Università degli Studi di Modena e Reggio Emilia

Facoltà di Ingegneria "Enzo Ferrari"

Corso di Laurea in Ingegneria Informatica

Tesi di Laurea Triennale

Data preparation e workflow management con KNIME

Relatore:
Prof.ssa Sonia Bergamaschi

Laureanda:
Cristina Ventilati

Correlatore:
Dott. Luca Zecchini

ANNO ACCADEMICO 2020-2021

Indice

INTRODUZIONE

L'obiettivo del presente elaborato è quello di studiare ed analizzare le caratteristiche del software KNIME Analytics Platform, una piattaforma gratuita e open source progettata per supportare processi di analisi dei dati e di data mining. La sua caratteristica principale è la possibilità di modellare graficamente il processo di data science assemblando diversi nodi per costruire un workflow visivo, con l'obiettivo di automatizzarlo.

Nel primo capitolo vengono introdotte le caratteristiche principali della data prepararion, ossia il processo di unione dei dati provenienti da sorgenti differenti, che consiste nell'assimilazione, pulizia, mappatura, trasformazione ed elaborazione dei dati per analizzare e utilizzare le informazioni in modo più efficace.

Nel secondo capitolo viene introdotto il software oggetto di questa tesi, KNIME Analytics Platform, e se ne illustrano le caratteristiche principali.

Negli ultimi due capitoli viene presente una valutazione delle funzionalità del software, in particolare, nel terzo capitolo si considerano le principali funzionalità per la data preparation, individuate e analizzate nel paper "Data Preparation: A survey of Commercial Tools", di Mazhar Hameed e Felix Naumann [2], e si verifica se e come KNIME sia in grado di eseguirle e con quali nodi. Nel quarto capitolo ho analizzato altre funzionalità comunemente usate per i processi di analisi dei dati, in particolare i nodi per la visualizzazione dei dati e i nodi per il controllo di flusso.

Nella conclusione vengono riassunti gli studi effettuati nel presente elaborato.

1. DATA PREPARATION

La preparazione dei dati (*data preparation*) è definita come l'insieme delle operazioni di pre-elaborazione necessarie per trasformare "dati grezzi", ovvero dati eterogenei, non elaborati e disorganizzati in informazioni utilizzabili. La data preparation è fondamentale per i processi di gestione (*data management*) e analisi dei dati (*data analysis*) ed è necessaria per garantire la bontà delle decisioni *data-driven*, che si basano cioè sull'analisi dei dati raccolti.

Un buon processo di preparazione dei dati, infatti, si focalizza su raccolta, pulizia, consolidamento, strutturazione e organizzazione dei dati per un loro utilizzo in applicazioni di Business Intelligence e Advanced Analytics. Ciò permette di ridurre il tempo necessario per la ricerca delle informazioni in quanto i dati saranno più consistenti, validi e senza errori. I "dati grezzi" sono spesso incompleti, inaccurati e inconsistenti poiché presentano valori mancanti, imprecisioni ed errori di varia natura.

Famoso è l'acronimo informatico GIGO, *Garbage In, Garbage Out*, (letteralmente "spazzatura dentro, spazzatura fuori), che implica che se i dati in input non sono di buona qualità, neanche le analisi basate su questi lo saranno.

Negli ultimi decenni lo sviluppo della tecnologia informatica ha consentito di raccogliere e archiviare sempre maggiori quantità di dati; tuttavia, con l'aumento del loro volume, è aumentata anche la complessità della loro gestione. Per comprendere il concetto di "qualità dei dati", può essere appropriato fornire la distinzione tra "dati", "informazione" e "conoscenza". Definizioni celebri di tali termini sono state fornite da Newell, Robertson, Scarbrough e Swan nell'articolo "Managing Knowledge Work" [8], che definiscono i dati come "catalogazione di segni e osservazioni raccolti da varie fonti", le informazioni come "dati che vengono mostrati in modo caratteristico in relazione a un particolare contesto di azione" e conoscenza percepita come di natura impersonale e statica o come personale e correlata all'azione. La qualità dei dati è spesso definita anche come "idoneità all'uso", ovvero una valutazione dell'utilità dei dati, e quindi anche della qualità, in base agli scopi dell'utente.

Dati di scarsa qualità possono implicare una moltitudine di conseguenze negative, come la difficoltà a creare fiducia nei dati aziendali, e per questo eventuali iniziative basate su di essi potrebbero essere respinte da parte dell'utente finale. Come riporta l'articolo "The costs of poor data quality" scritto da

Anders Haug, Frederik Zachariassen e Dennis van Liempd [6], molte aziende subiscono costi significativi a causa di dati di scarsa qualità, sebbene l'esatta entità di tali costi sia difficile da stimare.

I vantaggi di una buona data preparation sono molteplici:

- Aiuta a correggere gli errori rapidamente, in quanto ne consente la rilevazione prima dell'elaborazione. Gli errori diventano più difficili da comprendere e da correggere dopo che i dati vengono estrapolati dalla fonte principale.
- Consente di produrre dati di alta qualità grazie alla pulizia e alla riformattazione dei set di dati.
- Fornisce supporto per decisioni aziendali migliori e più informate in quanto si ha accesso a dati di maggiore affidabilità.

La data preparation non è un processo monolitico ma comprende diversi passaggi distinti. Possono esistere delle lievi variazioni nella distinzione dei passaggi elencati in seguito a seconda dei fornitori di servizi per la data preparation, ma in generale il processo prevede le seguenti attività:

- **Data discovery** (Raccolta ed esplorazione dei dati): La preparazione inizia con il processo di analisi preliminari e di raccolta dei dati provenienti da diverse fonti. I dati rilevanti vengono raccolti e viene fatta una esplorazione di questi per comprendere meglio cosa contengono.
- **Data validation** (Validazione dei dati): Dopo la raccolta è importante comprendere le regole e i vincoli per analizzare i dati, ossia se ne controlla la validità rispetto all'utilizzo che ne verrà fatto.
- **Data structuring** (Strutturazione dei dati): Comprende operazioni per la creazione, rappresentazione e strutturazione delle informazioni. In questa fase i dati vengono trasformati e modellati per renderli coerenti tra loro e utilizzabili da particolari algoritmi di analisi.
- **Data enrichment** (Arricchimento dei dati): Integra i dati esistenti con informazioni supplementari acquisite da fonti esterne. L'arricchimento e l'ottimizzazione servono per fornire degli approfondimenti ai dataset così che si possano produrre, attraverso la loro elaborazione, le informazioni aziendali utili alle decisioni di business o ad altri processi.
- **Data filtering** (Filtraggio dei dati): Genera dei sottoinsiemi dei dati così da facilitare l'ispezione manuale, riducendo la mole dei dati di input, ma senza compromettere la validità delle analisi.

- **Data cleaning** (Pulizia dei dati): È la parte che richiede più tempo ma è fondamentale per la rimozione, aggiunta o sostituzione di valori poco precisi con valori più accurati. Un buon processo di pulizia prevede, ad esempio, unione dei dati duplicati, inserimento dei valori mancanti, rimozione dei valori non validi e correzione delle inconsistenze.

In realtà il data cleaning riguarda le successive trasformazioni e correzioni dei dati a livello semantico e non viene eseguito insieme alla data preparation, ma la maggior parte degli strumenti di preparazione di dati si occupa anche di quest'area.

Trifacta, società di software per la gestione dei dati, nel report annuale *End User Data Preparation* del 28 febbraio 2018 [3], mostra che il 72% degli intervistati afferma che la preparazione dei dati è fondamentale, mentre l'88% ha evidenziato almeno la sua importanza, e solo il 4% ha indicato che non è importante. Inoltre, il 72% degli intervistati afferma di fare un uso costante della data preparation e solo il 7% la utilizza raramente/mai.
Si deve aggiungere che la data preparation è un'attività molto dispendiosa in termini di tempo e questo rischia di allungare di molto il lavoro degli esperti di analisi e gestione dei dati, soprattutto perché il volume di dati utilizzati nelle applicazioni di analisi continua ad aumentare. Tuttavia, il tempo dedicato a questo processo può essere ridotto drasticamente utilizzando strumenti che automatizzano i metodi di data preparation, consentendo di accedere, pulire e trasformare i dati in modo semplice ed interattivo. Una volta raccolti i dati, il tool/software esegue la data preparation attraverso un flusso di lavoro (*workflow*), durante il quale vengono applicate delle operazioni specifiche. Infine, i dati vengono trasferiti in un data warehouse o un altro archivio per essere analizzati.

Nell'ambito del presente elaborato si fa riferimento al paper "Data Preparation: A Survey of Commercial Tools" di Hameed e Naumann [2] in cui vengono analizzati alcuni dei tool più frequentemente utilizzati per la data preparation. Tra i software descritti in questo paper ritroviamo ad esempio:

- Paxata Self-Service Data Preparation, che offre molte funzionalità per organizzare e preparare i dati strutturati e gestisce anche dati semi-strutturati.
- SAS Data Preparation, che oltre alle funzionalità comuni offre trasformazioni basate sul codice e permette agli utenti di scrivere e condividere codice personalizzato.

- Talend Data Preparation, che offre molte funzionalità specifiche per la preparazione dei dati su misura per il lavoro da svolgere.
- Trifacta Wrangler, che prepara i dati utilizzando più funzioni e prevede in modo intelligente gli schemi per fornire dei suggerimenti che aiutano gli utenti a trasformare i dati.

Molte operazioni per la data preparation sono implementate anche in KNIME, il software analizzato nel presente elaborato, in quanto offre dei connettori nativi per accedere alla maggior parte delle fonti ed implementa qualsiasi trasformazione debba essere eseguita sui dati.

Soprattutto con il rilascio della versione 4.3, KNIME ha fatto un grande passo avanti per potenziare l'accesso ai dati e la loro preparazione e renderli ancora più potenti e facili da utilizzare grazie a una serie di estensioni per la gestione di file.

2. INTRODUZIONE A KNIME

KNIME Analytics Platform è una piattaforma open source nata nel luglio del 2006 all'Università di Costanza per analisi dei dati e reportistica. Il suo nome è una sorta di acronimo che ricorda con KN la sua città natale (Konstanz), con IM Information Mining, mentre la E finale si deve a puri motivi eufonici.

La realizzazione di tale piattaforma è iniziata all'inizio del 2004 ad opera di un team di sviluppatori, inizialmente guidato da Michael Berthold, di una società di software della Silicon Valley specializzata in applicazioni farmaceutiche.

È uno strumento utilizzato a livello aziendale in quanto offre un servizio che aiuta a prendere decisioni basate sui dati, garantendo un'interfaccia user friendly, utilizzo gratuito, espandibilità e modularità. Il software fornito da KNIME include la piattaforma KNIME Analytics, il server KNIME, le estensioni KNIME e le integrazioni KNIME. Nel 2019 KNIME è stato inserito da Gartner tra i software leader in questo ambito.

2.1 CARATTERISTICHE GENERALI

KNIME è scritto in Java e basato su Eclipse. L'ambiente di sviluppo software multilingua comprende un ambiente di sviluppo integrato (IDE) e un sistema di plug-in estensibile; è rilasciato con una licenza Open Source GPLv3.

È possibile scaricare l'ultima versione di KNIME (attualmente la 4.4.1) sul sito https://www.knime.com/downloads/download-knime selezionando la versione del software in base al sistema operativo: Windows, Linux e Mac.

KNIME Analytics Platform è una piattaforma gratuita ma KNIME Server, che offre delle funzionalità aggiuntive, come la possibilità di condividere competenze tra team, rispettare meglio le politiche di protezione dei dati, pianificare automaticamente flussi di lavoro, ridimensionare l'esecuzione del flusso di lavoro e molto altro, è a pagamento con tre diversi abbonamenti annuali.

Una volta installato il software e avviata l'applicazione, viene richiesto di definire la workspace directory, ossia la cartella sul computer locale per memorizzare i flussi di lavoro; quella di default è C:\User\CartellaUtente\knime-workspace. Dopo aver selezionato l'area di lavoro locale si apre

l'interfaccia utente della piattaforma (*workbench*) che di solito è organizzata come mostrato nella Figura 2.1.

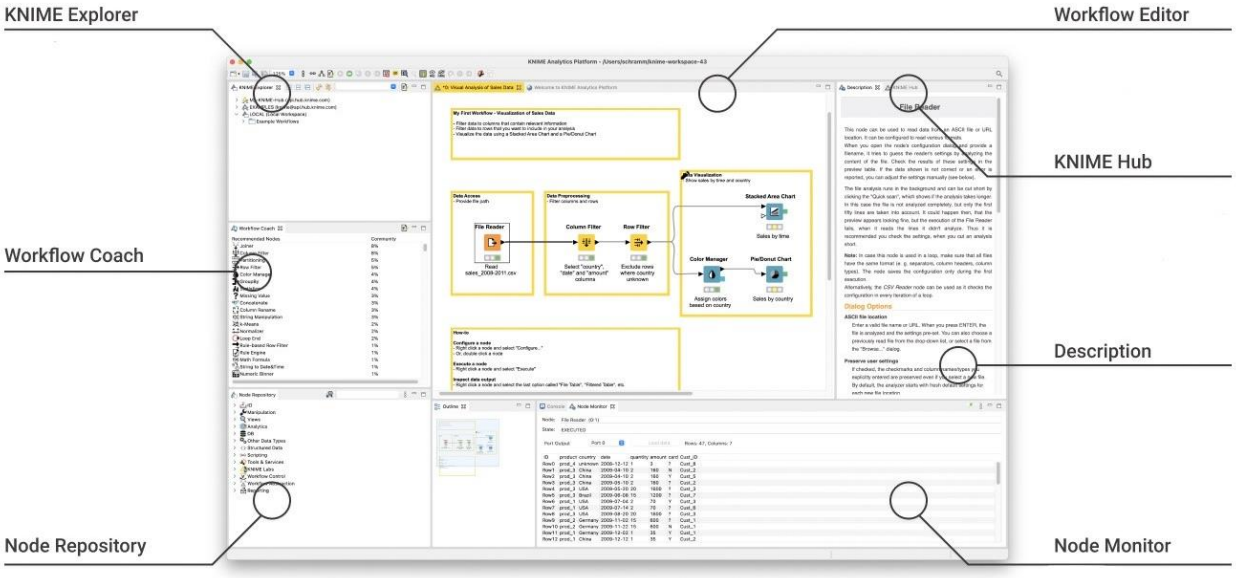


Figura 2.1 KNIME Workbench.

- **KNIME Explorer**: indica lo spazio di lavoro locale e viene fornita una panoramica dei flussi di lavoro disponibili.
- **Workflow Coach**: è attivo solo se viene permesso a KNIME di raccogliere le statistiche di utilizzo e consiglia dei nodi in base ai flussi di lavoro creati dalla community di KNIME.
- **Node repository**: vengono elencati tutti i nodi disponibili in KNIME e nelle estensioni installate. I nodi vengono organizzati per categoria ed è presente anche una casella di ricerca nella parte superiore del repository. La ricerca può avvenire con due modalità: standard, che ricerca una corrispondenza esatta del nome del nodo e fuzzy, che trova il nome del nodo più simile alla ricerca.
- **Workflow editor**: è lo spazio per la creazione e modifica del flusso di lavoro attivo.
- **Description**: il pannello di descrizione a destra del KNIME Workbench fornisce una descrizione del flusso di lavoro attualmente attivo o di un nodo selezionato nel repository o nell'editor del flusso di lavoro.
- **Outline**: nella parte inferiore del Workbench si trova una panoramica del flusso di lavoro attualmente attivo.
- **Console**: la scheda della console, sempre nella parte inferiore del Workbench, mostra i messaggi di avviso e di errore relativi all'esecuzione del flusso di lavoro.

- **Node Monitor**: si trova nello stesso pannello della console ed è particolarmente utile per esaminare le tabelle di output intermedie nel flusso di lavoro. Viene mostrato come impostazione predefinita a partire dalla versione 4.2.

2.2 NODI

In KNIME Analytics Platform le singole attività sono rappresentate dai nodi, che sono le unità base di computazione. Ad oggi esistono più di 4000 nodi che eseguono ogni tipo di attività, inclusa lettura e scrittura di file, trasformazione, creazione e visualizzazione dei dati. Ogni nodo viene rappresentato da un riquadro colorato con delle porte di input, output e uno stato. Nella Figura 2.2 è riportato il nodo che consente di filtrare le colonne da una tabella di input con evidenziate le varie caratteristiche di un nodo.

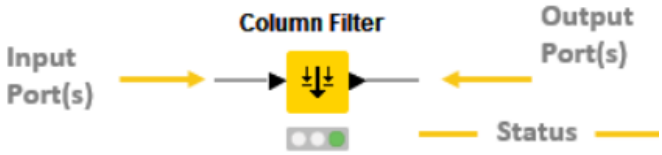


Figura 2.2 Nodo in KNIME.

Per inserire un nodo nel flusso di lavoro è possibile trascinarlo dal Node Repository oppure fare doppio clic sul nodo e questo verrà automaticamente visualizzato nell'editor del flusso di lavoro.

Ogni nodo ha delle impostazioni specifiche che si possono regolare dal pannello di configurazione.

Lo stato di un nodo può essere modificato configurandolo, eseguendolo o ripristinandolo.
Un nodo può trovarsi in quattro stati diversi e questo viene mostrato da un semaforo sotto ogni nodo.

Figura 1.3 Stato dei nodi

Not configured indica che il nodo è in attesa di essere configurato o in attesa di dati.
Configured indica che il nodo è stato configurato correttamente e può essere eseguito.
Executed indica che il nodo è stato eseguito e il risultato può essere mostrato e/o può essere usato come input per un altro nodo.
Error indica che il nodo ha riscontrato un errore durante l'esecuzione.

Quando un nuovo nodo viene aggiunto per la prima volta al flusso di lavoro, il suo stato è “non configurato” e può essere configurato nelle impostazioni del context menu facendo clic con il pulsante destro del mouse.

2.3 PORTE DI INPUT E OUTPUT

Gli input sono i dati da elaborare che vengono passati al nodo, mentre gli output sono i dati risultanti uscenti dal nodo. Un nodo può avere più porte di input e più porte di output come il nodo *Partitioning* che divide la tabella di input in due partizioni disponibili sulle due porte di output. Alcune porte di input possono essere vuote, ciò significa che il nodo può essere eseguito senza l’input, come la porta di input del nodo *DELETE Request,* che viene utilizzato per inviare richieste HTTP DELETE. Le richieste DELETE vengono utilizzate per eliminare le risorse su un server, di solito non si invia nessun dato con la richiesta e non si riceve nulla in cambio ma in modo facoltativo può essere passata all’input una tabella di dati contenente i parametri della richiesta.

Esistono porte di diverso tipo rappresentate da diversi colori; solo quelle dello stesso colore possono essere collegate tra loro (da dati a dati, da modello a modello). Nella figura 2.4 si riportano alcuni esempi di porte divise per tipologia:

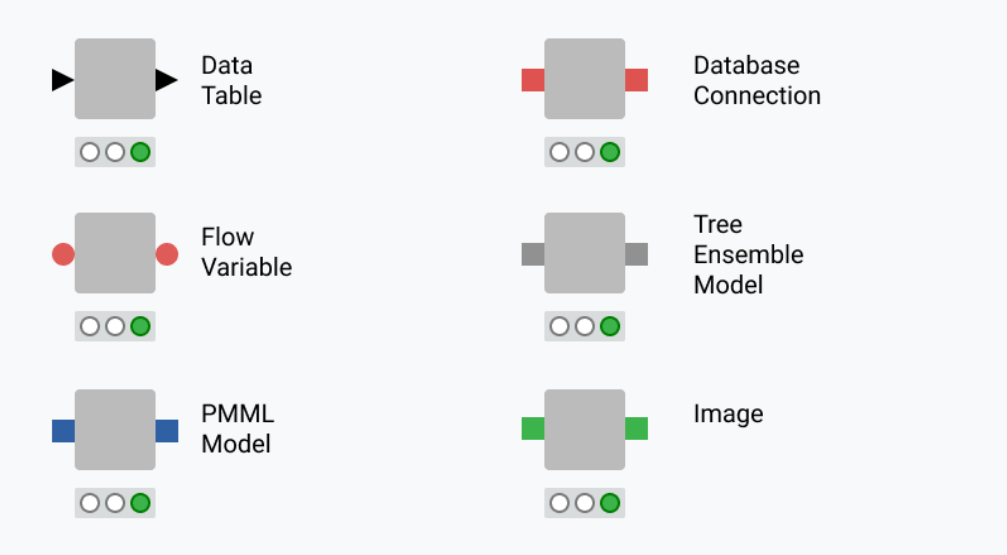


Figura 2.4 Porte di input e output.

2.4 COMPONENTI E METANODI

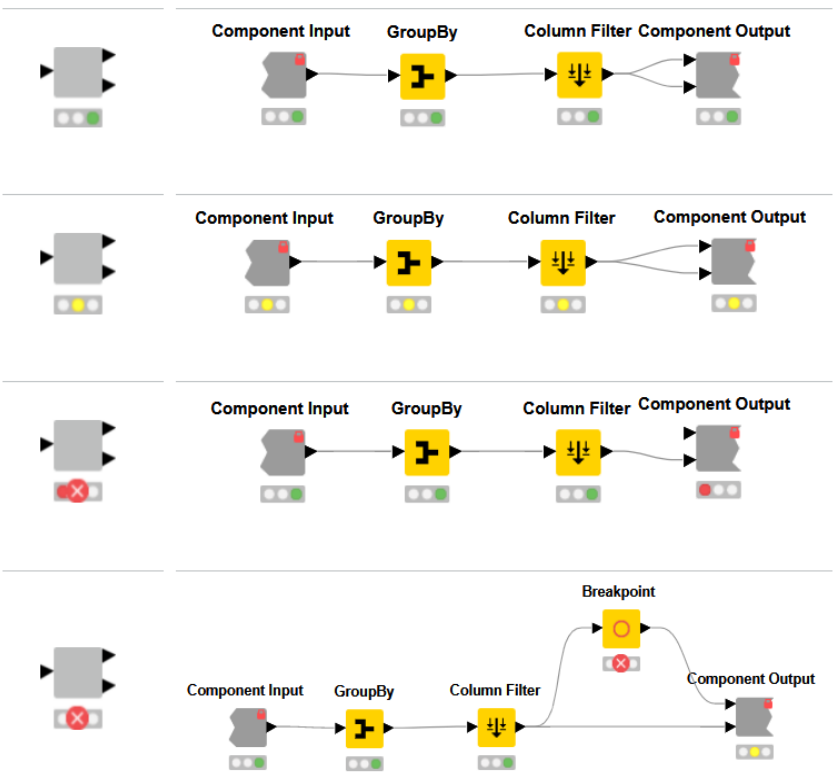
In KNIME è possibile diminuire il flusso di lavoro primario unendo tra loro dei nodi attraverso i componenti e i metanodi.

I componenti sono dei nodi che contengono del flusso di lavoro secondario e consentono di raggruppare delle funzionalità per la condivisione e il riutilizzo. I componenti hanno una propria finestra di dialogo e una vista interattiva personalizzata. Si possono usare per nascondere delle parti del flusso di lavoro e possono essere riutilizzati in altri workflow o in parti diverse dello stesso, oppure possono essere condivisi con altri utenti tramite *KNIME Server* o sul *KNIME Hub.* Inoltre, i componenti vengono utilizzati per definire le pagine di un'applicazione Web, che una volta distribuiti tramite *KNIME Server* sono accessibili tramite *KNIME WebPortal*.

I metanodi invece vengono utilizzati semplicemente per organizzare al meglio il lavoro, in quanto consentono di prendere una parte del flusso di lavoro e comprimerlo in un'unica casella grigia che nasconde quella parte della funzionalità del workflow. Questo rende molto più facile per gli altri utenti capire cosa fa il flusso di lavoro.

Per creare un metanodo o un componente si scelgono i nodi che si vogliono comprimere e si seleziona *Create Component* o *Create Metanode.* Una volta creati, di entrambi gli elementi, verranno visualizzate le porte di input e output in base alle connessioni di entrata e uscita. È possibile modificare diverse impostazioni dei metanodi e dei componenti come il nome, il numero di porte di input e output e i loro tipi.

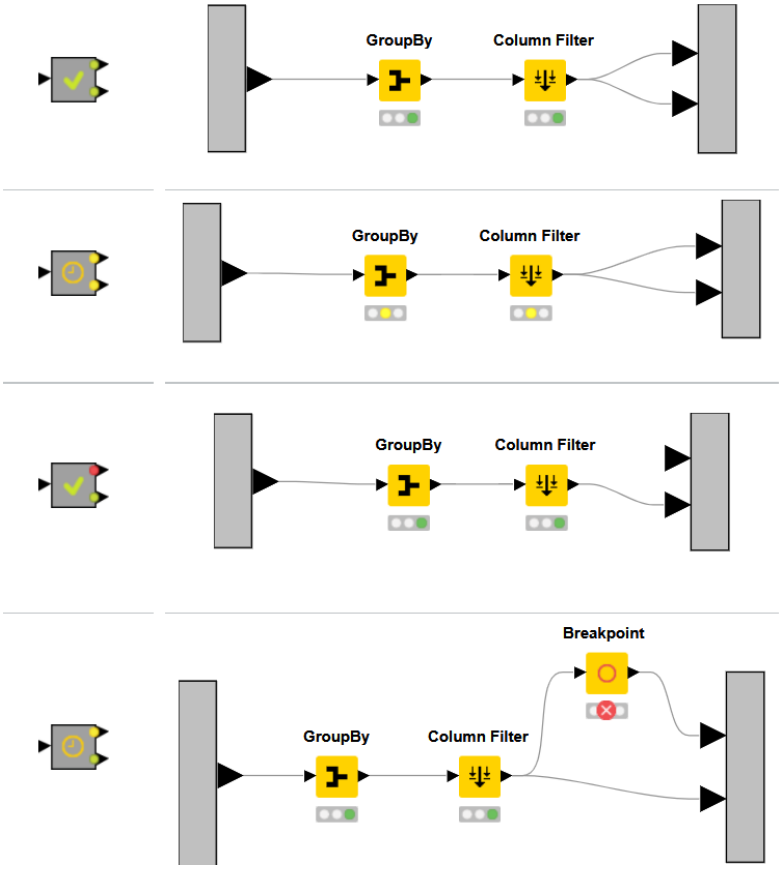
In modo molto simile ai nodi, i componenti possono essere configurati ed eseguiti e utilizzano un semaforo per indicare il loro stato di esecuzione In Figura 2.5 vengono riportati i vari stati:



Indica che il componente è stato eseguito con successo.
Indica che il componente è stato configurato e tutte le uscite sono collegate.
Indica che l'esecuzione è fallita in quanto contiene un output non connesso.
Indica che l'esecuzione è fallita in quanto un nodo interno non ha completato l'esecuzione.

Figura 2.5 Stato dei componenti.

È possibile eseguire anche i metanodi, il che significa che verranno eseguiti i nodi in esso contenuti. Dal momento che essi sono solo dei contenitori di nodi, non possono essere configurati. Lo stato dei metanodi è individuato da diverse icone. In Figura 2.6 si riporta la tabella:



Indica che il metanodo è stato eseguito con successo.
Indica che tutti i nodi all'interno del metanodo sono stati configurati e tutte le uscite sono collegate.
Vengono eseguiti i nodi del metanodo, ma esso contiene un output non connesso.
Vengono eseguiti tutti i nodi. Un nodo all'interno del flusso secondario non è stato eseguito correttamente.

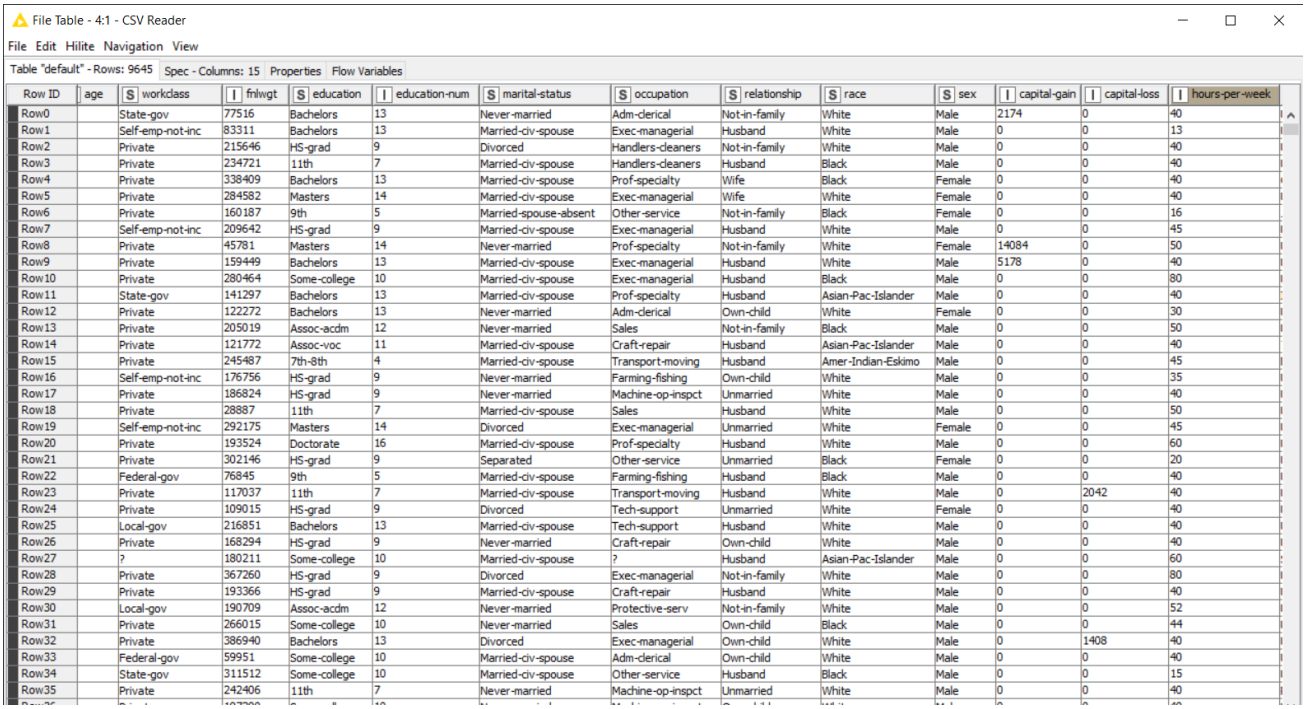
Figura 2.6 Stato dei metanodi.

2.5 TABELLE IN KNIME

Molto spesso le porte di input e output sono porte che accettano dei dati e corrispondono ai triangoli neri entranti e uscenti dal nodo. Una tabella di dati in KNIME è organizzata per righe e colonne dove gli elementi di ogni colonna devono essere tutti dello stesso tipo di dato. I tipi di dato supportati sono: *Integer, Double, String, Long, Boolean, URI, Document, Date&Time, Bit, Vector, Image e Blob*. KNIME supporta anche tipi di dato personalizzati.

Ogni riga è identificata da un ID univoco creato automaticamente dal nodo e ogni colonna è definita da un header. Un valore mancante è identificato con un punto interrogativo *?*.

In figura 2.7 si riporta una tabella creata da un nodo *CSV Reader.*



Row ID	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week
Row0		State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40
Row1		Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13
Row2		Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40
Row3		Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0	40
Row4		Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40
Row5		Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Female	0	0	40
Row6		Private	160187	9th	5	Married-spouse-absent	Other-service	Not-in-family	Black	Female	0	0	16
Row7		Self-emp-not-inc	209642	HS-grad	9	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	45
Row8		Private	45781	Masters	14	Never-married	Prof-specialty	Not-in-family	White	Female	14084	0	50
Row9		Private	159449	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	5178	0	40
Row10		Private	280464	Some-college	10	Married-civ-spouse	Exec-managerial	Husband	Black	Male	0	0	80
Row11		State-gov	141297	Bachelors	13	Married-civ-spouse	Prof-specialty	Husband	Asian-Pac-Islander	Male	0	0	40
Row12		Private	122272	Bachelors	13	Never-married	Adm-clerical	Own-child	White	Female	0	0	30
Row13		Private	205019	Assoc-acdm	12	Never-married	Sales	Not-in-family	Black	Male	0	0	50
Row14		Private	121772	Assoc-voc	11	Married-civ-spouse	Craft-repair	Husband	Asian-Pac-Islander	Male	0	0	40
Row15		Private	245487	7th-8th	4	Married-civ-spouse	Transport-moving	Husband	Amer-Indian-Eskimo	Male	0	0	45
Row16		Self-emp-not-inc	176756	HS-grad	9	Never-married	Farming-fishing	Own-child	White	Male	0	0	35
Row17		Private	186824	HS-grad	9	Never-married	Machine-op-inspct	Unmarried	White	Male	0	0	40
Row18		Private	28887	11th	7	Married-civ-spouse	Sales	Husband	White	Male	0	0	50
Row19		Self-emp-not-inc	292175	Masters	14	Divorced	Exec-managerial	Unmarried	White	Female	0	0	45
Row20		Private	193524	Doctorate	16	Married-civ-spouse	Prof-specialty	Husband	White	Male	0	0	60
Row21		Private	302146	HS-grad	9	Separated	Other-service	Unmarried	Black	Female	0	0	20
Row22		Federal-gov	76845	9th	5	Married-civ-spouse	Farming-fishing	Husband	Black	Male	0	0	40
Row23		Private	117037	11th	7	Married-civ-spouse	Transport-moving	Husband	White	Male	0	2042	40
Row24		Private	109015	HS-grad	9	Divorced	Tech-support	Unmarried	White	Female	0	0	40
Row25		Local-gov	216851	Bachelors	13	Married-civ-spouse	Tech-support	Husband	White	Male	0	0	40
Row26		Private	168294	HS-grad	9	Never-married	Craft-repair	Own-child	White	Male	0	0	40
Row27		?	180211	Some-college	10	Married-civ-spouse	?	Husband	Asian-Pac-Islander	Male	0	0	60
Row28		Private	367260	HS-grad	9	Divorced	Exec-managerial	Not-in-family	White	Male	0	0	80
Row29		Private	193366	HS-grad	9	Married-civ-spouse	Craft-repair	Husband	White	Male	0	0	40
Row30		Local-gov	190709	Assoc-acdm	12	Never-married	Protective-serv	Not-in-family	White	Male	0	0	52
Row31		Private	266015	Some-college	10	Never-married	Sales	Own-child	Black	Male	0	0	44
Row32		Private	386940	Bachelors	13	Divorced	Exec-managerial	Own-child	White	Male	0	1408	40
Row33		Federal-gov	59951	Some-college	10	Married-civ-spouse	Adm-clerical	Own-child	White	Male	0	0	40
Row34		State-gov	311512	Some-college	10	Married-civ-spouse	Other-service	Husband	Black	Male	0	0	15
Row35		Private	242406	11th	7	Never-married	Machine-op-inspct	Unmarried	White	Male	0	0	40

Figura 2.7 Esempio di tabella in KNIME.

Per ogni colonna può essere selezionato l'ordinamento crescente o decrescente che ha però solo effetto sulla visualizzazione della tabella e non ha alcun effetto sull'output del nodo.

È inoltre possibile modificare il modo in cui i valori numerici vengono visualizzati all'interno di una tabella: possono essere rappresentati come percentuale, come scala di grigi o come numeri complessi.

Quando il numero di nodi che generano delle tabelle diventa alto, KNIME deve decidere come gestire la memoria ed è per questo che esistono diverse possibilità per stabilire quali tabelle tenere in memoria centrale e quali spostare sul disco rigido. KNIME distingue tra tabelle piccole e grandi, di default meno o più di 5000 celle (ma questo parametro può essere modificato nel file *knime.ini*), cercando di mantenere nella memoria centrale le tabelle piccole, scaricandole su disco solo quando essa diventa scarsa. Inoltre, tenta di mantenere nella memoria centrale le tabelle di grandi dimensioni utilizzate nel breve periodo, per il principio della locazione temporale, fino a quando essa è disponibile; tuttavia, salva tali tabelle anche sul disco così che possano essere eliminate in qualsiasi momento. Le tabelle che vengono scritte su disco vengono invece compresse per ridurre la quantità di spazio occupato, utilizzando l'algoritmo Snappy; tale configurazione può essere modificata sempre nel file *knime.ini.*

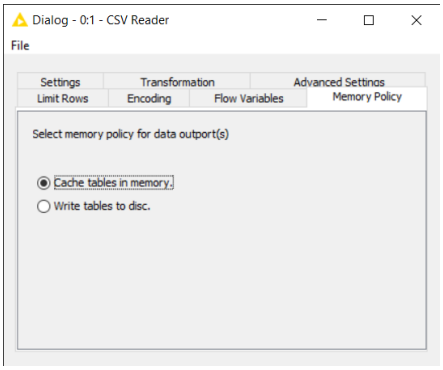


Figura 2.8 Occupazione di memoria delle tabelle.

Inoltre, è possibile decidere in modo arbitrario se salvare il nodo nella memoria centrale o direttamente nel disco nella configurazione del nodo stesso.

2.6 FLUSSO DI LAVORO

Per creare un flusso di lavoro vuoto, fare clic con il tasto destro del mouse nel *KNIME Explorer* in qualsiasi parte del lavoro locale e selezionare *New KNIME Workflow.*

Si definisce un flusso di lavoro (*workflow*) una serie di nodi interconnessi tra loro. I nodi vengono collegati tramite le porte di input e output e una volta eseguito il workflow i dati al suo interno scorrono da sinistra a destra attraverso le varie connessioni.

In figura 2.9 si riporta un semplice esempio di un flusso di lavoro in cui viene letto un file di input I dati vengono filtrati con i nodi *Row Filter* e *Column Filter,* poi vengono visualizzati i risultati con tre tipologie di visualizzazioni differenti.

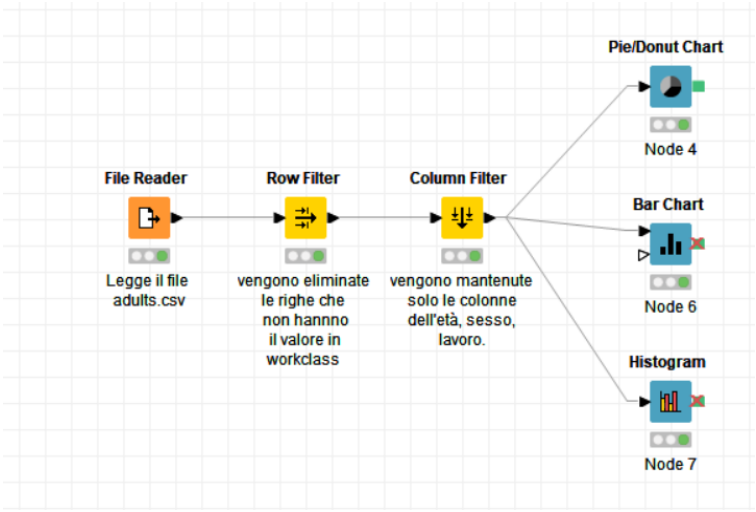


Figura 2.9 Breve flusso di lavoro.

Non tutti i flussi di lavoro hanno però un input statico e un solo ramo, in quanto spesso i dati vengono aggiornati regolarmente e alcune impostazioni possono variare. Esistono infatti dei nodi nella

categoria *Workflow Control* del repository che possono essere utilizzati per evitare di modificare manualmente il flusso di lavoro quando è richiesta un'esecuzione con impostazioni diverse.

2.7 INPUT SUPPORTATI

Gli input supportati sono divisi in tre categorie:

- Database: MySQL, PostgreSQL, JDBS
- Files: CSV, TXT, Exel, Word, XML, PMML, Images
- Web, cloud: REST, Webservices, Twitter, Google

2.8 COMMENTI E ANNOTAZIONI

Sono disponibili nell'editor di flusso due opzioni per documentare e commentare il lavoro svolto: aggiungendo un commento a un singolo nodo facendo doppio clic sul campo di testo sotto il nodo e modificando il contenuto oppure aggiungendo un commento generale al flusso di lavoro, facendo doppio clic con il pulsante destro del mouse e selezionando *New Workflow Annotation,* come mostrato in Figura 2.10.

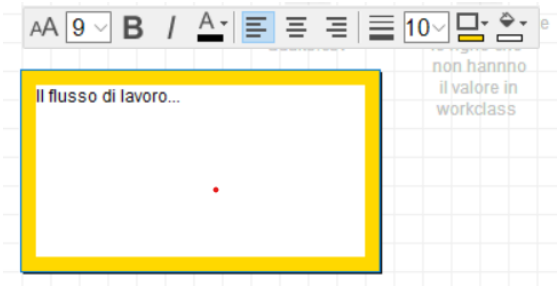


Figura 2.10 Commenti e annotazioni.

2.9 IMPORTARE ED ESPORTARE UN FLUSSO DI LAVORO

Sono disponibili tre possibilità per esportare un flusso di lavoro:

- Esportarlo come file.
- Salvarlo nello spazio personale KNIME Hub (è necessario essere loggati su My-KNIME-Hub).
- Distribuirlo su un server (necessita di una licenza).

Allo stesso modo, si può importare un flusso di lavoro nei seguenti modi:

- Importare un file.

- Salvare un flusso di lavoro che si trova su un server.

2.10 ESTENSIONI E INTEGRAZIONI

Le estensioni forniscono funzionalità aggiuntive, come l'elaborazione di dati complessi e il supporto di algoritmi avanzati.

Le integrazioni forniscono l'accesso ad alcuni programmi open source come Keras per il deep learning, H2O, Apache Spark per l'elaborazione di big data, Python e R.

Le estensioni disponibili possono essere fornite da:

- KNIME Community Extensions (trusted): fornisce estensioni dalla community di KNIME, testate per la compatibilità con le versioni precedenti e per la conformità con gli standard di qualità di KNIME.
- KNIME Partner Extensions: fornisce estensioni create dai partner di KNIME.
- Community Extensions (experimental): fornisce estensioni aggiuntive create dalla comunità di KNIME che però sono in fase sperimentale; quindi, non testate per la compatibilità e gli standard di qualità di KNIME.

3. DATA PREPARATION CON KNIME

3.1 INTRODUZIONE

Come detto in precedenza, per quanto riguarda la sezione dedicata alla data preparation, il presente elaborato si basa sul paper “Data Preparation: A Survey of Commercial Tools” [2], scritto da Mazhar Hameed e Felix Naumann. Gli autori, dopo una breve introduzione sull’importanza e la necessità della data preparation, sempre più rilevante nel contesto attuale, prendono in considerazione 40 funzionalità, indicate come “preparators”, considerate necessarie per un buon processo di preparazione dei dati. Partendo da un’indagine svolta su 42 software, gli autori hanno individuato 7 tools, che hanno analizzato e valutato verificando se implementassero o meno le 40 funzionalità prese in considerazione.

Come si può vedere, in Figura 3.1, i “preparators” sono divisi per macrocategorie e con questa divisione sono andata a sviluppare la tesi, prendendo in considerazione alcuni dei “preparators”, quelli più frequenti e rilevanti per ogni categoria, evidenziati in giallo.

Categories	Available features
Data discovery	Locate missing values (nulls)
	Locate outliers
	Search by pattern
	Sort data
Data validation	Compare values (selection and join)
	Check data range
	Check permitted characters
	Check column uniqueness
	Find type-mismatched data
	Find data-mismatched datatypes
Data structuring	Change column data type
	Delete column
	Detect & change encoding
	Pivot / unpivot
	Rename column
	Split column
	Transform by example [13]
Data enrichment	Assign semantic data type
	Calculate column using expressions
	Discover & merge external data
	Duplicate column
	Generate primary key column
	Join & union
	Merge columns
	Normalize numeric values
Data filtering	Delete/keep filtered rows
	Delete empty and invalid rows
	Extract value parts
	Filter with regular expressions
Data cleaning	Change date & time format
	Change letter case
	Change number format
	Deduplicate data
	Delete by pattern
	Edit & replace cell data
	Fill empty cells
	Remove extra whitespace
	Remove diacritics
	Standardize strings by pattern
	Standardize values in clusters

Figura 3.1 I 40 "preparators".

3.2 DATASET

Nell'articolo di riferimento [2], gli autori prendono in considerazione vari dataset per testare le diverse funzionalità sui software selezionati.

Per testare KNIME ho scelto di considerare un dataset tra quelli proposti dai ricercatori: *120 years of Olimpic history: athletes and result.* È un dataset storico sui Giochi Olimpici moderni, partendo dai Giochi di Atene 1896 fino a Rio 2016. Sono presenti due file: *athletic_events.csv* e *noc_regions.csv.*

Il primo file contiene 271116 righe e 15 colonne. Ogni riga corrisponde alla partecipazione di un atleta ad una certa edizione, per cui si riportano i seguenti attributi (colonne):

- ID: codice identificativo di ogni atleta.
- Name: nome dell'atleta.
- Sex: sesso dell'atleta.
- Age: età dell'atleta.
- Height: altezza dell'atleta.
- Weight: peso dell'atleta.
- Team: nome della squadra.
- NOC: codice di tre lettere del Comitato Olimpico Nazionale (*National Olympic Commitee*).
- Games: riporta anno e stagione (*summer o winter*).
- Year: anno di svolgimento.
- Season: stagione in cui si sono svolti i Giochi. Estate (*summer*) o inverno (*winter*)
- City: città dove si sono svolti i Giochi.
- Sport: sport praticato dall'atleta.
- Event: nome dell'evento.
- Medal: medaglia, che può essere oro (*gold*), argento (*silver*), bronzo (*bronze*) o NA.

In Figura 3.2 si riporta la tabella del file.

Row ID	ID	Name	Sex	Age	Height	Weight	Team	NOC	Games	Year	Season	City	Sport	Event	Medal
Row0	1	A Dijiang	M	24	180	80	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Basketball Men's Basketball	NA
Row1	2	A Lamusi	M	23	170	60	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Men's Extra-Lightw...	NA
Row2	3	Gunnar Niels...	M	24	?	?	Denmark	DEN	1920 Summer	1920	Summer	Antwerpen	Football	Football Men's Football	NA
Row3	4	Edgar Linde...	M	34	?	?	Denmark/Sw...	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's Tug-O...	Gold
Row4	5	Christine Ja...	F	21	185	82	Netherlands	NED	1988 Winter	1988	Winter	Calgary	Speed Skating	Speed Skating Women's ...	NA
Row5	5	Christine Ja...	F	21	185	82	Netherlands	NED	1988 Winter	1988	Winter	Calgary	Speed Skating	Speed Skating Women's ...	NA
Row6	5	Christine Ja...	F	25	185	82	Netherlands	NED	1992 Winter	1992	Winter	Albertville	Speed Skating	Speed Skating Women's ...	NA
Row7	5	Christine Ja...	F	25	185	82	Netherlands	NED	1992 Winter	1992	Winter	Albertville	Speed Skating	Speed Skating Women's ...	NA
Row8	5	Christine Ja...	F	27	185	82	Netherlands	NED	1994 Winter	1994	Winter	Lillehammer	Speed Skating	Speed Skating Women's ...	NA
Row9	5	Christine Ja...	F	27	185	82	Netherlands	NED	1994 Winter	1994	Winter	Lillehammer	Speed Skating	Speed Skating Women's ...	NA
Row10	6	Per Knut Aal...	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA
Row11	6	Per Knut Aal...	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA
Row12	6	Per Knut Aal...	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA
Row13	6	Per Knut Aal...	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA
Row14	6	Per Knut Aal...	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA
Row15	6	Per Knut Aal...	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA
Row16	6	Per Knut Aal...	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA
Row17	6	Per Knut Aal...	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA
Row18	7	John Aalberg	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA
Row19	7	John Aalberg	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA
Row20	7	John Aalberg	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA
Row21	7	John Aalberg	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA
Row22	7	John Aalberg	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA
Row23	7	John Aalberg	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA
Row24	7	John Aalberg	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA
Row25	7	John Aalberg	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA
Row26	8	Cornelia Cor...	F	18	168	?	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 100 m...	NA
Row27	8	Cornelia Cor...	F	18	168	?	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 4 x 10...	NA
Row28	9	Antti Sami A...	M	26	186	96	Finland	FIN	2002 Winter	2002	Winter	Salt Lake City	Ice Hockey	Ice Hockey Men's Ice Ho...	NA
Row29	10	Einar Ferdin...	M	26	?	?	Finland	FIN	1952 Summer	1952	Summer	Helsinki	Swimming	Swimming Men's 400 met...	NA
Row30	11	Jorma Ilmari...	M	22	182	76.5	Finland	FIN	1980 Winter	1980	Winter	Lake Placid	Cross Count...	Cross Country Skiing Me...	NA
Row31	12	Jyri Tapani ...	M	31	172	70	Finland	FIN	2000 Summer	2000	Summer	Sydney	Badminton	Badminton Men's Singles	NA
Row32	13	Minna Maari...	F	30	159	55.5	Finland	FIN	1996 Summer	1996	Summer	Atlanta	Sailing	Sailing Women's Windsurfer	NA
Row33	13	Minna Maari...	F	34	159	55.5	Finland	FIN	2000 Summer	2000	Summer	Sydney	Sailing	Sailing Women's Windsurfer	NA
Row34	14	Pirjo Hannel...	F	32	171	65	Finland	FIN	1994 Winter	1994	Winter	Lillehammer	Biathlon	Biathlon Women's 7.5 kilo...	NA
Row35	15	Arvo Ossian...	M	22	?	?	Finland	FIN	1912 Summer	1912	Summer	Stockholm	Swimming	Swimming Men's 200 met...	NA
Row36	15	Arvo Ossian...	M	22	?	?	Finland	FIN	1912 Summer	1912	Summer	Stockholm	Swimming	Swimming Men's 400 met...	NA
Row37	15	Arvo Ossian...	M	30	?	?	Finland	FIN	1920 Summer	1920	Summer	Antwerpen	Swimming	Swimming Men's 200 met...	Bronze
Row38	15	Arvo Ossian...	M	30	?	?	Finland	FIN	1920 Summer	1920	Summer	Antwerpen	Swimming	Swimming Men's 400 met...	Bronze
Row39	15	Arvo Ossian...	M	34	?	?	Finland	FIN	1924 Summer	1924	Summer	Paris	Swimming	Swimming Men's 200 met...	NA
Row40	16	Juhamatti T...	M	28	184	85	Finland	FIN	2014 Winter	2014	Winter	Sochi	Ice Hockey	Ice Hockey Men's Ice Ho...	Bronze
Row41	17	Paavo Joha...	M	28	175	64	Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Individ...	Bronze
Row42	17	Paavo Joha...	M	28	175	64	Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Team ...	Gold
Row43	17	Paavo Joha...	M	28	175	64	Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Floor E...	NA

Figura 3.2 Tabella di athletic_events.csv.

Il secondo file contiene 230 righe e 3 colonne in cui si associano i vari NOC con i rispettivi paesi.

In Figura 3.3 si riporta la tabella del secondo file.

Row ID	Col0	Col1	Col2
Row0	NOC	region	notes
Row1	AFG	Afghanistan	?
Row2	AHO	Curacao	Netherlands...
Row3	ALB	Albania	?
Row4	ALG	Algeria	?
Row5	AND	Andorra	?
Row6	ANG	Angola	?
Row7	ANT	Antigua	Antigua and...
Row8	ANZ	Australia	Australasia
Row9	ARG	Argentina	?
Row10	ARM	Armenia	?
Row11	ARU	Aruba	?
Row12	ASA	American Sa...	?
Row13	AUS	Australia	?
Row14	AUT	Austria	?
Row15	AZE	Azerbaijan	?
Row16	BAH	Bahamas	?
Row17	BAN	Bangladesh	?
Row18	BAR	Barbados	?
Row19	BDI	Burundi	?
Row20	BEL	Belgium	?
Row21	BEN	Benin	?
Row22	BER	Bermuda	?
Row23	BHU	Bhutan	?
Row24	BIH	Bosnia and ...	?
Row25	BIZ	Belize	?
Row26	BLR	Belarus	?
Row27	BOH	Czech Republic	Bohemia
Row28	BOL	Boliva	?
Row29	BOT	Botswana	?
Row30	BRA	Brazil	?
Row31	BRN	Bahrain	?
Row32	BRU	Brunei	?
Row33	BUL	Bulgaria	?
Row34	BUR	Burkina Faso	?
Row35	CAF	Central Afric...	?

Figura 3.3 Tabella di noc_regions.csv.

3.3 DATA DISCOVERY

3.3.1 Locate missing values (nulls) (Riconoscimento valori mancanti)

Grazie a questa funzione il software individua i valori mancanti all'interno di un database. Questi valori possono essere rappresentati in modo diverso come ad esempio "NA", "?" o con una casella vuota.

KNIME, una volta letto un file con un nodo per la lettura come il *File Reader* o un nodo più specifico come *CSV Reader*, *Excel Reader,* ecc., riconosce automaticamente i valori mancanti e assegna un punto interrogativo rosso. Se invece si deve personalizzare il pattern per identificare i valori nulli, esso si inserisce nella configurazione del nodo *File Reader*.

In Figura 3.4 si cambia il pattern dei valori nulli della colonna *Medal* del dataset *athletic_events.csv* da "?" a "NA" (che nel dataset originale non rappresenta un valore nullo ma che un certo atleta in una specifica partecipazione non ha vinto nessuna medaglia). Nel primo caso NA non è riconosciuto come valore nullo mentre nel secondo caso sì.

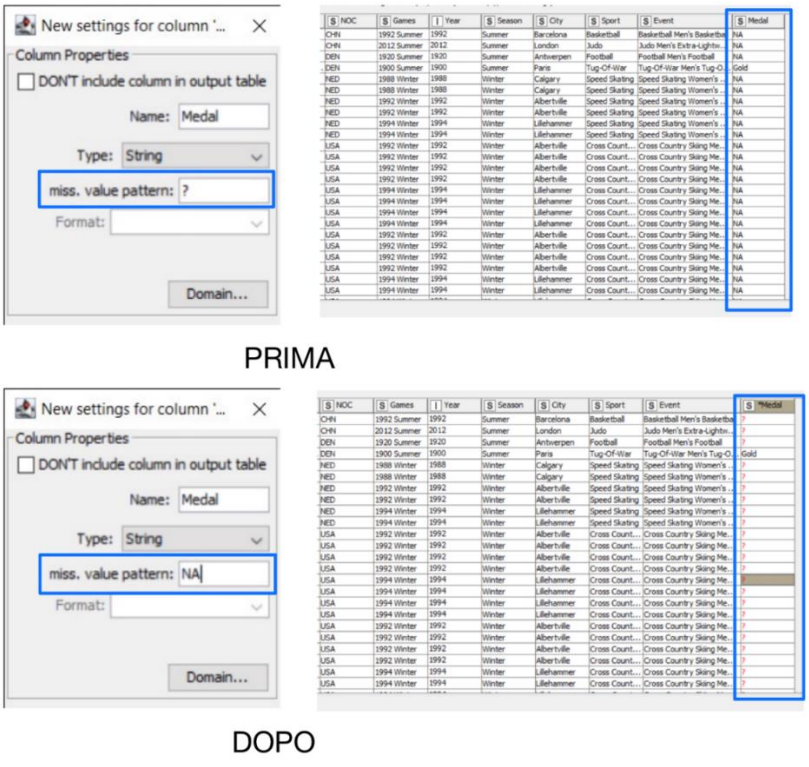


Figura 3.4 Passaggi per modificare il "miss. value pattern".

3.3.2 Locate outliers (Individuazione valori anomali)

Le anomalie all'interno di un dataset possono essere un problema quando si applicano tecniche statistiche. Spesso sono il risultato di errori nelle misurazioni o di condizioni eccezionali del sistema che non descrivono il suo funzionamento standard. Infatti, la migliore pratica consiste nell'implementare una fase di rimozione dei valori anomali prima di procedere con un'ulteriore analisi.

KNIME implementa questa funzionalità con il nodo *Numeric outliers,* che rileva e tratta i valori anomali per ciascuna delle colonne selezionate nella fase di configurazione, mediante l'intervallo interquatile (IQR). Per rilevare gli outliers per una certa colonna vengono calcolati il primo e terzo quartile; un valore viene contrassegnato come outlier se si trova al di fuori dell'intervallo R = [Q_1 - k(IQR), Q_3 + k(IQR)] dove IQR= Q_3 - Q_1 e k deve essere maggiore di 0, deciso in fase di configurazione.

Se un valore è contrassegnato come anomalo, è possibile sostituirlo con un altro valore o rimuovere la riga corrispondente.

In Figura 3.5 è riportato un esempio di utilizzo del nodo sulla colonna *Age* del dataset *athletic_events.csv;* nella prima immagine abbiamo i valori calcolati dal nodo e nelle altre due immagini abbiamo rispettivamente la tabella originaria ordinata in modo discendente per la colonna *Age* e la tabella prodotta dal nodo sempre ordinata in modo discendente per la colonna *Age*. Inoltre, si può notare che tra le due tabelle riduce il numero di righe totali, in quanto in fase di configurazione del nodo si è deciso di eliminare le righe corrispondenti ai valori anomali.

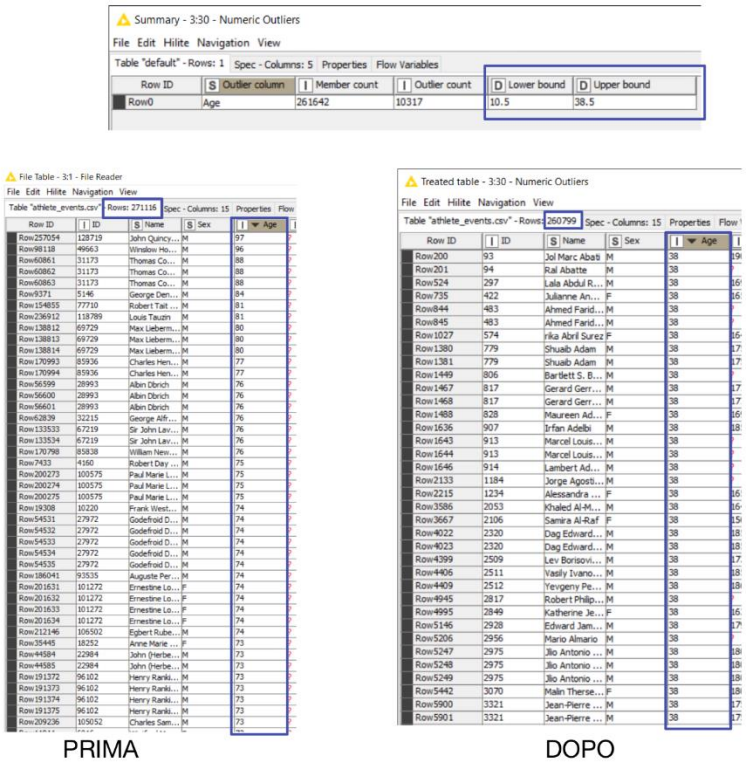


Figura 3.5 Esempio del nodo Numeri outliers sulla colonna Age.

3.3.3 Sort data (Ordinamento dei dati)

L'ordinamento dei dati è un processo molto utile per la loro analisi, in quanto permette di disporli in ordine di grandezza.

In KNIME è possibile implementare questa funzionalità in due modi:

- Direttamente nella tabella di output di un nodo. Il dataset viene ordinato per colonne e possono essere ordinate assieme due colonne alla volta. In Figura 3.6 si può vedere che le colonne possono essere ordinate secondo un criterio crescente o decrescente. Questo ordinamento ha però effetto solo sulla visualizzazione e non sull'output del nodo.

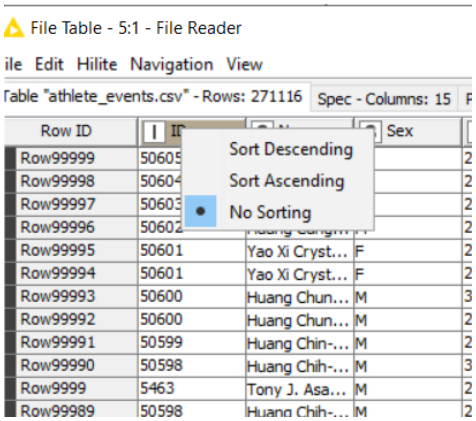


Figura 3.6 Come possono essere organizzate le colonne.

- Oppure attraverso il nodo *Sorter* che ordina le righe in base ai criteri definiti dall'utente. Nella finestra di dialogo si selezionano le colonne in base alle quali i dati devono essere ordinati e se devono essere ordinati in ordine crescente o decrescente.

Si riporta, in Figura 3.7, una tabella ordinata del dataset *athletic_events.csv per Games* in ordine crescente e per *Age* in ordine decrescente.

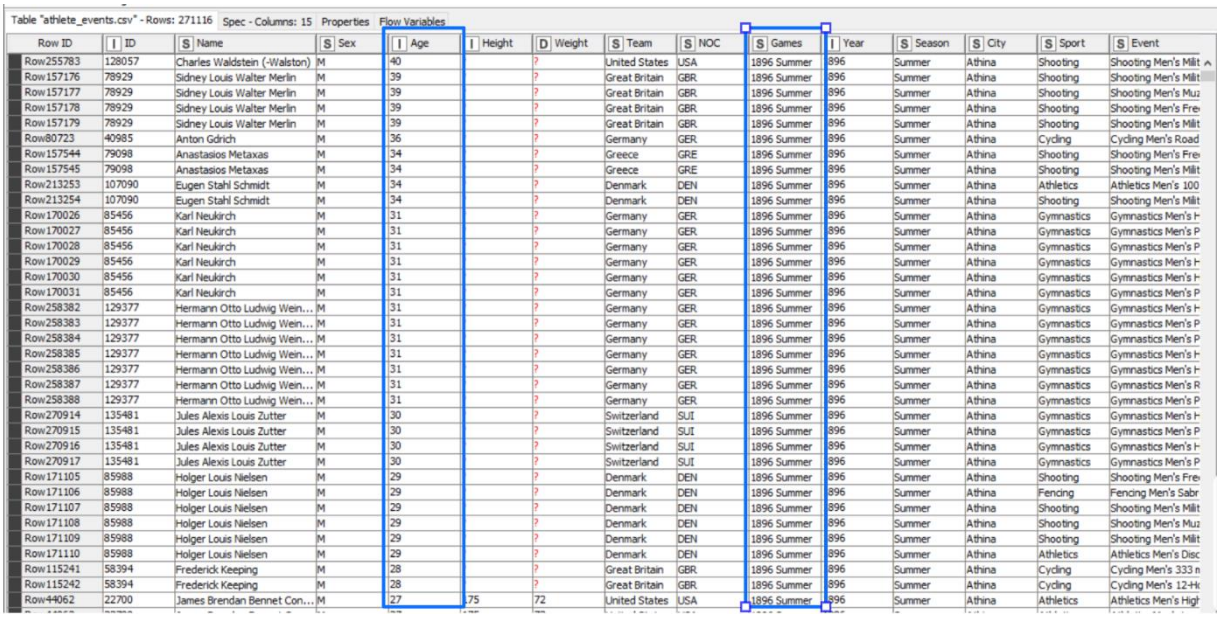


Figura 3.7 Tabella non ordinata e ordinata.

3.4 DATA VALIDATION

3.4.1 Compare values (Confronto di valori)

In KNIME è supportata la possibilità di comparare i dati tra loro attraverso il nodo *Column Comparator,* che confronta i valori delle celle di due colonne selezionate presenti all'interno della stessa riga. Sono disponibili diversi metodi di confronto: == (uguale), != (non_uguale), < (minore), > (maggiore), <= (minore_uguale) e >= (maggiore_uguale). Viene creata una nuova colonna nella tabella di output contenente, a seconda dei casi, il valore di sinistra *(left_value),* il valore di destra (*right_value*), la cella vuota (*missing*) o un valore definito dall'utente (*user_defined*). In Figura 3.8 si riporta un esempio di un dataset che rappresenta i voti degli studenti in due appelli differenti, il nodo crea una nuova colonna *Voto Finale* che tiene il voto più alto tra i due appelli.

Row ID	S Nome	I Matricola	D Voti Appello 1	S Voti Appello 2
Row0	Lucia Rossi	123456	25.7	12
Row1	Franco Lucio	156780	30	28
Row2	Ludovica Bia...	563478	26.9	28.9
Row3	Giovanni Verdi	234561	18	20
Row4	Simone Ferrari	124567	13	18

PRIMA

Row ID	S Nome	I Matricola	D Voti Ap...	S Voti Ap...	? Voto Finale
Row0	Lucia Rossi	123456	25.7	12	25.7
Row1	Franco Lucio	156780	30	28	30.0
Row2	Ludovica Bia...	563478	26.9	28.9	28.9
Row3	Giovanni Verdi	234561	18	20	20
Row4	Simone Ferrari	124567	13	18	18

DOPO

Figura 3.8 Finestra di configurazione del nodo Compare values.

3.4.2 Check data range (Controlla l'intervallo di valori)

Controllare l'intervallo di valori che una variabile assume in un dataset può essere una buona funzionalità per comprendere in modo rapido i valori minimi e massimi che assume una variabile.

In KNIME è possibile visualizzare tali risultati direttamente nella tabella di output di un nodo.

File Table - 3:1 - File Reader

File

Table "athlete_events.csv" - Rows: 271116 | Spec - Columns: 15 | Properties | Flow Variables

Columns: 15	Column Type	Column Index	Lower Bound	Upper Bound	Color Handler	Size Handler
Year	Number (integer)	9	1,896	2,016		
Weight	Number (double)	5	25	214		
Team	String	6	?	?		
Sport	String	12	?	?		
Sex	String	2	?	?		
Season	String	10	?	?		
Name	String	1	?	?		
NOC	String	7	?	?		
Medal	String	14	?	?		
ID	Number (integer)	0	1	135,571		
Height	Number (integer)	4	127	226		
Games	String	8	?	?		
Event	String	13	?	?		
City	String	11	?	?		
Age	Number (integer)	3	10	97		

Figura 3.9 Limiti superiori e inferiori.

3.4.3 Find type-mismatched data (Trovare dati non corrispondenti al tipo assegnato)

Questa operazione si riferisce alla capacità di trovare dei dati sbagliati rispetto al tipo di dato assegnato alla colonna.

KNIME non presenta nessun nodo che implementa questa funzionalità anche perché KNIME non consente di assegnare a una colonna un tipo non compatibile coi dati.

In Figura 3.10 si è tentato di modificare il tipo della colonna *Nome* da String a Integer, ma questo non è consentito e la colonna è stata evidenziata di rosso.

Table Creator Settings | Flow Variables | Job Manager Selecti

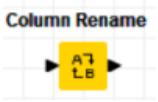
Input line:

	I Nome	S Age	S Data ap...
Row0	Alice	17	23.09.21
Row1	Lucia	23	29.09.21
Row2	Rachele	78	30.09.21
Row3	Luca	34	01.10.21

Figura 3.10 Esempio cambiare tipo di dato a una colonna.

3.5 DATA STRUCTURING

3.5.1 Change column data type (Modifica del tipo di dato delle colonne)



Modificare i tipi di dato può essere utile per rendere più consistente il dataset analizzato.

Per realizzare tale funzionalità in KNIME si possono sfruttare due metodi:

- Si possono modificare i tipi di dato delle colonne direttamente nella configurazione del nodo *File reader.* Sono presenti vari tipi di dato come *Bit vector, Byte vector, Boolean value, Duration, Local date and time, Number (integer, double, long), String, Period, URL*.
- È possibile utilizzare il nodo *Column rename* con cui si può sia rinominare le colonne che modificare i tipi di dato. Nella finestra di configurazione, come si vede in Figura 3.11, si può quindi modificare il tipo di dato di tutte le colonne presenti nel database selezionando uno dei possibili tipi compatibili. Un tipo si definisce compatibile se le celle di una colonna possono essere convertite o trasformate dal tipo attuale ad esso in modo sicuro.

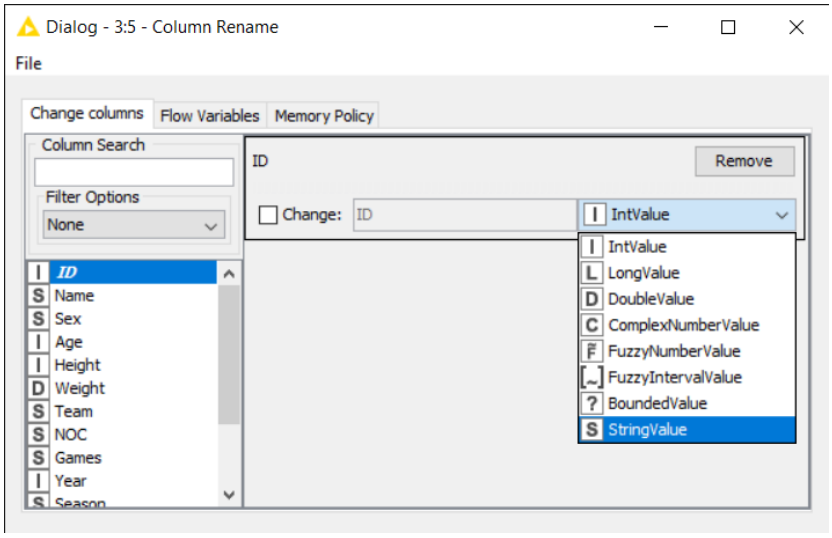
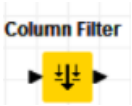


Figura 3.11 Finestra di configurazione del nodo Column rename.

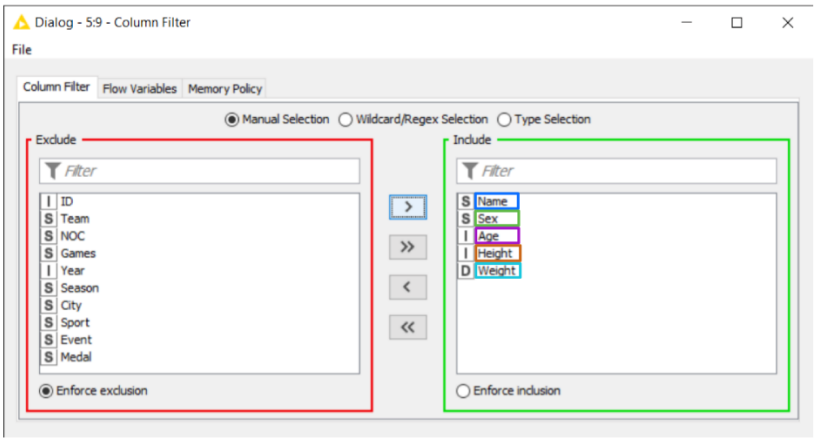
3.5.2 Delete column (Rimozione delle colonne)

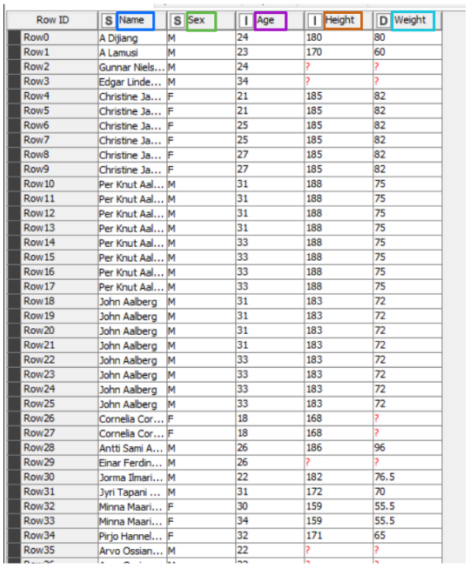


La funzionalità di rimozione delle colonne può essere necessaria se bisogna analizzare solo una parte del dataset tralasciando dei valori.

In KNIME tale funzionalità è implementata dal nodo *Column filter,* che consente di filtrare le colonne della tabella di input passando all'output solo le colonne di interesse. Tramite la finestra di configurazione del nodo è possibile spostare le colonne tra l'elenco *Include* e *Exclude.*

In Figura 3.12 si riporta un esempio di utilizzo del nodo, in cui nel dataset *athletic_events.csv* vengono inclusi solo i dati anagrafici degli atleti (*Name, Sex, Age, Height, Weight*) ed eliminati tutti gli altri valori.

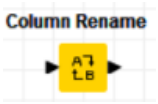




Row ID	Name	Sex	Age	Height	Weight
Row0	A Dijiang	M	24	180	80
Row1	A Lamusi	M	23	170	60
Row2	Gunnar Niels...	M	24	?	?
Row3	Edgar Linde...	M	34	?	?
Row4	Christine Ja...	F	21	185	82
Row5	Christine Ja...	F	21	185	82
Row6	Christine Ja...	F	25	185	82
Row7	Christine Ja...	F	25	185	82
Row8	Christine Ja...	F	27	185	82
Row9	Christine Ja...	F	27	185	82
Row10	Per Knut Aal...	M	31	188	75
Row11	Per Knut Aal...	M	31	188	75
Row12	Per Knut Aal...	M	31	188	75
Row13	Per Knut Aal...	M	31	188	75
Row14	Per Knut Aal...	M	33	188	75
Row15	Per Knut Aal...	M	33	188	75
Row16	Per Knut Aal...	M	33	188	75
Row17	Per Knut Aal...	M	33	188	75
Row18	John Aalberg	M	31	183	72
Row19	John Aalberg	M	31	183	72
Row20	John Aalberg	M	31	183	72
Row21	John Aalberg	M	31	183	72
Row22	John Aalberg	M	33	183	72
Row23	John Aalberg	M	33	183	72
Row24	John Aalberg	M	33	183	72
Row25	John Aalberg	M	33	183	72
Row26	Cornelia Cor...	F	18	168	?
Row27	Cornelia Cor...	F	18	168	?
Row28	Antti Sami A...	M	26	186	96
Row29	Einar Ferdin...	M	26	?	?
Row30	Jorma Ilmari...	M	22	182	76.5
Row31	Jyri Tapani ...	M	31	172	70
Row32	Minna Maari...	F	30	159	55.5
Row33	Minna Maari...	F	34	159	55.5
Row34	Pirjo Hanniel...	F	32	171	65
Row35	Arvo Ossian...	M	22	?	?

Figura 3.12 Esempio di filtraggio delle colonne.

3.5.3 Rename column (Rinominare le colonne)



Le colonne raffigurano dei valori quindi in certi casi, soprattutto quando è presente una grande quantità di dati, è necessario cambiare il nome dell'header per renderle più rappresentative.

Per fare ciò KNIME offre gli stessi meccanismi che utilizza per la modifica dei tipi di dato delle colonne:

- Si possono modificare i nomi delle colonne direttamente nella configurazione del nodo *File reader.* In Figura 3.13 si rinomina la prima colonna del dataset *athletic_events.csv.*

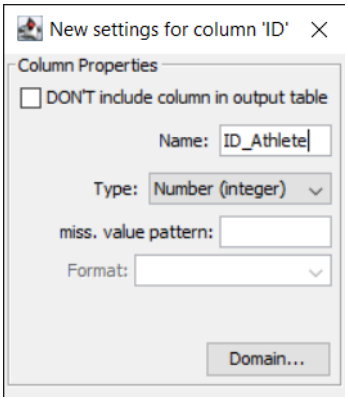
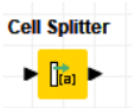


Figura 3.13 Cambio del nome di una colonna nel nodo File reader.

- Oppure si può utilizzare il nodo *Column rename.* Nella finestra di configurazione si può quindi modificare il nome di tutte le colonne presenti nel dataset.

3.5.4 Split column (Divisione delle colonne)



Può succedere che in alcune celle siano presenti più dati divisi da un delimitatore e si vogliano separare tali dati in più colonne.

KNIME esegue questa funzionalità con il nodo *Cell splitter* che utilizza un carattere delimitatore, specificato dall'utente, per dividere in più parti il contenuto di una colonna selezionata, creando tante colonne quanti sono gli elementi divisi dal delimitatore.

In Figura 3.14 si riporta un esempio di questo nodo su una tabella che contiene 3 righe e 3 colonne (Nome, Età e Altezza). Il nodo divide la prima colonna separando il nome dal cognome.

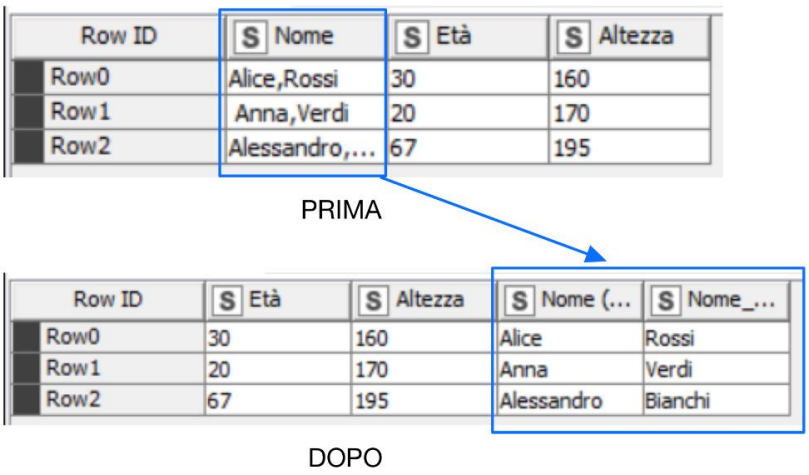
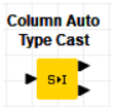


Figura 3.14 Divisione della colonna "Nome".

3.6 DATA ENRICHMENT

3.6.1 Assign semantic data type (Assegnamento del tipo di dato)



Queta operazione consente di rilevare e assegnare il tipo di dato a ogni colonna.

KNIME implementa questa funzione con il nodo *Column auto type cast,* che determina il tipo più specifico nelle colonne delle stringhe e modifica, di conseguenza, il tipo della colonna. L'ordine dei tipi consiste di verificare prima se i valori sono date, poi int,long, double e infine string. Per le date è possibile specificare il formato desiderato.

In figura 3.15 si riporta un esempio di questo nodo.

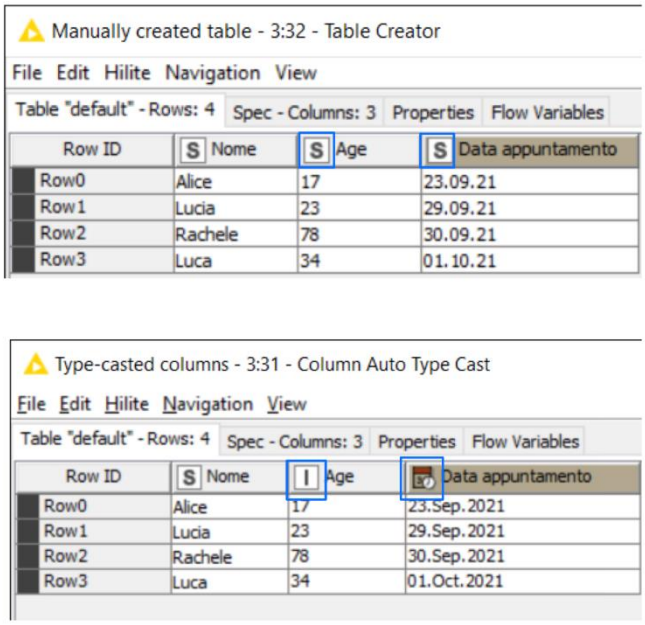


Figura 3.15 Esempio del nodo Column auto type cast.

3.6.2 Calculate column using expression (Calcolare le colonne utilizzando delle espressioni)

Questa funzionalità è molto importante in quanto permette di calcolare i dati tra loro per rendere più esplicite delle informazioni.

KNIME implementa tale funzionalità con il nodo *Column Expression* che appartiene all'estensione *KNIME Expression* da installare.

Questo nodo offre la possibilità di aggiungere un numero arbitrario di colonne o modificare le colonne esistenti utilizzando delle espressioni. Per ogni colonna che deve essere aggiunta o modificata viene definita un'espressione separata.

Si riporta, in Figura 3.16, il database *athletic_events.csv* in cui sono state aggiunte due colonne: *Anno di nascita*, calcolato come sottrazione tra la colonna *Year* e *Age* e *Età attuale*, calcolato come sottrazione tra l'anno corrente e *Anno di nascita*.

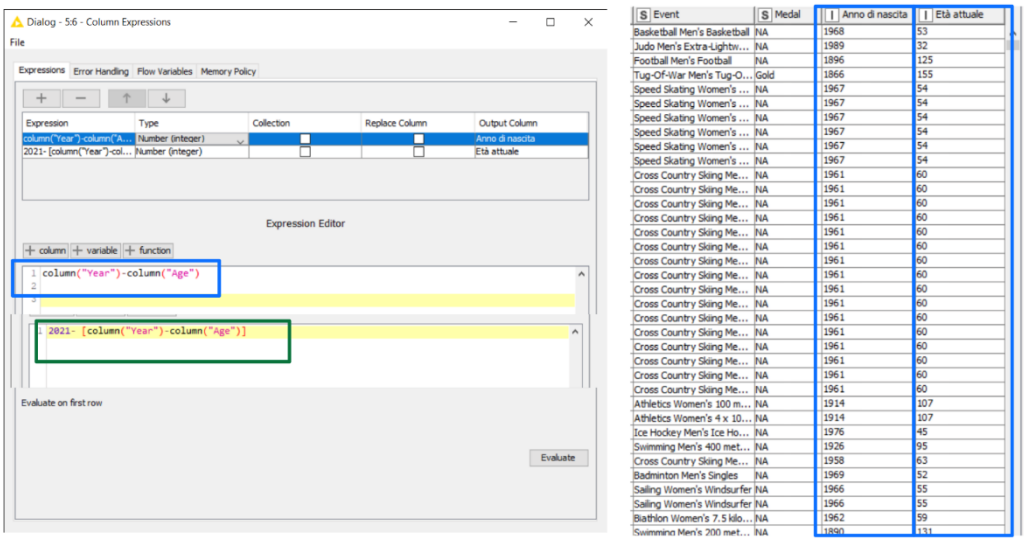


Figura 3.16 Finestra di configurazione del nodo Column Expression.

3.6.3 Duplicate column (Duplicare le colonne)

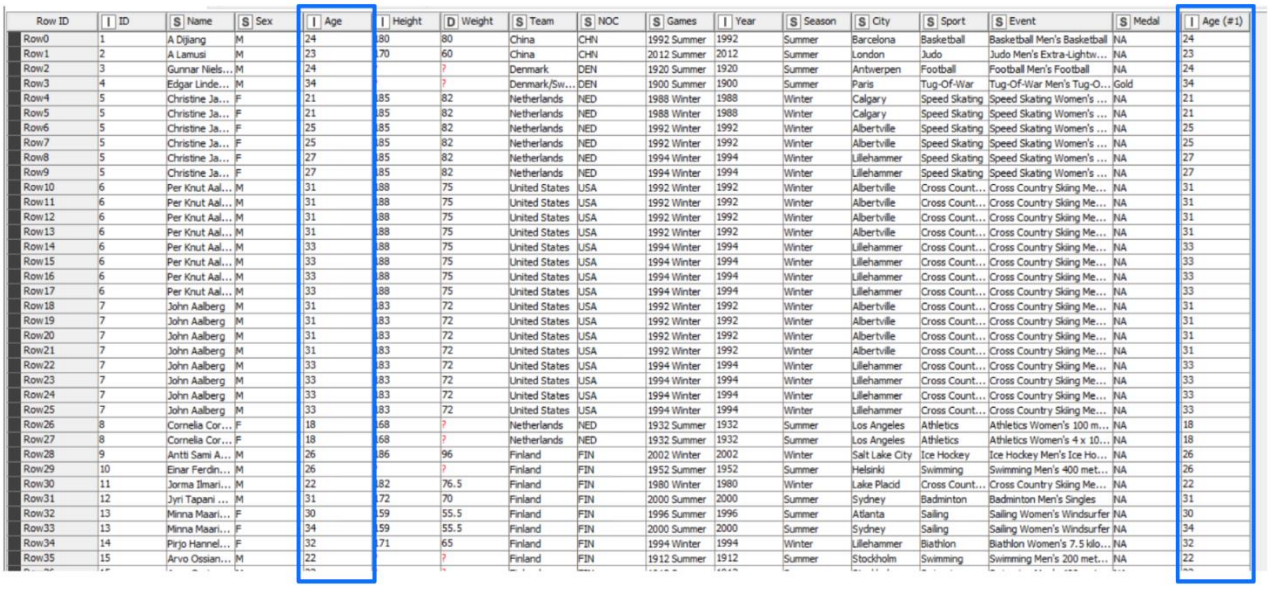
La funzionalità di duplicare le colonne può essere utile se una colonna deve essere modificata ma si vuole comunque tenere quella originale inalterata.

Per fare ciò in KNIME è necessario utilizzare una combinazione di nodi:

- Prima si filtra il database originale con il nodo *Column Filter* (come spiegato nella sezione 3.5.2) includendo solo le colonne che si vogliono duplicare.

- In seguito, si utilizza il nodo *Column appender* che prende in input la tabella originale e la tabella filtrata e le combina, unendo le colonne di entrambe.

Si riporta, in Figura 3.17, l'utilizzo di questi due nodi applicati al database *athletic_events.csv* in cui viene duplicata la colonna *Age.*



Row ID	ID	Name	Sex	Age	Height	Weight	Team	NOC	Games	Year	Season	City	Sport	Event	Medal	Age (#1)
Row0	1	A Dijiang	M	24	180	80	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Basketball Men's Basketball	NA	24
Row1	2	A Lamusi	M	23	170	60	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Men's Extra-Lightw...	NA	23
Row2	3	Gunnar Niels...	M	24		?	Denmark	DEN	1920 Summer	1920	Summer	Antwerpen	Football	Football Men's Football	NA	24
Row3	4	Edgar Linde...	M	34		?	Denmark/Sw...	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's Tug-O...	Gold	34
Row4	5	Christine Ja...	F	21	185	82	Netherlands	NED	1988 Winter	1988	Winter	Calgary	Speed Skating	Speed Skating Women's ...	NA	21
Row5	5	Christine Ja...	F	21	185	82	Netherlands	NED	1988 Winter	1988	Winter	Calgary	Speed Skating	Speed Skating Women's ...	NA	21
Row6	5	Christine Ja...	F	25	185	82	Netherlands	NED	1992 Winter	1992	Winter	Albertville	Speed Skating	Speed Skating Women's ...	NA	25
Row7	5	Christine Ja...	F	25	185	82	Netherlands	NED	1992 Winter	1992	Winter	Albertville	Speed Skating	Speed Skating Women's ...	NA	25
Row8	5	Christine Ja...	F	27	185	82	Netherlands	NED	1994 Winter	1994	Winter	Lillehammer	Speed Skating	Speed Skating Women's ...	NA	27
Row9	5	Christine Ja...	F	27	185	82	Netherlands	NED	1994 Winter	1994	Winter	Lillehammer	Speed Skating	Speed Skating Women's ...	NA	27
Row10	6	Per Knut Aal...	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA	31
Row11	6	Per Knut Aal...	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA	31
Row12	6	Per Knut Aal...	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA	31
Row13	6	Per Knut Aal...	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA	31
Row14	6	Per Knut Aal...	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA	33
Row15	6	Per Knut Aal...	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA	33
Row16	6	Per Knut Aal...	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA	33
Row17	6	Per Knut Aal...	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA	33
Row18	7	John Aalberg	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA	31
Row19	7	John Aalberg	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA	31
Row20	7	John Aalberg	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA	31
Row21	7	John Aalberg	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...	NA	31
Row22	7	John Aalberg	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA	33
Row23	7	John Aalberg	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA	33
Row24	7	John Aalberg	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA	33
Row25	7	John Aalberg	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...	NA	33
Row26	8	Cornelia Cor...	F	18	168	?	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 100 m...	NA	18
Row27	8	Cornelia Cor...	F	18	168	?	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 4 x 10...	NA	18
Row28	9	Antti Sami A...	M	26	186	96	Finland	FIN	2002 Winter	2002	Winter	Salt Lake City	Ice Hockey	Ice Hockey Men's Ice Ho...	NA	26
Row29	10	Einar Ferdin...	M	26		?	Finland	FIN	1952 Summer	1952	Summer	Helsinki	Swimming	Swimming Men's 400 met...	NA	26
Row30	11	Jorma Ilmari...	M	22	182	76.5	Finland	FIN	1980 Winter	1980	Winter	Lake Placid	Cross Count...	Cross Country Skiing Me...	NA	22
Row31	12	Jyri Tapani ...	M	31	172	70	Finland	FIN	2000 Summer	2000	Summer	Sydney	Badminton	Badminton Men's Singles	NA	31
Row32	13	Minna Maari...	F	30	159	55.5	Finland	FIN	1996 Summer	1996	Summer	Atlanta	Sailing	Sailing Women's Windsurfer	NA	30
Row33	13	Minna Maari...	F	34	159	55.5	Finland	FIN	2000 Summer	2000	Summer	Sydney	Sailing	Sailing Women's Windsurfer	NA	34
Row34	14	Pirjo Hannel...	F	32	171	65	Finland	FIN	1994 Winter	1994	Winter	Lillehammer	Biathlon	Biathlon Women's 7.5 kilo...	NA	32
Row35	15	Arvo Ossian...	M	22		?	Finland	FIN	1912 Summer	1912	Summer	Stockholm	Swimming	Swimming Men's 200 met...	NA	22

Figura 3.17 Tabella con colonna "Age" duplicata.

3.6.4 Join and Union

Queste due funzionalità servono per concatenare insieme due o più database.

Per la funzionalità di Join KNIME utilizza il nodo *Joiner* che unisce due tabelle in base a uno o più valori comuni. Le modalità di join supportate sono *Inner join, left outer join, right outer join, full outer join.*

Per quanto riguarda la funzionalità Union, KNIME utilizza il nodo *Concatenate* che non fa altro che concatenare assieme due tabelle. Il nodo accetta due input e i record della tabella che è passata all'ingresso 0 sono posizionati all'inizio della tabella di output. Le colonne con nomi uguali vengono concatenate e se una colonna non è presente in una tabella allora le celle corrispondenti a tale tabella verranno marcate con il valore nullo.

In Figura 3.18 si riportano due tabelle con tre righe e tre colonne ciascuna per verificare il funzionamento del nodo.

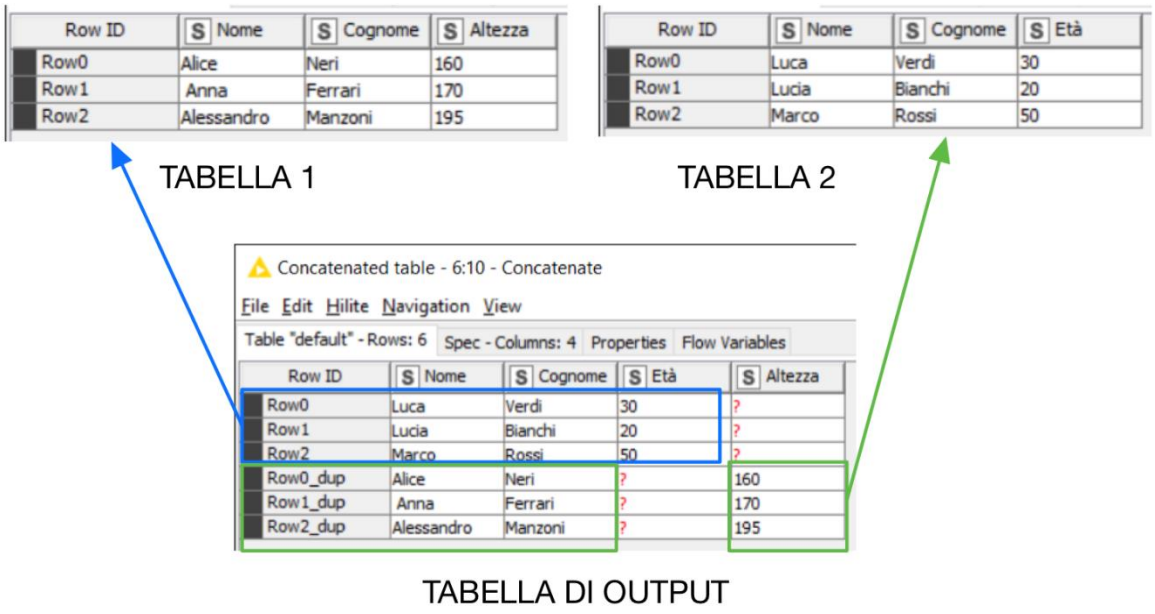
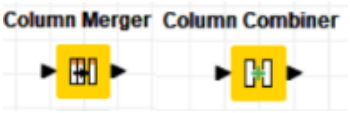


Figura 3.18 Esempio del nodo Concatenate.

3.6.5 Merge column (Unire le colonne)



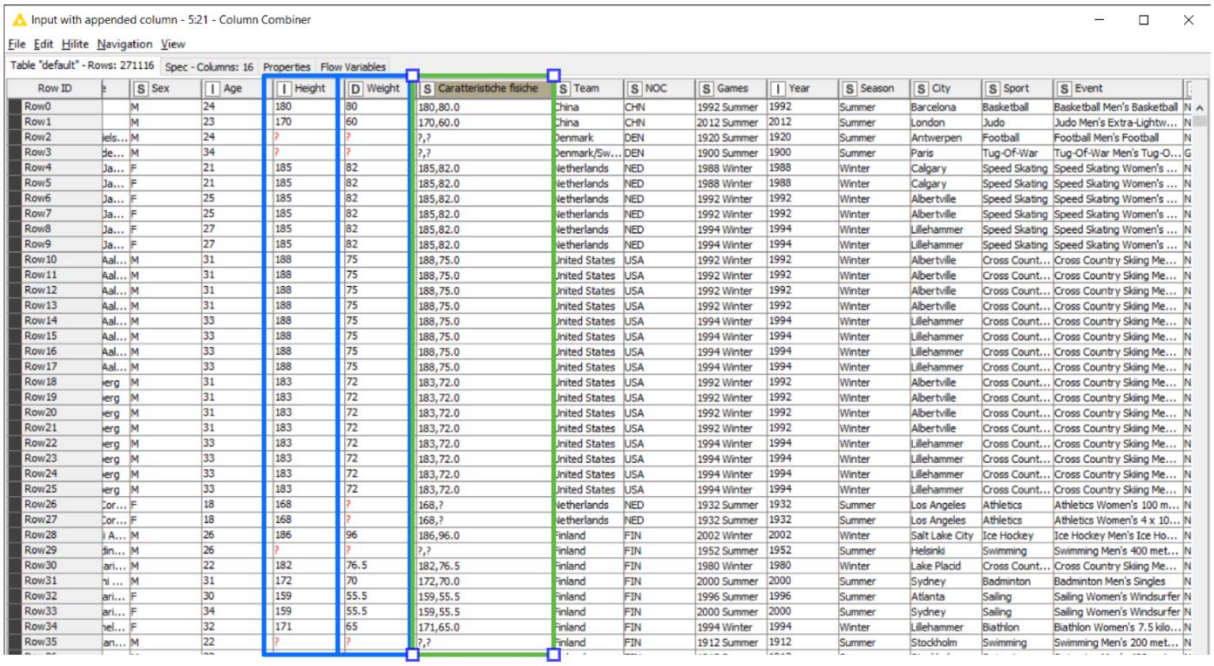
Unire in un'unica colonna più dati, divisi in colonne differenti, può essere necessario quando abbiamo delle informazioni che sono più comprensibili se considerate insieme (come, ad esempio, una data divisa in tre colonne differenti che indicano giorno, mese, anno).

In KNIME sono presenti due nodi che uniscono tra loro le colonne, con funzionalità differenti:

- Il nodo *Column merge* unisce due colonne in una. Nella finestra di configurazione si deve scegliere la colonna primaria e quella secondaria e l'output del nodo sarà una nuova colonna in cui il valore per ogni riga è il valore della colonna primaria e se esso manca il valore della colonna secondaria.

- Il nodo *Column combiner,* invece combina il contenuto di più colonne, decise dall'utente in fase di configurazione, in una nuova colonna che contiene tutti i valori separati da un delimitatore sempre deciso dall'utente.

Si riporta l'esempio, in Figura 3.19, del nodo *Column combiner*, in cui vengono unite le colonne *Weight* e *Height* nella colonna *Caratteristiche fisiche* del database *athletic_events.csv.*

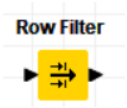


Row ID	Sex	Age	Height	Weight	Caratteristiche fisiche	Team	NOC	Games	Year	Season	City	Sport	Event
Row0	M	24	180	80	180,80.0	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Basketball Men's Basketball
Row1	M	23	170	60	170,60.0	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Men's Extra-Lightw...
Row2	M	24	?	?	?,?	Denmark	DEN	1920 Summer	1920	Summer	Antwerpen	Football	Football Men's Football
Row3	M	34	?	?	?,?	Denmark/Sw...	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's Tug-O...
Row4	F	21	185	82	185,82.0	Netherlands	NED	1988 Winter	1988	Winter	Calgary	Speed Skating	Speed Skating Women's ...
Row5	F	21	185	82	185,82.0	Netherlands	NED	1988 Winter	1988	Winter	Calgary	Speed Skating	Speed Skating Women's ...
Row6	F	25	185	82	185,82.0	Netherlands	NED	1992 Winter	1992	Winter	Albertville	Speed Skating	Speed Skating Women's ...
Row7	F	25	185	82	185,82.0	Netherlands	NED	1992 Winter	1992	Winter	Albertville	Speed Skating	Speed Skating Women's ...
Row8	F	27	185	82	185,82.0	Netherlands	NED	1994 Winter	1994	Winter	Lillehammer	Speed Skating	Speed Skating Women's ...
Row9	F	27	185	82	185,82.0	Netherlands	NED	1994 Winter	1994	Winter	Lillehammer	Speed Skating	Speed Skating Women's ...
Row10	M	31	188	75	188,75.0	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...
Row11	M	31	188	75	188,75.0	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...
Row12	M	31	188	75	188,75.0	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...
Row13	M	31	188	75	188,75.0	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...
Row14	M	33	188	75	188,75.0	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...
Row15	M	33	188	75	188,75.0	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...
Row16	M	33	188	75	188,75.0	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...
Row17	M	33	188	75	188,75.0	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...
Row18	M	31	183	72	183,72.0	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...
Row19	M	31	183	72	183,72.0	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...
Row20	M	31	183	72	183,72.0	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...
Row21	M	31	183	72	183,72.0	United States	USA	1992 Winter	1992	Winter	Albertville	Cross Count...	Cross Country Skiing Me...
Row22	M	33	183	72	183,72.0	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...
Row23	M	33	183	72	183,72.0	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...
Row24	M	33	183	72	183,72.0	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...
Row25	M	33	183	72	183,72.0	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross Count...	Cross Country Skiing Me...
Row26	F	18	168	?	168,?	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 100 m...
Row27	F	18	168	?	168,?	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 4 x 10...
Row28	M	26	186	96	186,96.0	Finland	FIN	2002 Winter	2002	Winter	Salt Lake City	Ice Hockey	Ice Hockey Men's Ice Ho...
Row29	M	26	?	?	?,?	Finland	FIN	1952 Summer	1952	Summer	Helsinki	Swimming	Swimming Men's 400 met...
Row30	M	22	182	76.5	182,76.5	Finland	FIN	1980 Winter	1980	Winter	Lake Placid	Cross Count...	Cross Country Skiing Me...
Row31	M	31	172	70	172,70.0	Finland	FIN	2000 Summer	2000	Summer	Sydney	Badminton	Badminton Men's Singles
Row32	F	30	159	55.5	159,55.5	Finland	FIN	1996 Summer	1996	Summer	Atlanta	Sailing	Sailing Women's Windsurfer
Row33	F	34	159	55.5	159,55.5	Finland	FIN	2000 Summer	2000	Summer	Sydney	Sailing	Sailing Women's Windsurfer
Row34	F	32	171	65	171,65.0	Finland	FIN	1994 Winter	1994	Winter	Lillehammer	Biathlon	Biathlon Women's 7.5 kilo...
Row35	M	22	?	?	?,?	Finland	FIN	1912 Summer	1912	Summer	Stockholm	Swimming	Swimming Men's 200 met...

Figura 3.19 Unione delle colonne Weight e Height.

3.7 DATA FILTERING

3.7.1 Delete/keep filtered rows (Eliminare/tenere colonne filtrate)



Questa funzionalità è utile quando si vogliono analizzare solo determinati valori, scartandone altri.

In KNIME è possibile implementare tale funzionalità grazie al nodo *Row filter*, che consente di filtrare le righe in base a determinati criteri. I valori possono essere esclusi o inclusi secondo specificati intervalli, secondo un determinato ID di riga o secondo un certo valore in una colonna. Ogni nodo gestisce un filtro quindi se si vogliono unire più filtri è necessario mettere in cascata più nodi *Row filter* come mostrato in seguito.

In figura 3.20 viene mostrata la finestra di configurazione del nodo

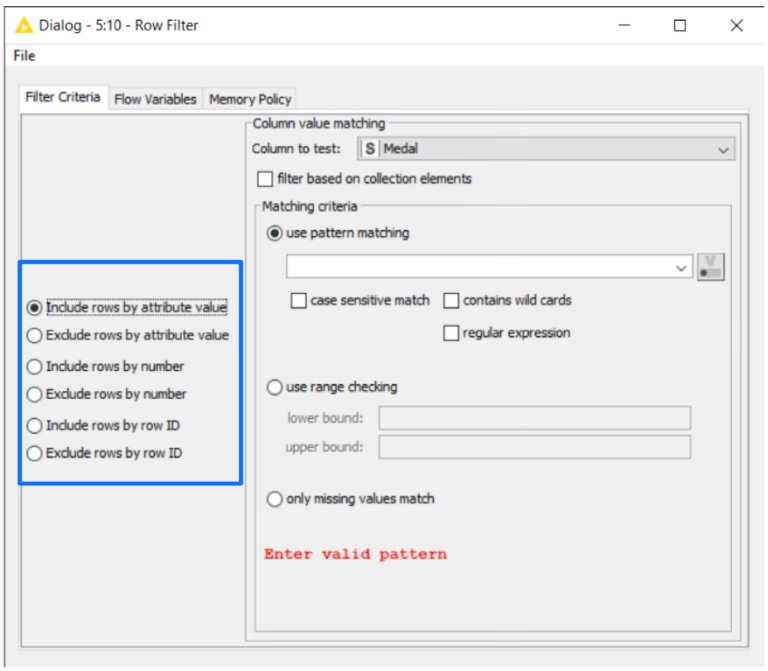


Figura 3.20 Finestra di configurazione del nodo Row filter.

Si riporta, in Figura 3.21, un esempio di funzionamento del nodo con il database *athletic_events.csv* al fine di mantenere solo il sottoinsieme degli atleti italiani che hanno vinto una medaglia d'oro ai Giochi Olimpici invernali. In questo caso sono serviti 3 nodi *Row filter.* Come si evince dalla figura, il numero di righe si è notevolmente ridotto.

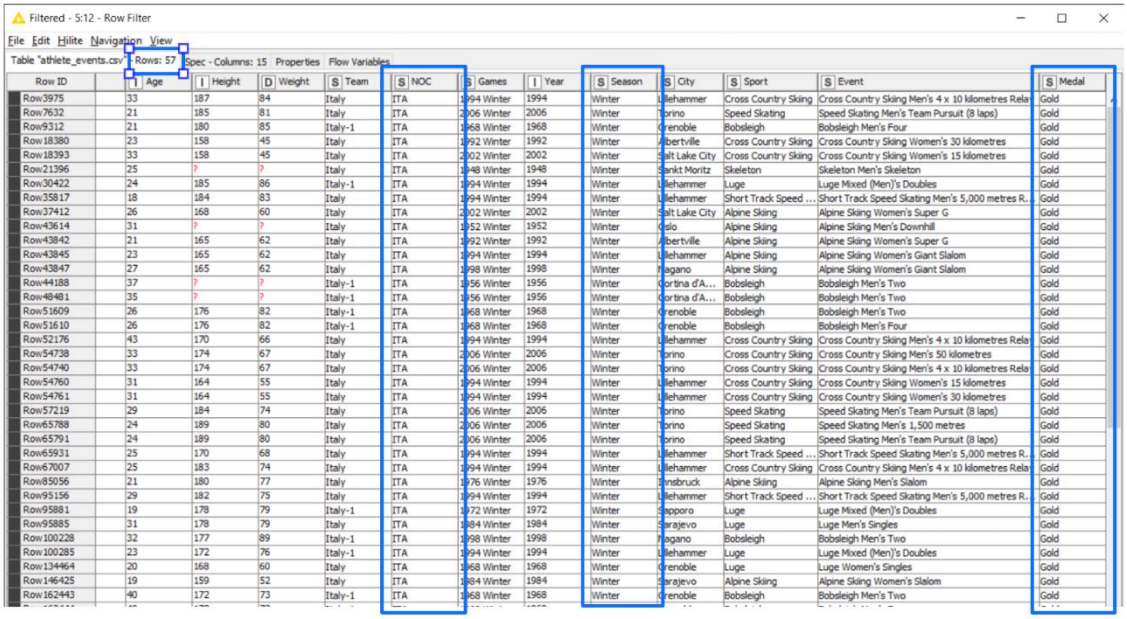


Figura 3.21 Tabella risultante da un nodo Row filter.

3.7.2 Delete empty and invalid rows (Eliminare righe vuote e non valide)

L'eliminazione delle righe vuote in KNIME viene implementata con il nodo *Remove empty rows,* che rimuove le righe vuote dalla tabella di input. Una riga vuota può essere definita o come una riga che ha solo celle vuote, o una riga che non presenta colonne; nella configurazione del nodo si possono gestire entrambe le definizioni.

In figura 3.22 viene presentato un esempio di questo nodo.

Table "default" - Rows: 7 | Spec - Columns: 3 | Properties | Flow Va

Row ID	S column1	S column2	S column3
Row0	a	a	?
Row1	a	?	a
Row2	a	a	a
Row3	?	?	?
Row4	a	a	a
Row5	?	?	?
Row6	?	a	a

PRIMA

Table "default" - Rows: 5 | Spec - Columns: 3 | Properties | Flow Va

Row ID	S column1	S column2	S column3
Row0	a	a	?
Row1	a	?	a
Row2	a	a	a
Row4	a	a	a
Row6	?	a	a

DOPO

Figura 2.22 Esempio del nodo Remove empty rows.

Per l'eliminazione delle righe non valide non sono presenti specifici nodi in KNIME.

3.8 DATA CLEANING

3.8.1 Change date & time format (Cambiare il formato della data e dell'ora)

La data e l'ora non sempre sono memorizzate in un unico formato: a volte possono essere memorizzate come stringa e altre volte possono essere memorizzate in un formato diverso da quello desiderato.

In KNIME è possibile:

- Trasformare una stringa in un formato Date & Time utilizzando il nodo *String to Date & Time*, in cui vengono analizzate le stringhe nelle colonne selezionate e convertite in celle Date & Time utilizzando il formato fornito. È presente anche il nodo inverso, *Date & Time to String.*
  In Figura 3.23 viene trasformata una colonna da formato *String* a *Date & Time.*

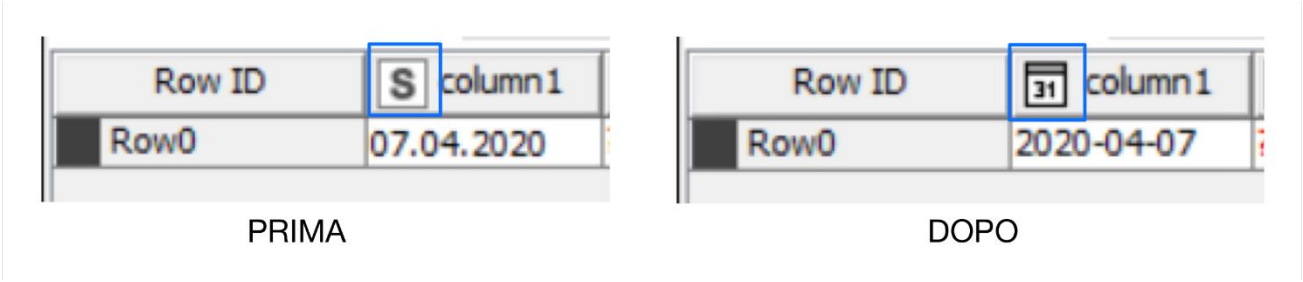


Figura 3.23 Trasformazione di una colonna da tipo String a tipo Date e Time.

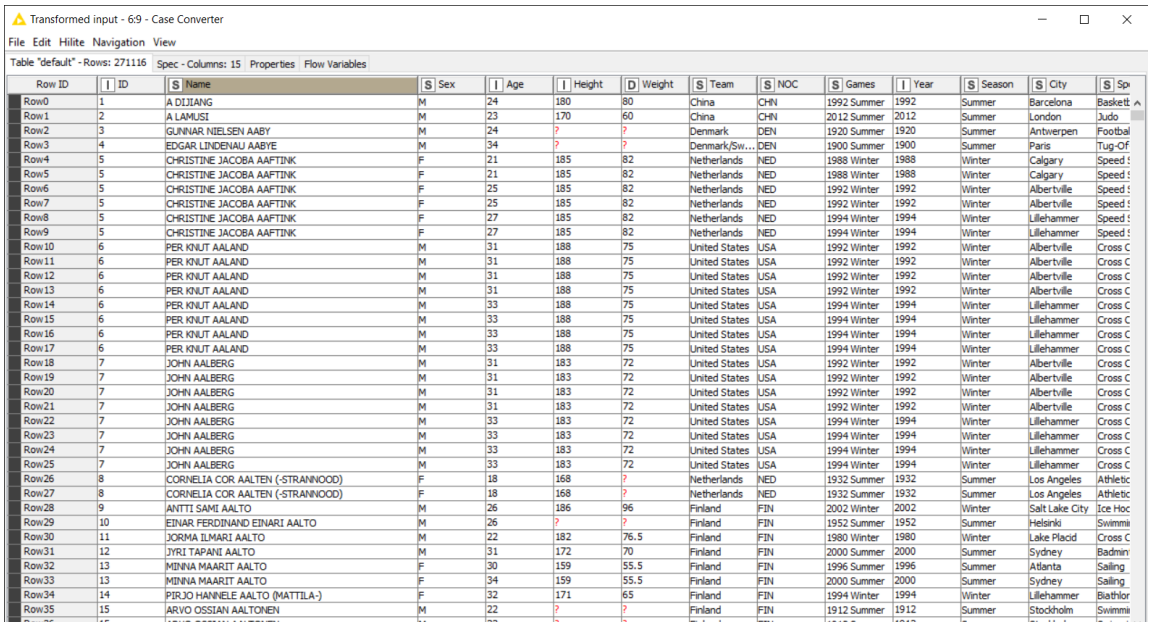
- Passare dal nuovo tipo di dato Date & Time al tipo legacy utilizzando il *nodo Date & Time to legacy Date & Time* e per fare l'operazione inversa bisogna utilizzare il nodo *Legacy Date & Time to Date & Time.*

3.8.2 Change letter case (Cambiare da maiuscolo a minuscolo e viceversa le lettere)

Poter modificare un testo cambiando tutte le lettere da maiuscolo a minuscolo (e viceversa) è utile per non doversi preoccupare in seguito se si eseguono delle operazioni che sono case-sensitive.

In KNIME esiste il nodo *Case convert*.

In Figura 3.24 si riporta l'utilizzo di questo nodo sul database *athletic_events.csv* in cui la colonna *Name* è stata trasformata in maiuscolo.



Row ID	ID	Name	Sex	Age	Height	Weight	Team	NOC	Games	Year	Season	City	Sp
Row0	1	A DIJIANG	M	24	180	80	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketb
Row1	2	A LAMUSI	M	23	170	60	China	CHN	2012 Summer	2012	Summer	London	Judo
Row2	3	GUNNAR NIELSEN AABY	M	24	?	?	Denmark	DEN	1920 Summer	1920	Summer	Antwerpen	Footbal
Row3	4	EDGAR LINDENAU AABYE	M	34	?	?	Denmark/Sw...	DEN	1900 Summer	1900	Summer	Paris	Tug-Of
Row4	5	CHRISTINE JACOBA AAFTINK	F	21	185	82	Netherlands	NED	1988 Winter	1988	Winter	Calgary	Speed S
Row5	5	CHRISTINE JACOBA AAFTINK	F	21	185	82	Netherlands	NED	1988 Winter	1988	Winter	Calgary	Speed S
Row6	5	CHRISTINE JACOBA AAFTINK	F	25	185	82	Netherlands	NED	1992 Winter	1992	Winter	Albertville	Speed S
Row7	5	CHRISTINE JACOBA AAFTINK	F	25	185	82	Netherlands	NED	1992 Winter	1992	Winter	Albertville	Speed S
Row8	5	CHRISTINE JACOBA AAFTINK	F	27	185	82	Netherlands	NED	1994 Winter	1994	Winter	Lillehammer	Speed S
Row9	5	CHRISTINE JACOBA AAFTINK	F	27	185	82	Netherlands	NED	1994 Winter	1994	Winter	Lillehammer	Speed S
Row10	6	PER KNUT AALAND	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross C
Row11	6	PER KNUT AALAND	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross C
Row12	6	PER KNUT AALAND	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross C
Row13	6	PER KNUT AALAND	M	31	188	75	United States	USA	1992 Winter	1992	Winter	Albertville	Cross C
Row14	6	PER KNUT AALAND	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross C
Row15	6	PER KNUT AALAND	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross C
Row16	6	PER KNUT AALAND	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross C
Row17	6	PER KNUT AALAND	M	33	188	75	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross C
Row18	7	JOHN AALBERG	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross C
Row19	7	JOHN AALBERG	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross C
Row20	7	JOHN AALBERG	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross C
Row21	7	JOHN AALBERG	M	31	183	72	United States	USA	1992 Winter	1992	Winter	Albertville	Cross C
Row22	7	JOHN AALBERG	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross C
Row23	7	JOHN AALBERG	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross C
Row24	7	JOHN AALBERG	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross C
Row25	7	JOHN AALBERG	M	33	183	72	United States	USA	1994 Winter	1994	Winter	Lillehammer	Cross C
Row26	8	CORNELIA COR AALTEN (-STRANNOOD)	F	18	168	?	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletic
Row27	8	CORNELIA COR AALTEN (-STRANNOOD)	F	18	168	?	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletic
Row28	9	ANTTI SAMI AALTO	M	26	186	96	Finland	FIN	2002 Winter	2002	Winter	Salt Lake City	Ice Hoc
Row29	10	EINAR FERDINAND EINARI AALTO	M	26	?	?	Finland	FIN	1952 Summer	1952	Summer	Helsinki	Swimmi
Row30	11	JORMA ILMARI AALTO	M	22	182	76.5	Finland	FIN	1980 Winter	1980	Winter	Lake Placid	Cross C
Row31	12	JYRI TAPANI AALTO	M	31	172	70	Finland	FIN	2000 Summer	2000	Summer	Sydney	Badmin
Row32	13	MINNA MAARIT AALTO	F	30	159	55.5	Finland	FIN	1996 Summer	1996	Summer	Atlanta	Sailing
Row33	13	MINNA MAARIT AALTO	F	34	159	55.5	Finland	FIN	2000 Summer	2000	Summer	Sydney	Sailing
Row34	14	PIRJO HANNELE AALTO (MATTILA-)	F	32	171	65	Finland	FIN	1994 Winter	1994	Winter	Lillehammer	Biathlon
Row35	15	ARVO OSSIAN AALTONEN	M	22	?	?	Finland	FIN	1912 Summer	1912	Summer	Stockholm	Swimmi

Figura 3.24 Tabella risultante del nodo Case convert sulla colonna Name.

3.8.3 Change number format (Cambiare il formato dei numeri)

Poter modificare la visualizzazione dei dati è utile per migliorare la leggibilità del dataset considerato.

In KNIME è possibile modificare direttamente questo parametro sulla tabella di output di un nodo cliccando col tasto destro sulla colonna che si vuole modificare. Le varie opzioni sono mostrate in Figura 3.25.

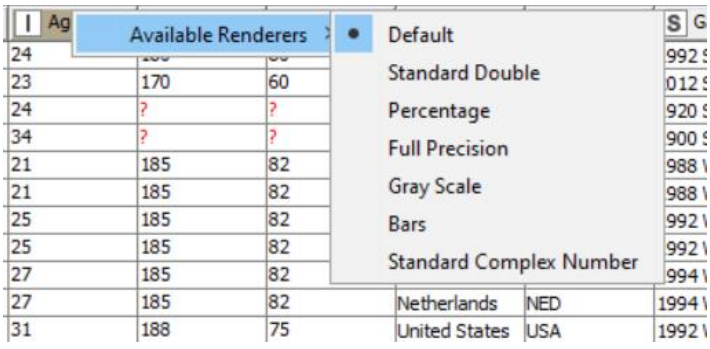
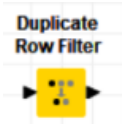


Figura 3.25 Vari formati dei numeri.

3.8.4 Deduplicate data (Rimuovere dati duplicati)



Eliminare i valori duplicati è un'operazione molto importante per la data preparation.

KNIME utilizza il nodo *Duplicate Row filter* per eliminare le righe duplicate. Una volta scelte le colonne per rilevare i duplicati, questo nodo riconosce tutte le righe duplicate nella tabella di input. È possibile rimuovere tutte le righe duplicate e mantenere solo le righe univoche oppure contrassegnare le righe con delle informazioni aggiuntive sul loro stato (unique, duplicate).

In Figura 3.26 si riporta una tabella del dataset *noc_regions.csv* in cui vengono eliminate le righe che contengono lo stesso Stato nella colonna *Col1.*

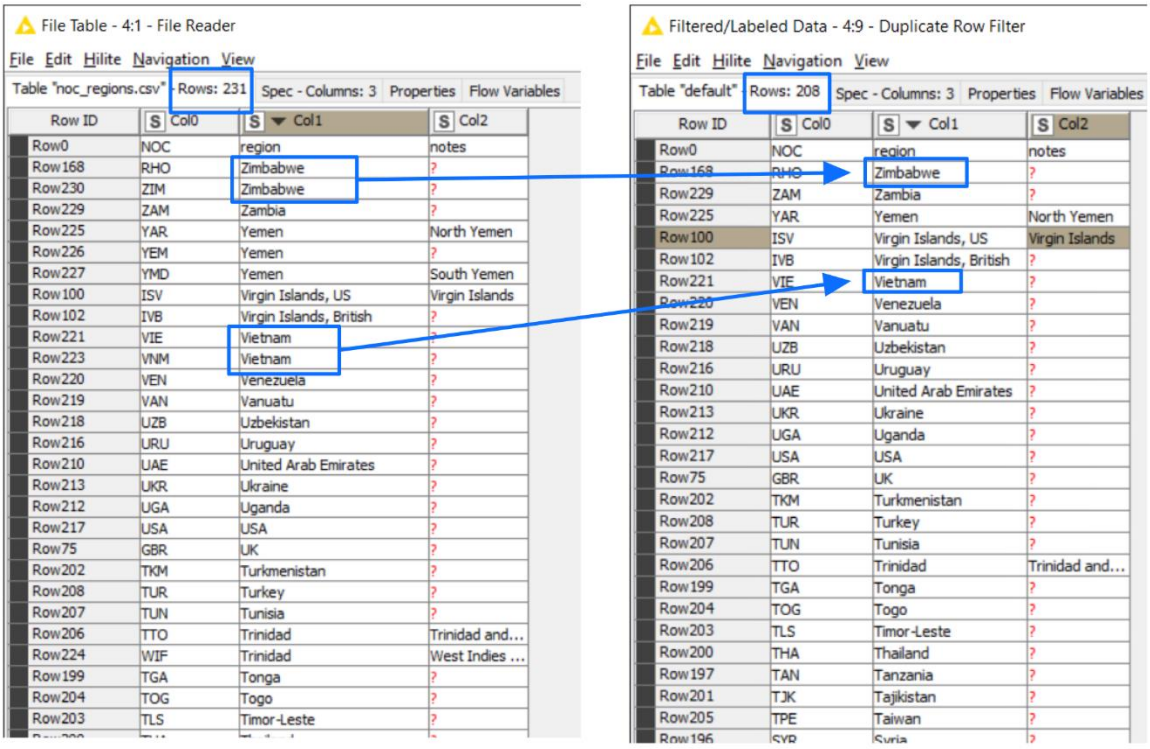
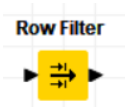


Figura 3.26 Esempio del nodo Row filter.

3.8.5 Delete by pattern (Eliminare in base a dei valori definiti)



Per velocizzare la pulizia del database può essere utile dover eliminare tutti i dati che corrispondo a un certo valore.

In KNIME è possibile utilizzare il nodo *Row filter*, come descritto nella sezione 3.7.1, selezionando nella finestra di configurazione *Exclude rows by attribute value*, eliminando tutte le righe in cui è presente un valore che corrisponde al pattern deciso dall'utente.

In Figura 3.27 vengono eliminati tutti gli atleti che non hanno vinto una medaglia, cioè tutti i valori contrassegnati con "NA" nella colonna *Medal.*

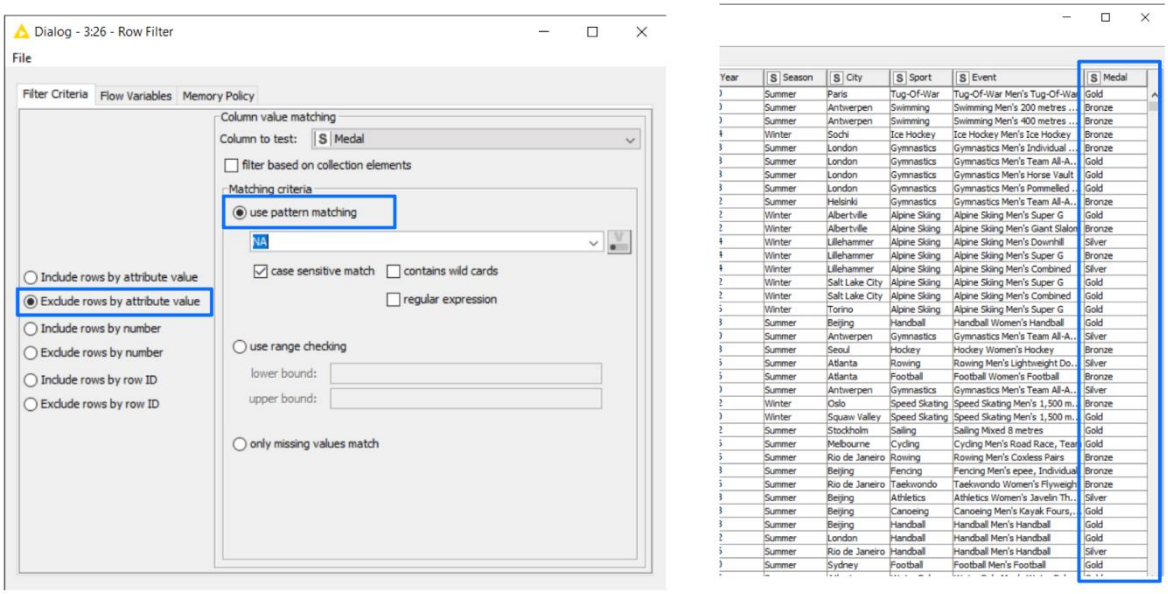


Figura 3.27 Passaggi per eliminare le righe con un certo valore.

3.8.6 Fill empty cell (Popolare le celle vuote)

KNIME implementa la funzionalità di riempire le celle vuote utilizzando il nodo *Missing value* che aiuta a gestire i valori nulli trovati nella tabella di input. Dalla finestra di configurazione è possibile decidere che azione effettuare quando si trova una cella vuota: sostituirlo con il valore più frequente

nella colonna, non fare nulla, sostituirlo con un valore deciso dall'utente, eliminare la riga corrispondente, sostituirlo con il valore precedente o con il valore successivo.

In Figura 3.28 vengono sostituite le celle vuote delle colonne *Height* e *Weight*, del dataset *athletic_events.csv*, con il valore -1.

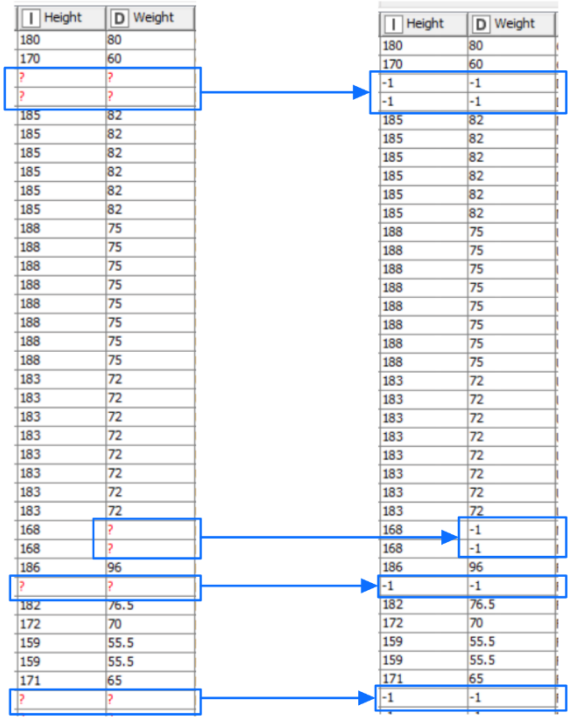
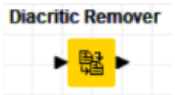


Figura 3.28 Gli elementi nulli della tabella vengono sostituiti.

3.8.7 Remove extrawhite space (Rimuove spazi bianchi aggiuntivi)

Rimuovere spazi bianchi è necessario per avere un database più pulito e leggibile. KNIME non presenta però un nodo che implementi direttamente questa funzionalità.

3.8.8 Remove diacritics (Rimuovere segni diacritici)



I segni diacritici sono quei segni aggiunti a un carattere per modificarne la pronuncia o per distinguere il significato di parole simili.

Rimuovere dei segni speciali come gli accenti è utile per rendere più leggibile e utilizzabile il database. Inoltre, a volte i dati non vengono interpretati correttamente in presenza di alcuni di questi segni.

In KNIME tale funzionalità è implementata dal nodo *Diacritics remove* che è presente nell'espansione *KNIME Textprocessing.* Questo nodo rimuove tutti i segni diacritici nei documenti forniti. Dato che il nodo riceve in input solo documenti, prima di collegarlo al database è necessario aggiungere, tra i due, il nodo *String to document* che converte le stringhe di una colonna selezionata in un documento creando una nuova colonna.

Si riporta un esempio, in Figura 3.29, di questi due nodi.

Row ID	S 1	Document	Senza segni diacritici
Row0	Chloë	"Chloë"	"Chloe"
Row1	Brontë	"Brontë"	"Bronte"
Row2	piña	"piña"	"pina"
Row3	caffè	"caffè"	"caffe"

Figura 3.29 Tabella risultante dei nodi String to Document e Remove diacritics

4. WORKFLOW MANAGEMENT IN KNIME

Oltre adi nodi per le operazioni di data preparation, KNIME mette a disposizione apposite sezioni di nodi dedicati alla visualizzazione dei dati e al controllo di flusso.

4.1 DATA VISUALIZATION

La visualizzazione dei dati (*data visualization*) è la traduzione del contenuto di un dataset in elementi visivi come diagrammi, grafici e mappe. La rappresentazione visiva dei dati semplifica l'identificazione di tendenze, valori anomali (*outliers*), ricorrenze e nuove informazioni.

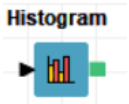
Con l'aumento della quantità di dati disponibili, sempre più utenti utilizzano strumenti di visualizzazione dei dati per accedere a informazioni dettagliate, come Microsoft Power BI oTableau Desktop.

KNIME offre dei nodi basati su Javascript per l'indagine visiva, che consentono di creare grafici classici come ad esempio, il grafico a barre o il grafico a dispersione, ma anche alcuni grafici più innovativi come grafico a raggiera.

Figura 3.1 Esempi di grafici in KNIME

Tutti i nodi per la data visualization vengono raggruppati nel *Node repository* nella categoria *Views;* di seguito si riportano i più rilevanti.

4.1.1 Histogram (Istogramma)



Il nodo *Histogram* viene utilizzato per esaminare una variabile continua. I dati vengono divisi in insiemi (*bin*) e vengono tracciate le occorrenze all'interno di ogni bin. Il nodo è basato sulla libreria NVD3 e supporta lo stile CSS.

In Figura 4.2 si riporta la visualizzazione della colonna Age del dataset *athletic_events.csv.*

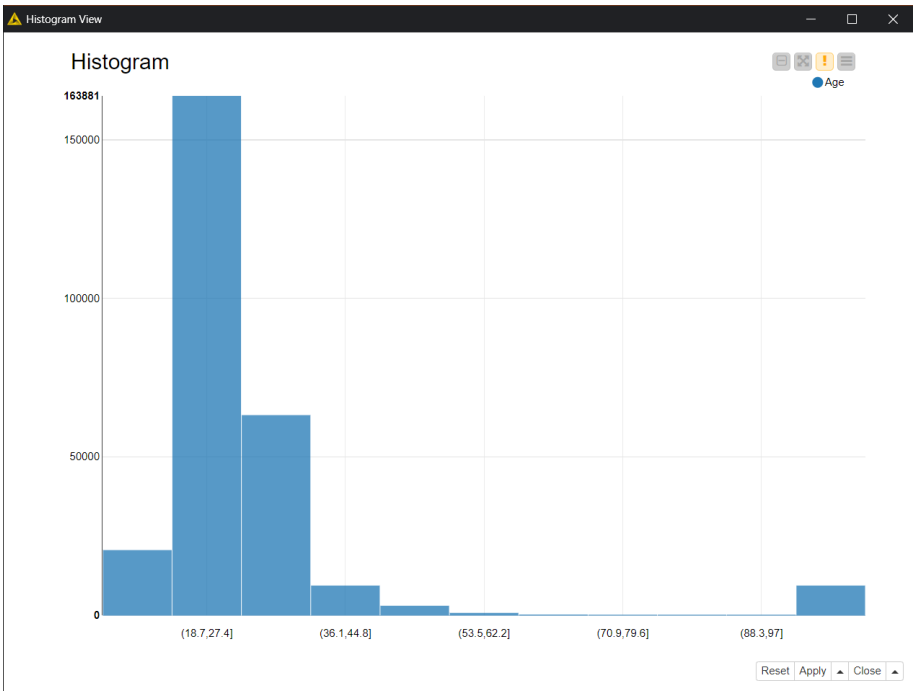


Figura 4.2 Istogramma.

4.1.2 Scatter Plot (Grafico a dispersione)

Un grafico a dispersione è un grafico in cui due variabili di un dataset sono riportate in uno spazio cartesiano. I dati sono visualizzati tramite una collezione di punti ciascuno con una posizione sull'asse orizzontale determinato da una variabile e sull'asse verticale determinato dall'altra.

La configurazione del nodo *Scatter Plot* permette di scegliere le colonne per x e y; i valori mancanti, così come NaN o valori infiniti non possono essere rappresentati nella vista e vengono omessi con un messaggio di avviso corrispondente.

In Figura 4.3 troviamo ad esempio la variabile *Height* e la variabile *Weight* del dataset *athletic_events.csv* rispettivamente sull'asse delle ascisse e sull'asse delle ordinate.

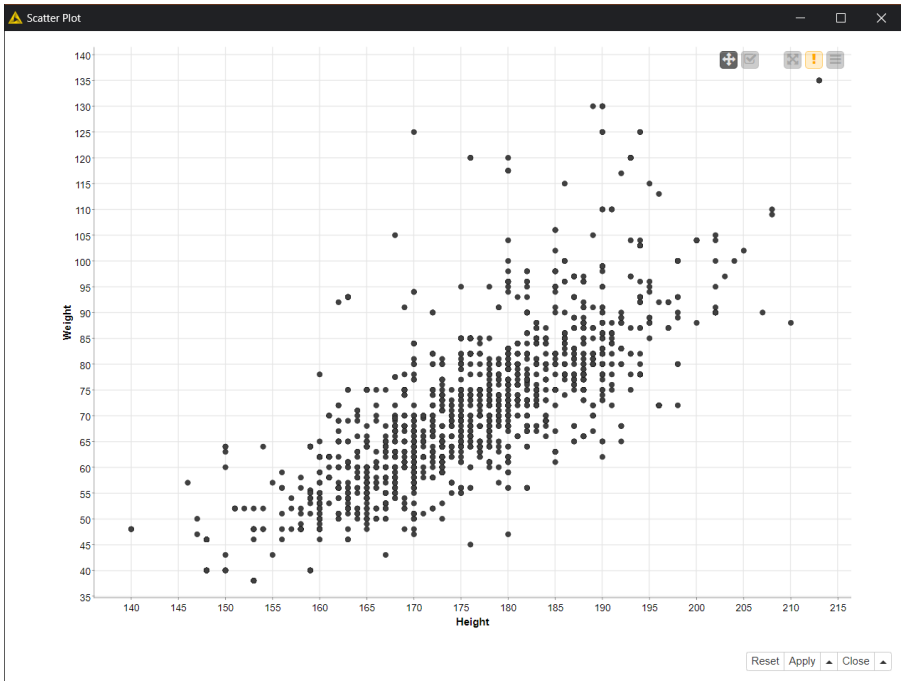


Figura 4.3 Grafico a dispersione.

4.1.3 Data Explorer (Esplorazione dei dati)

Il nodo *Data Explorer* offre una vasta gamma di opzioni per visualizzare le proprietà dei dati di input in una vista interattiva. Il nodo supporta lo stile CSS personalizzato ed è contenuto nell'estensione *KNIME JavaScript View (Labs).*

La vista interattiva del nodo è composta da tre schede: una scheda delle statistiche (Numeric tab) una scheda nominale (Nominal tab) e una scheda di anteprima dei dati (Data preview tab). Le prime due schede riassumono le informazioni di tutte le colonne e l'utente può decidere quali includere in un'ulteriore analisi. La scheda delle statistiche mostra le principali proprietà dei dati numerici, come

minimo, massimo, mediana, varianza, somma complessiva e il numero di valori mancanti. Per ogni colonna viene anche calcolato un istogramma che mostra la distribuzione del valore.

Nella scheda nominale viene riportato, per ogni colonna, il numero dei valori nulli, il numero dei valori univoci; inoltre mostra tutti i valori nominali e un grafico delle frequenze.

In Figura 4.4 si riporta la visualizzazione, attraverso il nodo, del dataset *athletic_events.csv.*

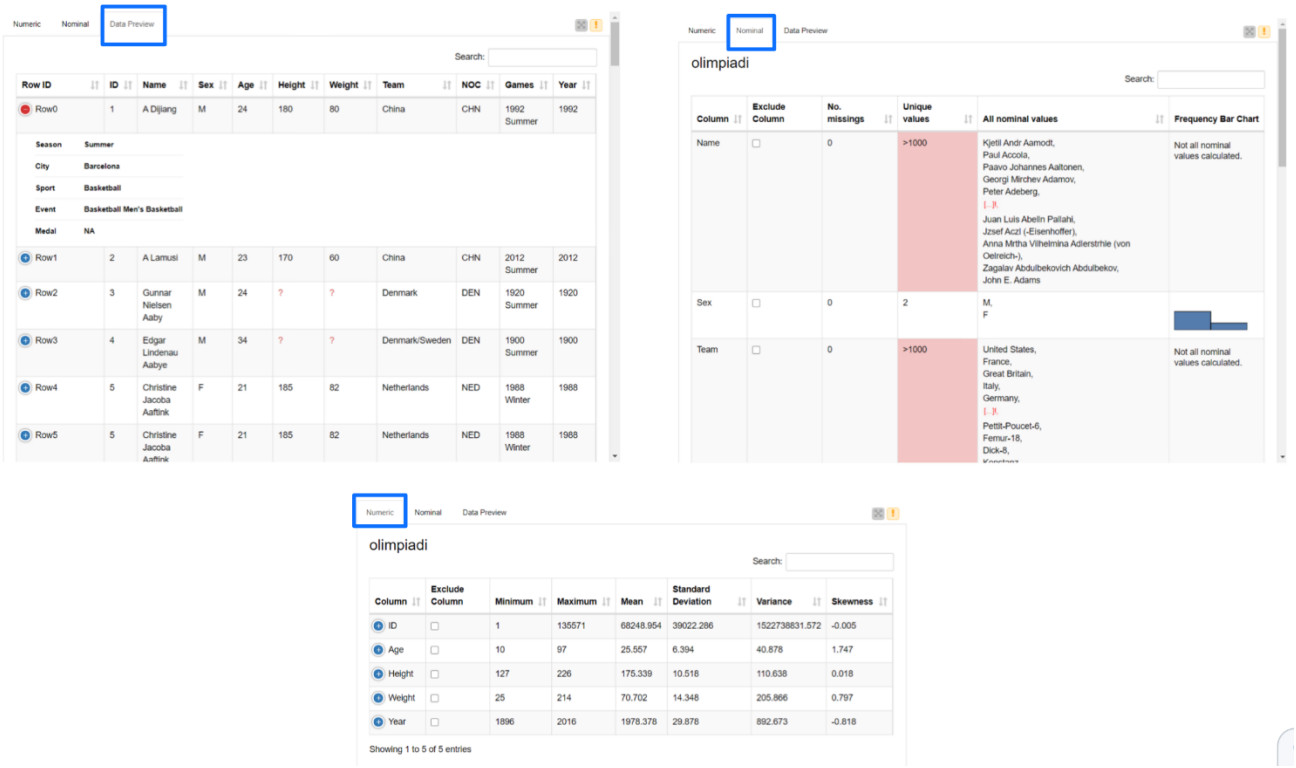


Figura 4.4 Le tre schede del nodo Data Explorer.

4.1.4 Pie/Donut Chart (Grafico a torta/a ciambella)

Un grafico a torta o a ciambella è un tipo di grafico a forma circolare che fornisce la rappresentazione dei dati sotto forma di spicchi la cui dimensione è proporzionale a quella del dato che rappresentano.

Il grafico a ciambella, a differenza del grafico a torta, presenta un'area circolare vuota al centro ed è possibile, in KNIME, scegliere tra i due nelle opzioni della vista del grafico.

In Figura 4.5 si riporta l'utilizzo del nodo *Pie/Donut Chart* per visualizzare i valori presenti per la categoria *Sport* del dataset *athletic_events.csv*.

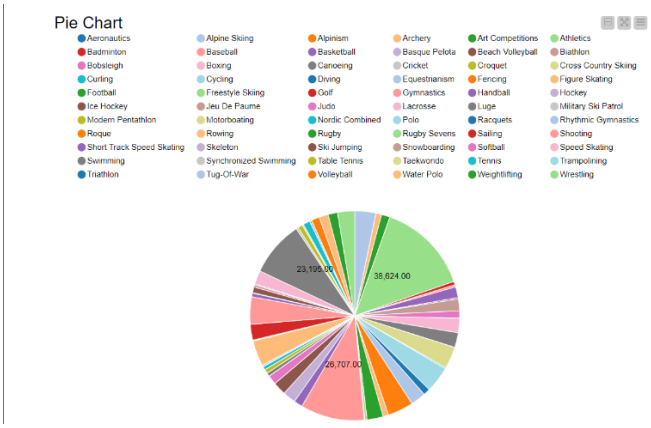


Figura 4.5 Grafico a torta.

4.2 WORKFLOW CONTROL

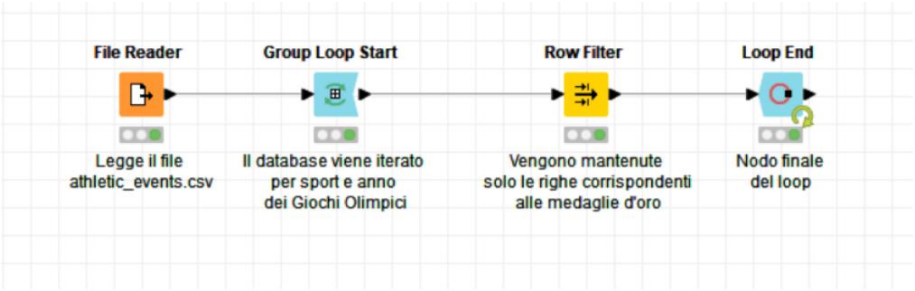
Un flusso di lavoro (*Workflow*) è un modello digitale che consiste nell'automazione totale o parziale di un processo in cui i documenti o le informazioni passano da un'attività ad un'altra per svolgere un determinato lavoro, secondo un insieme di regole definite, con l'obiettivo di ottimizzare le performance e rendere efficiente l'uso di risorse. Ogni attività è collegata ad un input che riceve e trasforma e un output che produce che diventerà l'input per la prossima attività.

Non tutti i flussi di lavoro hanno un input statico e un solo ramo; per questo, KNIME mette a disposizione diversi nodi contenuti all'interno della categoria *Workflow Control* nel repository dei nodi.

4.2.1 Loop

All'interno di un flusso di lavoro possono essere presenti dei loop per ripetere automaticamente parti del workflow; un ciclo in KNIME inizia con un nodo Loop Start e termina con un Loop End e tutte le operazioni si collocano tra questi due nodi. Il nodo Loop Start è responsabile dell'incremento del contatore delle iterazioni e dell'invio dei dati al body del ciclo; dopo che i dati sono stati eseguiti il nodo Loop End verifica se la condizione finale è soddisfatta e, in caso contrario, il nodo Loop Start aumenta il contatore ed esegue una nuova iterazione. Quando la condizione finale è soddisfatta il nodo Loop End raccoglie i dati e viene eseguito il nodo successivo al loop.

In Figura 4.6 si riporta un esempio di loop in cui il dataset *athletic_events.csv* viene iterato per tipo di Sport e per l'anno dei Giochi Olimpici, sfruttando il nodo *Row filter* per conservare solo le righe corrispondenti a una medaglia d'oro. Si riportano, a titolo di esempio, due porzioni della tabella risultante; nella prima sono riportati vari sport per l'anno 1900 mentre nella seconda sono riportati i vari sport per l'anno 1996.



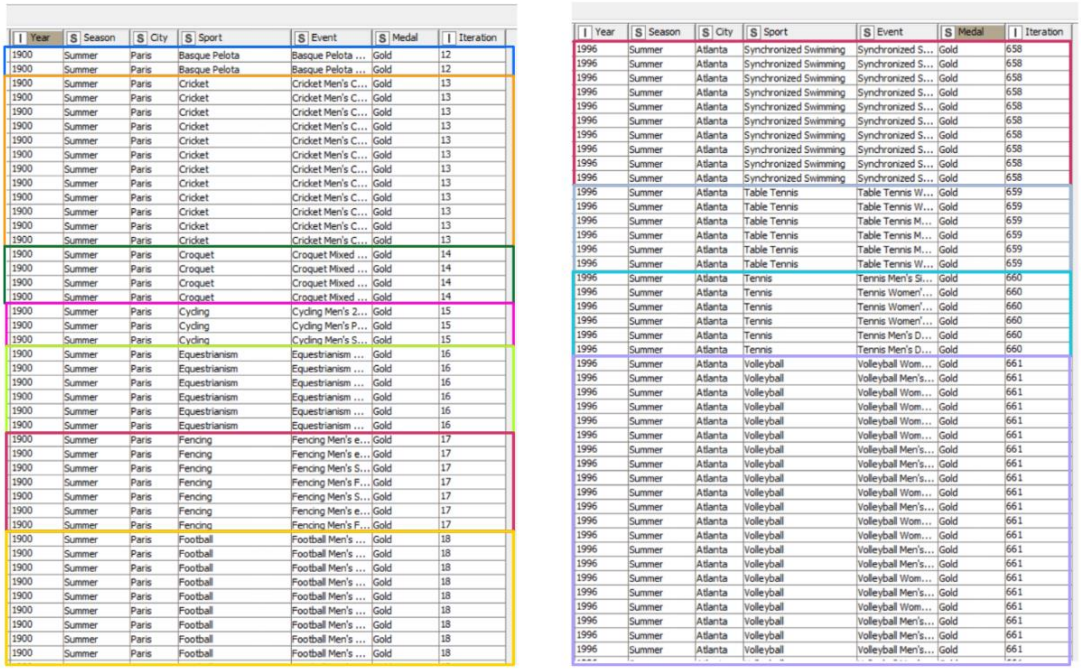


Figura 4.6 Esempio di loop sul datatset athletic_events.csv.

KNIME mette a disposizione diversi nodi, nella categoria *Loop Support*, che consentono di definire diversi tipi di loop. Si introducono di seguito si introducono i più importanti.

4.2.1.1 Counting Loop

Il counting loop accetta in input una tabella e ripete l'esecuzione dei nodi all'interno del costrutto del ciclo per un numero definito di volte, indicato nella configurazione del nodo start.

Questo tipo di loop inizia con il nodo *Counting Loop Start* e può essere terminato da un nodo *Loop End*, che concatena gli output, o da un nodo *Variable Condition Loop End*.

4.2.1.2 Generic Loop

Il generic loop ripete il ciclo fino a quando non viene soddisfatta una condizione in una delle variabili di flusso. Il nodo iniziale non richiede impostazioni specifiche, mentre è necessario impostare la condizione per la variabile di flusso desiderata nella finestra di configurazione del nodo di fine ciclo.

4.2.1.3 Interval Loop

Questo ciclo è simile al counting loop, con la differenza che invece di definire un numero di cicli arbitrario, si inserisce un numero iniziale (“da”) e uno finale (“a”) e un numero come passo, nella configurazione del nodo start.

Il nodo di inizio del ciclo è *Interval Loop Start* e quello di fine è *Loop End.*

4.2.1.4 Chunk Loop

Questo tipo di ciclo consente di scorrere le righe di una tabella a blocchi.

Si può definire nella configurazione del nodo start:

- O il numero di righe per blocco, per cui il numero di iterazioni è definito dal numero di righe totali della tabella diviso per il numero di righe definito dall’utente.
- Oppure il numero di iterazioni; in questo caso, il numero di righe per ogni iterazione viene calcolato prendendo il numero totali delle righe nella tabella di input e dividendolo per il numero di iterazioni deciso dall’utente.

Il ciclo viene iniziato dal nodo *Chunk Loop Start* e può essere terminato da un nodo *Loop End* o un

4.2.1.6 Group Loop

Il ciclo group loop divide le righe della tabella di input in gruppi in base a uno o più attributi.

Il nodo di inizio è *Group Loop Start* e nella sua configurazione bisogna selezionare le colonne da includere per il raggruppamento delle varie iterazioni; il nodo di fine è *Loop End.* Si può vedere un esempio di questo loop alla Figura 4.6.

4.2.2 Switches

Nel caso sia necessario eseguire delle operazioni differenti su diversi gruppi di dato è possibile utilizzare una logica in grado di suddividere il flusso in rami, per questo sono presenti in KNIME i nodi IF e CASE nella sezione *Switches* del repository dei nodi.

In Figura 4.7 viene rappresentato il nodo *IF Switch* che crea due rami nel workflow che possono essere attivati o meno in modo alternato; una volta terminata l'esecuzione, i rami vengono concatenati utilizzando un nodo *End IF.*

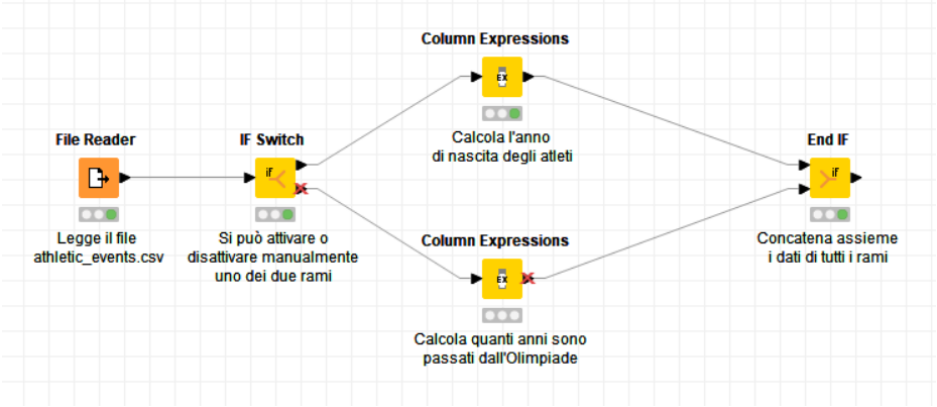


Figura 4.7 Esempio del nodo IF Switch.

Con il nodo *CASE Switch Data* è possibile attivare uno dei tre rami in un flusso di lavoro. In modo simile al nodo IF Switch, questa operazione può essere eseguita manualmente o tramite una condizione.

4.2.3 Try-Catch

Durante l'esecuzione di un flusso di lavoro possono verificarsi diversi tipi di errori, come una connessione non riuscita a un servizio remoto, o un riferimento a un database non accessibile; per questo è presente un meccanismo di gestione degli errori *Try-Catch*. La parte *Try* esegue alcuni nodi e se l'esecuzione fallisce viene attivato il ramo *Catch;* in caso contrario, viene seguito il ramo *Default.*

CONCLUSIONI

Nel presente elaborato è stata svolta un'analisi delle caratteristiche principali di KNIME, concentrando l'attenzione sulle funzionalità di data preparation e di workflow management.

Si può concludere che:

- KNIME implementa metodi molto semplici e intuitivi per supportare i processi ETL (Extract, traform, load), ossia estrazione, trasformazione e caricamento dei dati. Questi metodi sono tutti realizzati dai diversi nodi presenti in KNIME. Nel terzo capitolo, "Data preparation con KNIME", sono state esaminate alcune funzionalità imprescindibili per la data preparation.
- KNIME implementa un'interfaccia user friendly, ossia intuitiva, con menu e funzionalità chiare; le icone dei nodi curate e significative ed essi sono facilmente utilizzabili grazie alla presenza di brevi descrizioni.
- È possibile lavorare in team in quanto i progetti possono essere esportati, commentanti e modularizzati con estrema facilità.
- In KNIME è presente una parte di visualizzazione dei dati, anche se non è uno dei punti forti di KNIME. Nel capitolo quattro vengono comunque analizzati i nodi principali per questa categoria.

In conclusione, KNIME risulta essere un software maturo per la data preparation e il workflow management. Inoltre, le sue funzionalità sono in continua crescita, poiché essendo un applicativo open source, consente e favorisce l'inserimento di funzionalità innovative (o il miglioramento di quelle esistenti) da parte degli sviluppatori della community. Infine, la community, gestisce un forum, utilizzabile dagli utenti come riferimento per la risoluzione di problemi.

BIBLIOGRAFIA

1. KNIME. www.knime.com/

2. Mazhar Hameed, Felix Naumann. “Data Preparation: A Survey of Commercial Tools”. *SIGMOD Record* 49(3): 18-29, 2020. https://sigmodrecord.org/publications/sigmodRecord/2009/pdfs/04_Surveys_Hameed.pdf

3. “End User Data Preparation Market Study - 2018 Edition”, Wisdom of Crowds Series - Licensed to Trifacta, 2018. https://www.trifacta.com/wp-content/uploads/2018/02/End-User-Data-Preparation-Market-Study-2018.pdf

4. “Preparazione dei dati”. www.ibm.com. https://www.ibm.com/it-it/analytics/data-preparation

5. “What is Data Preparation?”. www.talend.com. https://www.talend.com/resources/what-is-data-preparation/

6. Anders Haug, Frederik Zachariassen, Dennis Van Liempd. “The costs of poor data quality”. Journal of Industrial Engineering and Management (JIEM), 4(2): 168–193, 2011.

7. KNIME Hub. https://hub.knime.com/search

8. Sue Newell, Maxine Robertson, Harry Scarbrough, Jacky Swann. “Managing Knowledge Work”. Palgrave-Macmillan, 2002.

9. “120 years of Olympic history: athletes and results”. www.kaggle.com. https://www.kaggle.com/heesoo37/120-years-of-olympic-history-athletes-and-results