process.



- Utilizzare l'algoritmo di **K-Means** per dividere la collezione in gruppi omogenei di documenti che parlano di uno stesso topic.



Per i dati testuali la misura per calcolare la distanza tra punti (in questo caso tra due documenti) è la **Cosine Similarity**. Come viene calcolata? Rispetto a cosa si differenzia dalla norma Euclidea?

- Riesegui il K-means diverse volte per cercare il parametro K migliore. Utilizza l'SSE come metrica di valutazione (guarda la prima parte dell'esercitazione).
- Visualizza i cluster trovati attraverso la tecnica SVD.
- Ha senso calcolare la correlazione prima di eseguire il K-means? Perché?

Facoltativo. Prova a rieseguire l'analisi utilizzando l'algoritmo **DBScan**.

b) L'obiettivo dell'esercitazione è:

- **Applicare l'algoritmo di clustering K-means disponibile in RapidMiner per suddividere un dataset di tweet reali in gruppi omogenei di tweet riguardanti uno stesso argomento e caratterizzarli.**

Dati testuali – Seconda parte

Il dataset denominato dataset (dataset.csv) contiene 200 tweet relativi a programmi televisivi italiani reali.

La lista completa degli attributi dei dataset a disposizione è riportata di seguito.

1. post_id: identificativo univoco del tweet
2. username: username della persona che ha inviato il tweet
3. hash_tag: hash_tag della trasmissione
4. sent_time: timestamp del tweet
5. tweet: testo del tweet
6. user_id: identificativo dell'utente
7. source: sorgente d'invio
8. is_retweet: è un retweet*
9. is_reply: è un reply**
10. lang: lingua
11. retweet_count: conteggio di retweet
12. reply_count: conteggio di reply
13. latitude: latitudine
14. longitude: longitudine

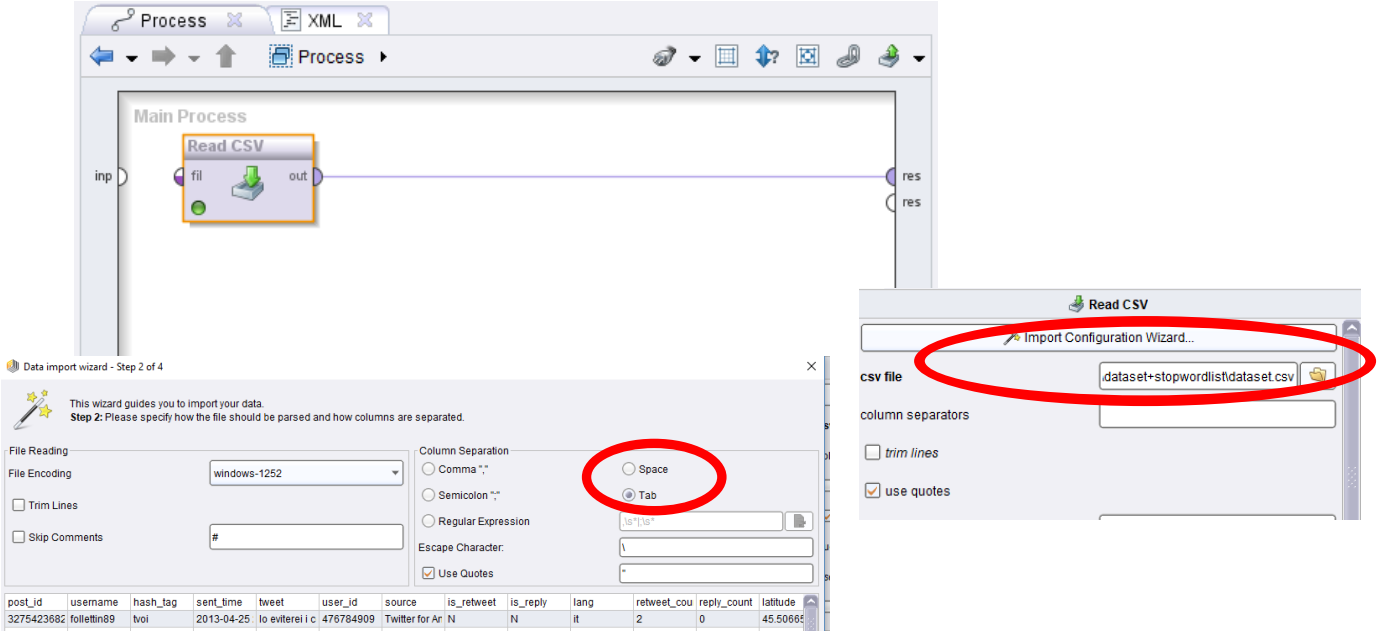
* Il Retweet si verifica quando si inoltra il tweet di un altro utente ai propri follower.

** La reply è la risposta a qualcosa scritto da qualcuno su Twitter.

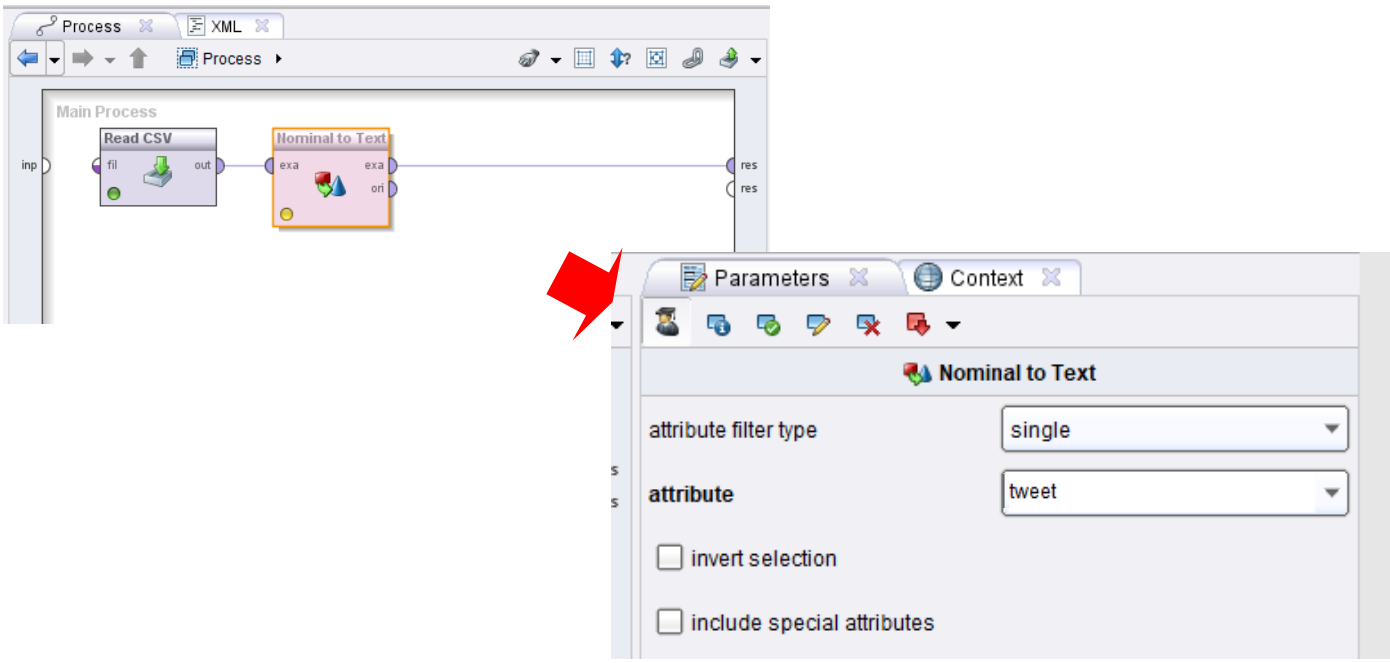
In questo laboratorio vedremo come suddividere i tweet in cluster, ovvero in gruppi di tweet semanticamente correlati. A tale scopo, utilizzeremo l'algoritmo K-Means, un algoritmo di clustering partizionale ben conosciuto in letteratura.

Passi per risolvere il problema e sua implementazione in RapidMiner:

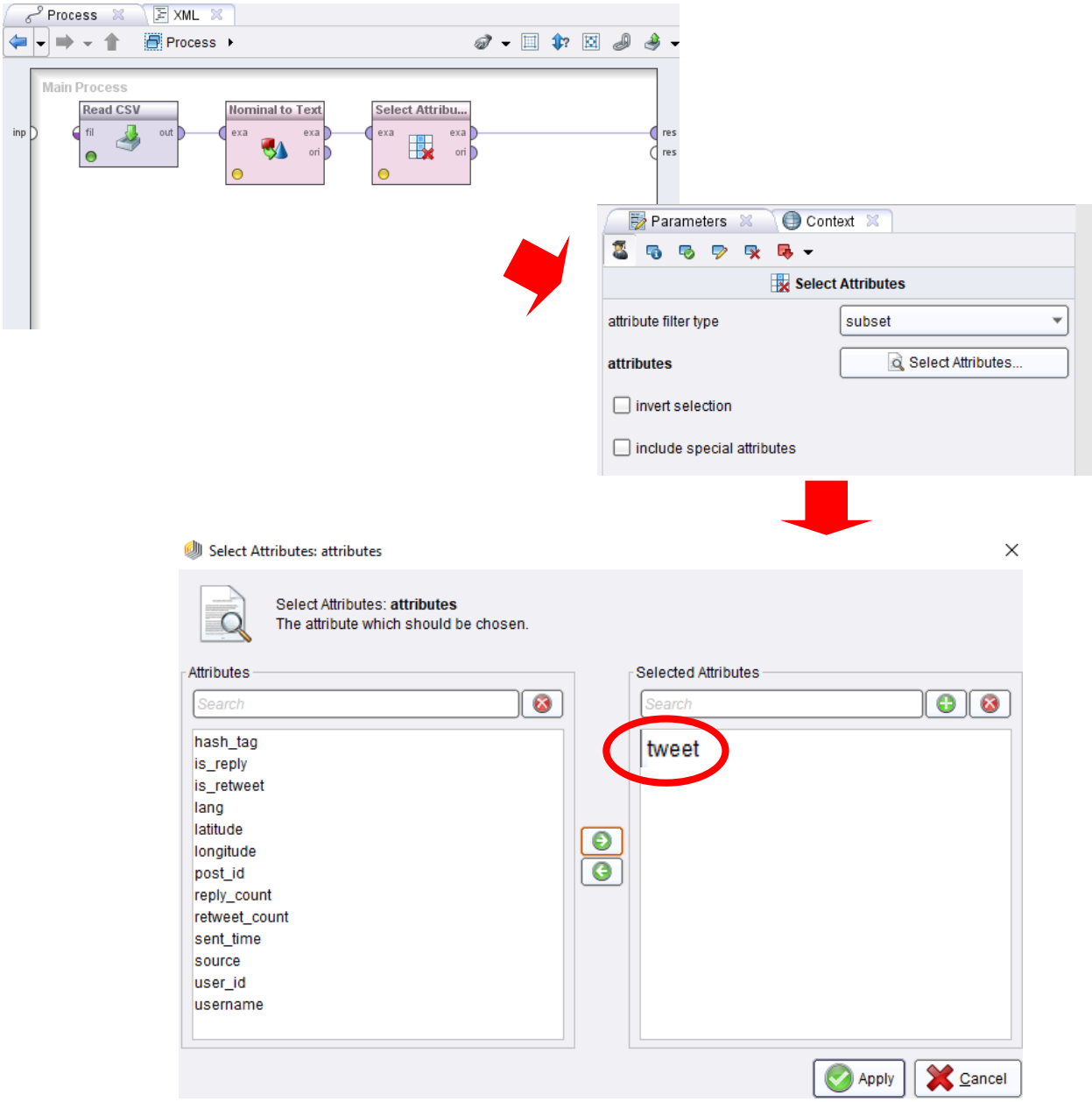
- Come primo passo è necessario creare un processo nuovo in RapidMiner
    - Caricare i dati presenti in dataset.csv usando l'operatore Read CSV. Usare anche in questo caso il Wizard per importare in modo corretto i dati. Utilizzare il come separatore.



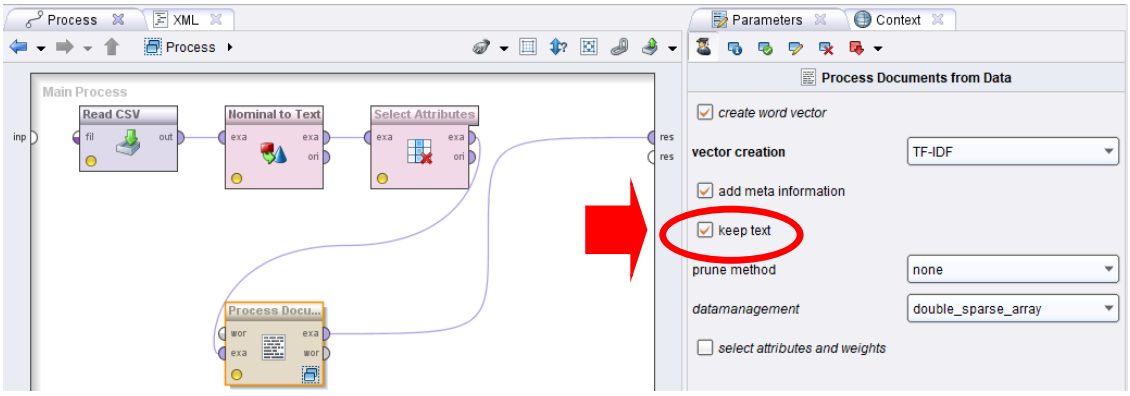
- Specificare qual è l'attributo di testo (attributo da clusterizzare) usando l'operatore Nominal to Text.



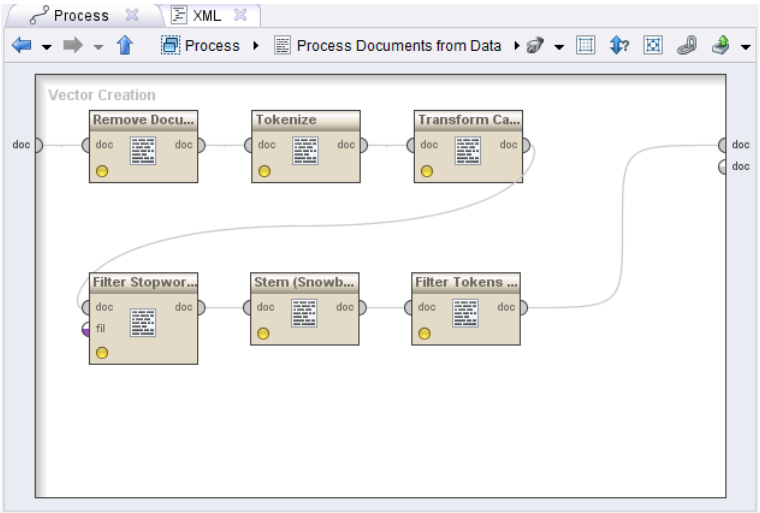
- Selezionare solo l'attributo **tweet** di interesse per la nostra analisi.



- Processiamo il documento come abbiamo fatto durante l'esercitazione delle regole di associazione usando l'operatore **Process Document from Data**



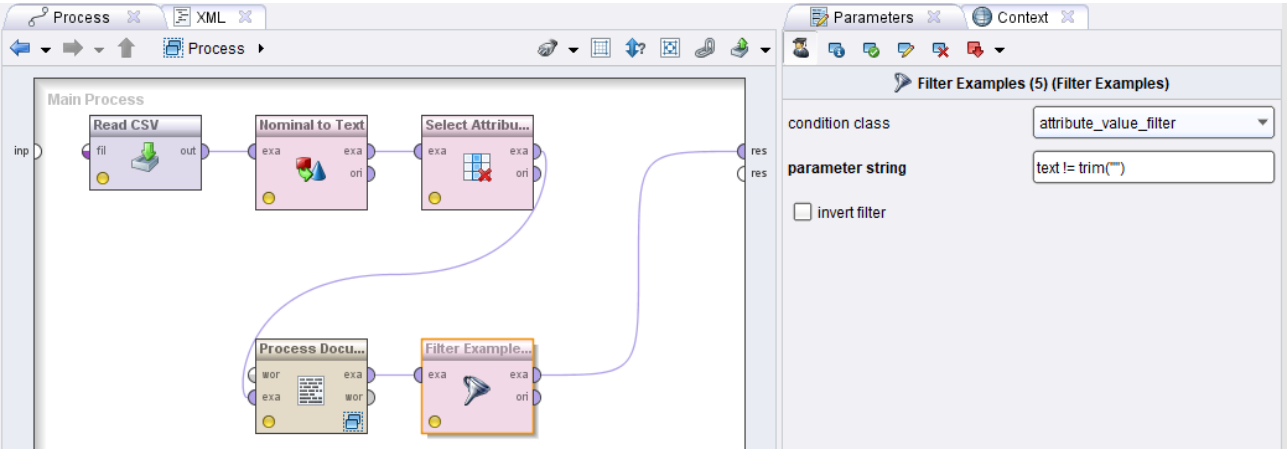
L'operatore *Process Document from Data* ammette un sottoprocesso per poter pulire il dataset e trasformarlo in una tabella chiamata matrice documenti*termini. La tabella avrà una riga per ogni tweet presente nel file csv letto e una colonna per ogni termine presente all'interno della collezione di tweet.



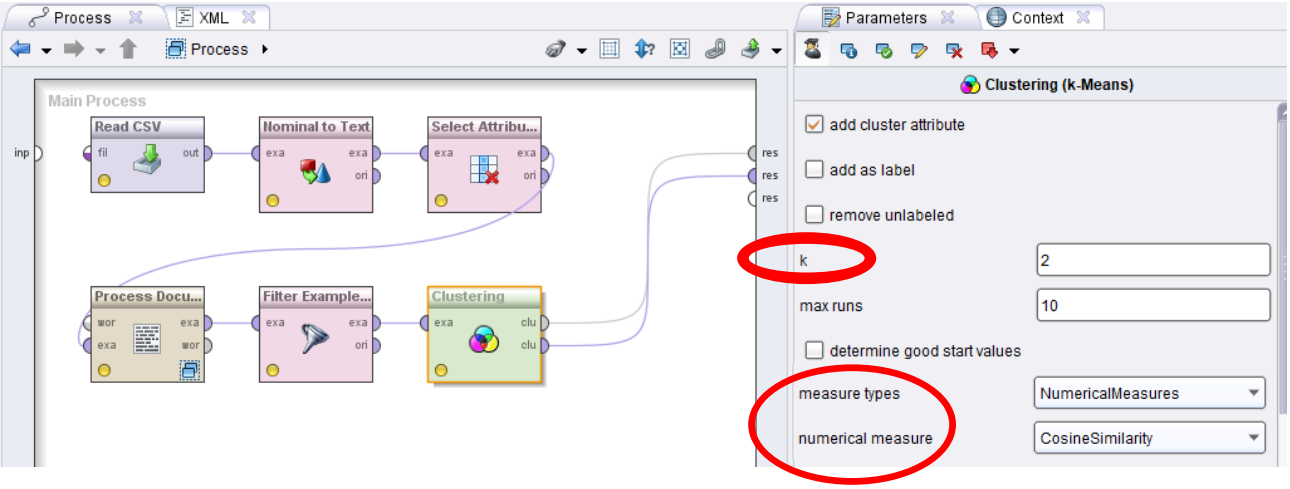
I passi eseguiti sono:

- Remove Document Parts: eliminiamo tutti i caratteri speciali con la seguente regex -> http[://.a-zA-Z-0-9]*
- Tokenize: divide il testo in token, ovvero in parole
- Transform Case: trasforma tutto il testo in minuscolo
- Filter Stopword (Dictionary): Elimina le parole prive di significato nella lingua italiana. *Utilizzare il file **stopword.txt** presente all'interno della cartella dell'esercitazione*.
- Stem (snowball): riduce ogni parola alla propria radice
- Filter Tokens (By Length): eliminiamo parole troppo lunghe (parametro max chars = 15) o troppo corte (parametro min chars = 3).

    - Dopo la pulizia, alcuni tweet potrebbero rimanere vuoti. Utilizziamo l'operatore Filter Examples per rimuovere questi record.



- Applichiamo adesso l'algoritmo di **K-Means**. La scelta del K è un problema molto complesso in letteratura.

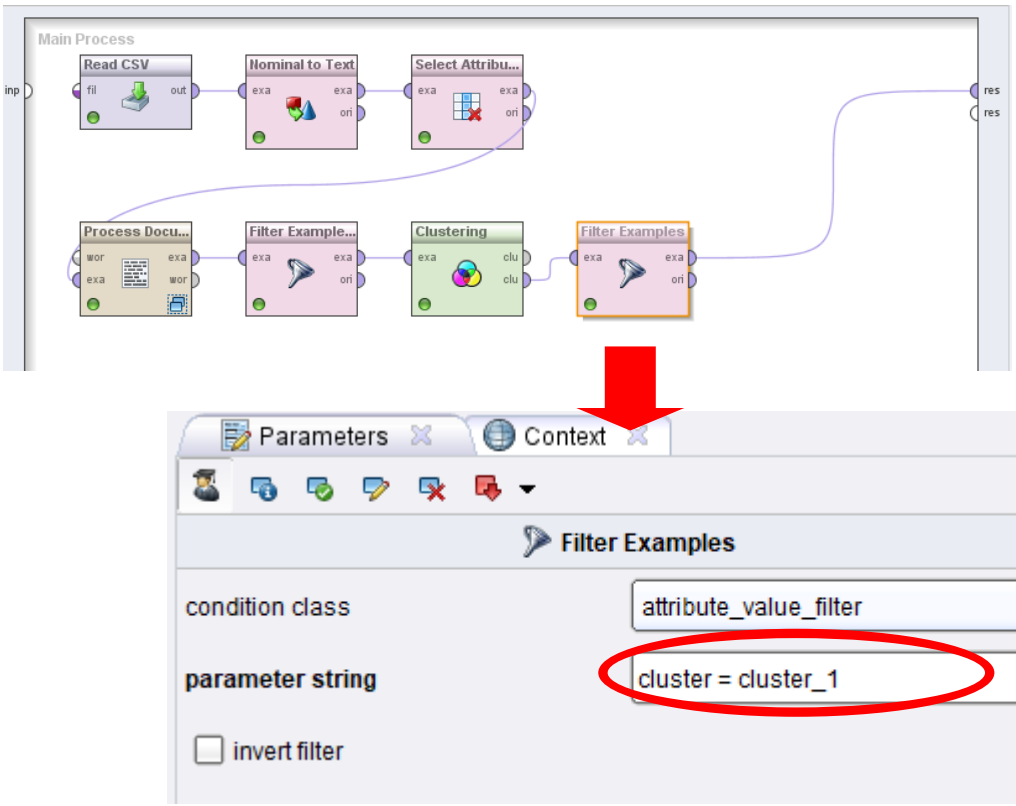


Analizzate i cluster generati dal processo, al variare del parametro K. Provate a variare il parametro K utilizzando K=2, K=5 e K=6. Riuscite a capire di cosa parlano i tweet dei vari cluster?

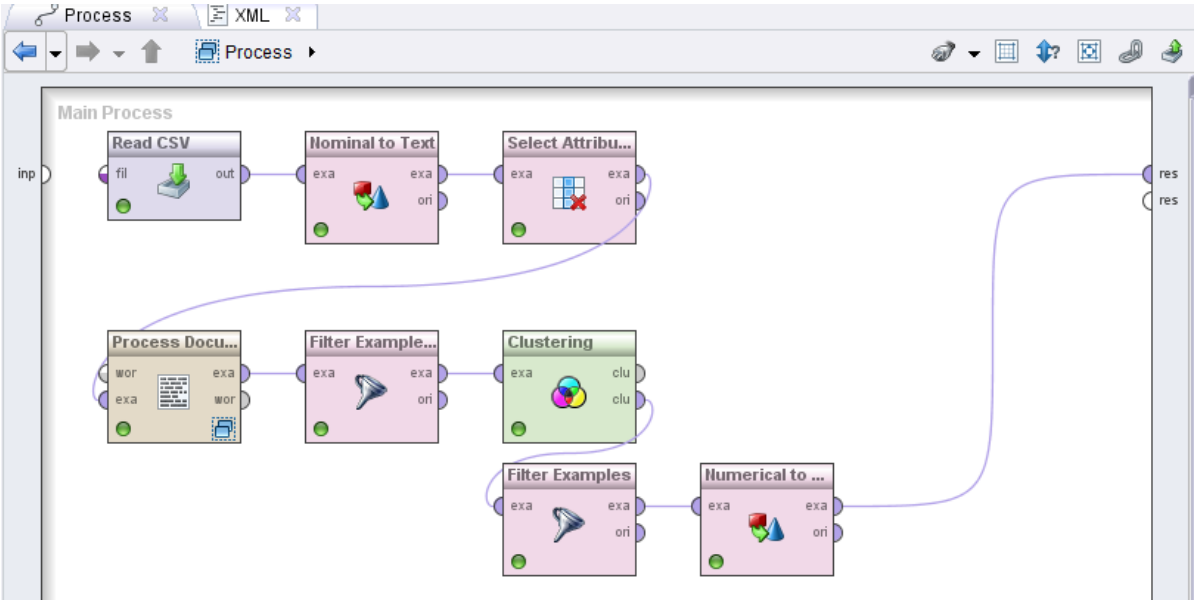
- **Estrazione itemset**.

- Un'altra tecnica di Data Mining utile per leggere il "contenuto" dei cluster è l'**estrazione degli itemsets** più frequenti all'interno di ciascun cluster.

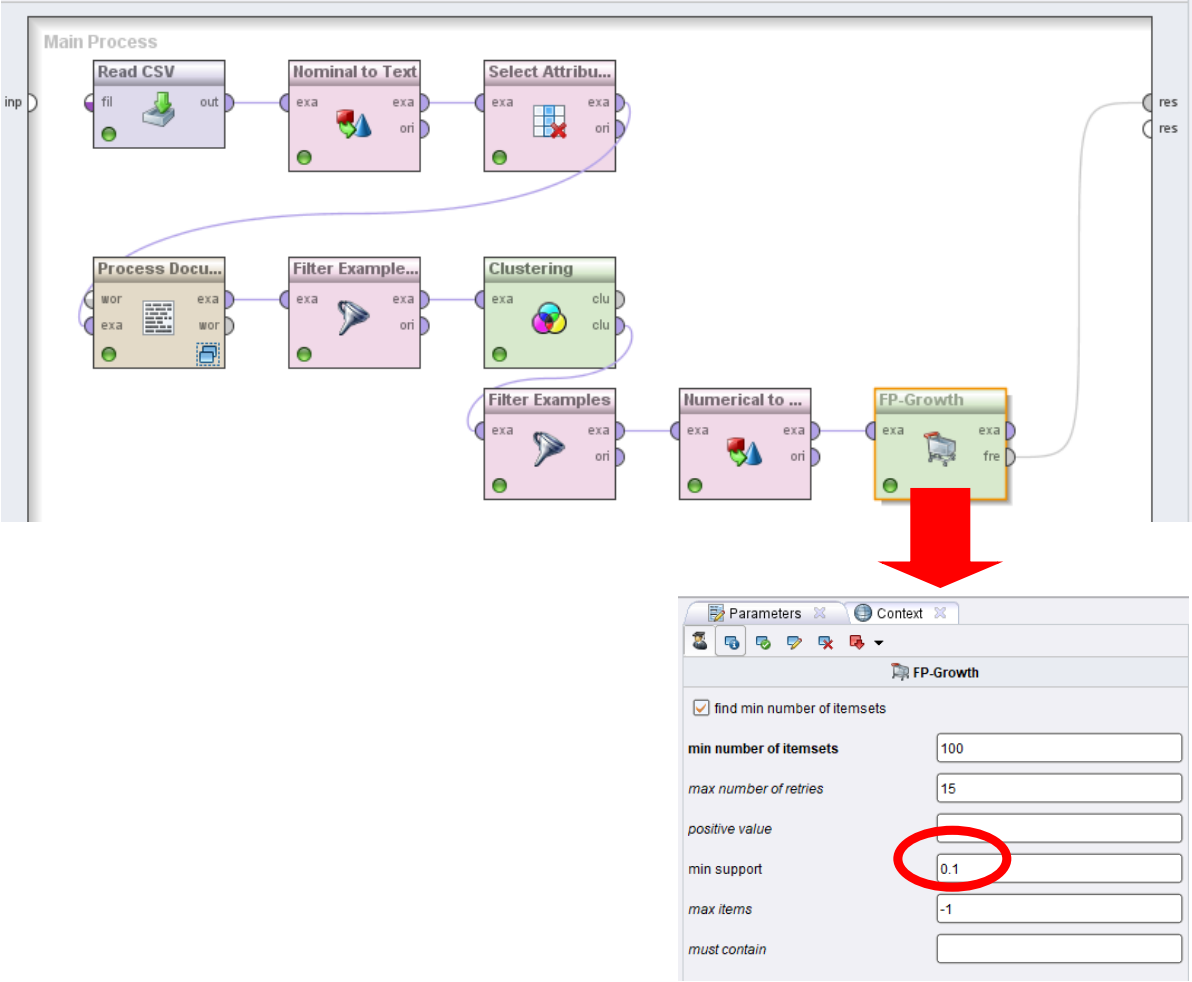
- Rieseguite il processo di clustering con K=2.
- Selezioniamo con l'operatore **Filter Examples** i tweet di un cluster alla volta utilizzando una condizione di selezione.



- Trasformiamo i dati in dati binomiali usando l'operatore **Numerical to Binomial**.



- Applichiamo l'algoritmo **FP-Growth** per estrarre gli item e gli itemset più frequenti per ciascun cluster.



Riuscite a identificare un **argomento** per ciascun cluster? Un K maggiore potrebbe essere interessante da analizzare nel vostro esempio? Le regole potrebbero caratterizzare maggiormente il contenuto infromativo?