UNIVERSITÀ DEGLI STUDI DI TERAMO

Big Data Analytics

Machine Learning in KNIME

Prof.ssa Romina Eramo
Università degli Studi di Teramo
Dipartimento di Scienze della Comunicazione
reramo@unite.it

Esempio

» **Contesto:**
  - Predire i prezzi degli immobili di una città a partire dalle loro caratteristiche
  - Dataset: tabella con caratteristiche degli immobili + prezzo di vendita

» **Obiettivo**
  - Creare un modello in grado di:
    » imparare dai dati storici sugli immobili
    » predire il prezzo di vendita di immobili futuri
    » con un certo grado di accuratezza

Passi fondamentali

» Per costruire un modello di machine learning in KNIME servono quattro nodi:

- **Partitioning**

  Divide il dataset in *training set* e *test set* per evitare overfitting.

- **Learner**

  Applica l'algoritmo di apprendimento sul training set e genera il modello.

- **Predictor**

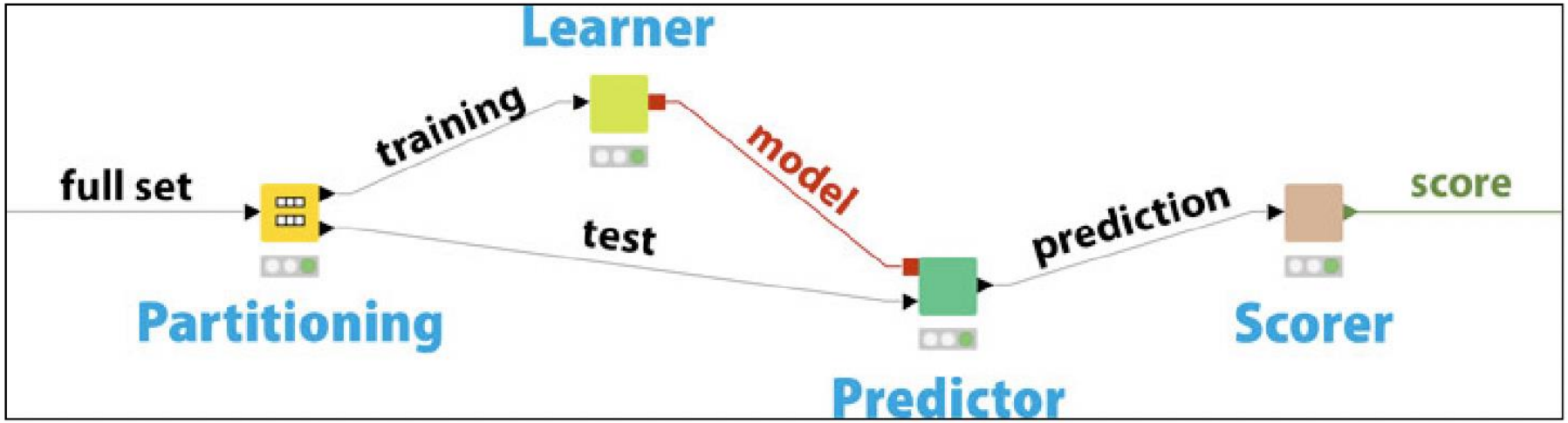
  Usa il modello per produrre predizioni sul test set.

- **Scorer**

  Valuta l'efficacia del modello confrontando valori reali vs. predetti (RMSE, R^2, ecc.).

Tipico workflow

» Tipico workflow per implementare l'apprendimento automatico in KNIME
  - Il *Partitioning* prepara i dati
  - Il *Learner* genera il modello
  - Il *Predictor* lo applica
  - Lo *Scorer* ne misura la qualità



Partitioning

» **Perché è necessario**

- Evita l'overfitting
- Permette di misurare la capacità del modello di generalizzare

» **Cosa fa**

- Divide la tabella di partenza (*full set*) in:
- **Training set** → usato dal learner
- **Test set** → usato per la validazione

Learner

» **Funzione**

- Implementa l'algoritmo di learning scelto (regressione, albero, random forest, ecc.).

» **Output**

- Un **modello appreso**, cioè un insieme di parametri che rappresentano la relazione appresa dai dati.

Predictor

» **Obiettivo**
  - Applicare il modello del learner al test set.

» **Output**
  - Una tabella con:
    » colonne originali del test set
    » **colonna aggiuntiva con le predizioni del prezzo**

Scorer

» **Perché serve**
  - Valuta la qualità del modello in modo oggettivo.

» **Metriche utilizzate**
  - **RMSE** (errore quadratico medio)
  - **R^2** (capacità esplicativa)
  - Confusion matrix (per classificazione)

» **Cosa restituisce**
  - Indicatori numerici di performance.

Tutorial: Predire i prezzi degli immobili (caso Ames)

» **Descrizione del caso: Ames (Iowa)**
- Ames: città USA di ~50.000 abitanti.
- Committente: agenzia immobiliare molto attiva.
- Esigenza: **stimare velocemente il prezzo a cui un immobile sarà venduto**.

» **Perché è importante?**
- Prezzo troppo alto → perdita di tempo e mancata vendita.
- Prezzo troppo basso → perdita di valore e commissioni ridotte.

» **Obiettivo dell'analisi**
- Costruire una **macchina predittiva** che stimi il prezzo di vendita a partire dai dati storici sugli immobili.

Le caratteristiche del dataset

Campi principali:

- **Id** – Identificativo della vendita
- **MSZoning** – Zona urbanistica / destinazione d'uso
- **LotArea** – Dimensione del lotto
- **Neighborhood** – Quartiere
- **OverallQual / OverallCond** – Qualità e condizione generale (1–10)
- **YourRemodAdd** – Anno ristrutturazione
- **RoofStyle** – Tipo di tetto
- **Exterior1st/2nd** – Materiali esterni
- **BedroomAbvGr** – Camere da letto
- **KitchenAbvGr** – Cucine
- **TotRmsAbvGrd** – Numero totale di stanze
- **MoSold / YrSold** – Mese e anno di vendita
- **SalePrice** – Prezzo di vendita

Download dataset

» Il dataset può essere scaricato qui:

- https://hub.knime.com/knime/spaces/Educators%20Alliance/Guide%20to%20Intelligent%20Data%20Science/Exercises/Chapter8_Decision_and_Regression_Trees/data~PuYB8Y5PxbX9sGbl/

Considerazioni

- **Osservazioni sul dataset**
  - I dataset reali contengono molti più campi → necessaria selezione delle variabili.
  - I database sono tipicamente composti da:
  - **tabelle master data** (statiche, descrivono proprietà)
  - **tabelle transazionali** (dinamiche, descrivono eventi)
  - Per stimare i prezzi servono dati puliti, consistenti e combinati.
- **Cosa abbiamo capito finora?**
  - Il problema è un caso classico di **supervised learning**.
  - Target: **SalePrice** (regressione).
  - Obiettivo: ottenere un modello accurato e riutilizzabile.
- **Requisiti operativi**
  - Usare dati storici degli ultimi anni.
  - Configurare un modello di regressione in KNIME.
  - Applicare il workflow: **partitioning → learner → predictor → scorer**.

Strategia di risoluzione

» **Caricare i dati Excel o CSV** in KNIME.

» Assicurarsi che i nomi delle colonne siano corretti (header).

» Pulire i dati da:

- valori mancanti
- valori impossibili
- codifiche errate

» Applicare il modello di machine learning:

- suddividere i dati (training/test)
- addestrare il learner
- generare predizioni col predictor
- valutare con lo scorer

Step 1: Caricamento dei dati in KNIME

» **Come importare il file**

- Trascinare il file Excel/CSV nel Workflow Editor, oppure
- Usare il nodo **Excel Reader (XLS)** o **File Reader (CSV)**

» **Configurazioni importanti**

- Selezionare il foglio corretto
- Attivare *"Column table contains column headers"*
- Verificare tipi di dato (numerico, stringa, data)

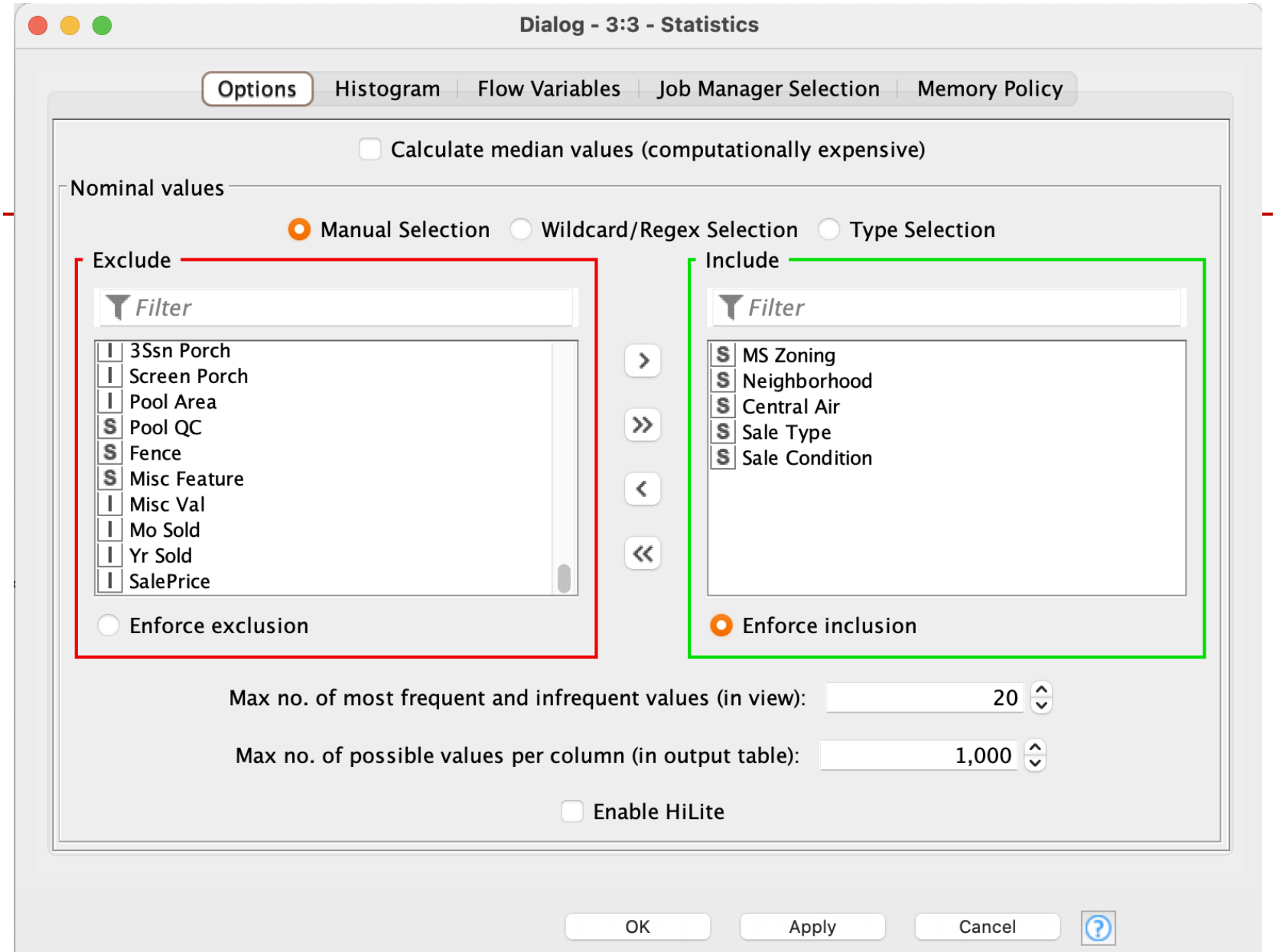
Step 2: Statistics

» Il nodo Statistics genera un **sommario statistico** delle colonne della tabella in input.

» Permette di analizzare:

- valori numerici
- valori nominali/categorici

» Utile per comprendere rapidamente **la struttura dei dati**.


Dialog - 3:3 - Statistics
Options
Histogram
Flow Variables
Job Manager Selection
Memory Policy
Calculate median values (computationally expensive)
Nominal values
Manual Selection
Wildcard/Regex Selection
Type Selection
Exclude
Filter
3Ssn Porch
Screen Porch
Pool Area
Pool QC
Fence
Misc Feature
Misc Val
Mo Sold
Yr Sold
SalePrice
Enforce exclusion
Include
Filter
MS Zoning
Neighborhood
Central Air
Sale Type
Sale Condition
Enforce inclusion
Max no. of most frequent and infrequent values (in view):
20
Max no. of possible values per column (in output table):
1,000
Enable HiLite
OK
Apply
Cancel

Step 2: Statistics

» **Configurazione del nodo (opzioni principali)**
  - Calcolare la mediana (computazionalmente costoso per dataset grandi).
  - Selezionare quali colonne analizzare:
  - Manual selection
  - Wildcard/Regex selection
  - Type selection (per scegliere automaticamente, in base al tipo, le colonne nominali)

» **Risultato** ➔ tre tabelle in output:
  - » *Statistics Table*
  - » *Nominal Histogram Table*
  - » *Occurrences Table*

Output 1: Statistics Table

» Include statistiche numeriche come:
  - Minimo / Massimo
  - Media / Mediana
  - Deviazione standard / Varianza
  - Skewness (asimmetria)
  - Kurtosis (piattezza)
  - Somma totale
  - **Numero di valori mancanti o non numerici** (slide successiva)
  - Istogramma della distribuzione

» Utile per avere **un quadro immediato** delle variabili numeriche.

Statistics Table → No. Missing
Come gestire i valori mancanti

» **Node Repository → Manipulation → Column → Missing Value**

» **Column Settings (trattamento colonna per colonna)**

- **Remove Row** → elimina le righe che contengono missing
- **Fix Value** → sostituisci con un valore definito da te (es. 0, "Unknown", ecc.)
- **Mean/Median** → sostituisci con media o mediana
- **Minimum/Maximum**
- **Most Frequent Value**
- **Previous/Next Value**
- **Linear Interpolation**

Se i missing sono:

- **< 5%** → sostituisci con media/moda
- **5–30%** → valuta sostituzione o rimozione selettiva
- **> 30%** → probabilmente la colonna non è utile per il modello
- **> 50%** → quasi sempre da eliminare

Output 2: Nominal Histogram Table

» **Contenuto**

- Istogramma nominale o diagramma a barre
- Valori unici della colonna, ordinati per frequenza decrescente

» **Quando usarlo**

- Per analizzare colonne categoriche
- Per verificare la distribuzione dei valori nominali
- Per identificare classi molto frequenti o molto rare

Output 3: Occurrences Table

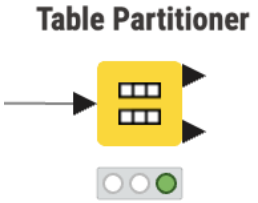
» **Contenuto**

- Frequenza assoluta di ogni valore nominale
- Incidenza relativa (frequenza / totale)
- Ordinamento discendente

» **Utilità**

- Verificare la presenza di categorie dominanti
- Preparare analisi successive o encoding dei dati (One-Hot, Ordinal, ecc.)

Partitioning



» A cosa serve
  - Dividere la tabella iniziale in due sottoinsiemi:
    - »**Training set** → usato dal modello per imparare
    - »**Test set** → usato per valutare l'accuratezza del modello

» **Perché è fondamentale:**
  - Evita **overfitting**
  - Permette di misurare la capacità di **generalizzazione** del modello

Come impostare la dimensione delle partizioni

» Nel nodo **Partitioning** puoi scegliere:

- ✓ **Percentuale (Relative [%])**
  - » (es. 80% training, 20% test)
- ✓ **Numero assoluto (Absolute)**
  - » (es. prime 100 righe → training)

» La seconda tabella conterrà automaticamente tutte le righe rimanenti.

Metodi di campionamento

» **Take from top**
  - Prende le prime righe in ordine
  - *Sconsigliato* per machine learning
» **Linear sampling**
  - Seleziona righe a intervalli regolari
  - Es: riga 1, 3, 5… nella prima partizione
» **Draw randomly**
  - Campionamento casuale
  - *Metodo consigliato per regressione*
» **Stratified sampling**
  - Mantiene la stessa distribuzione di una variabile nominale tra training e test
  - Ideale per classificazione

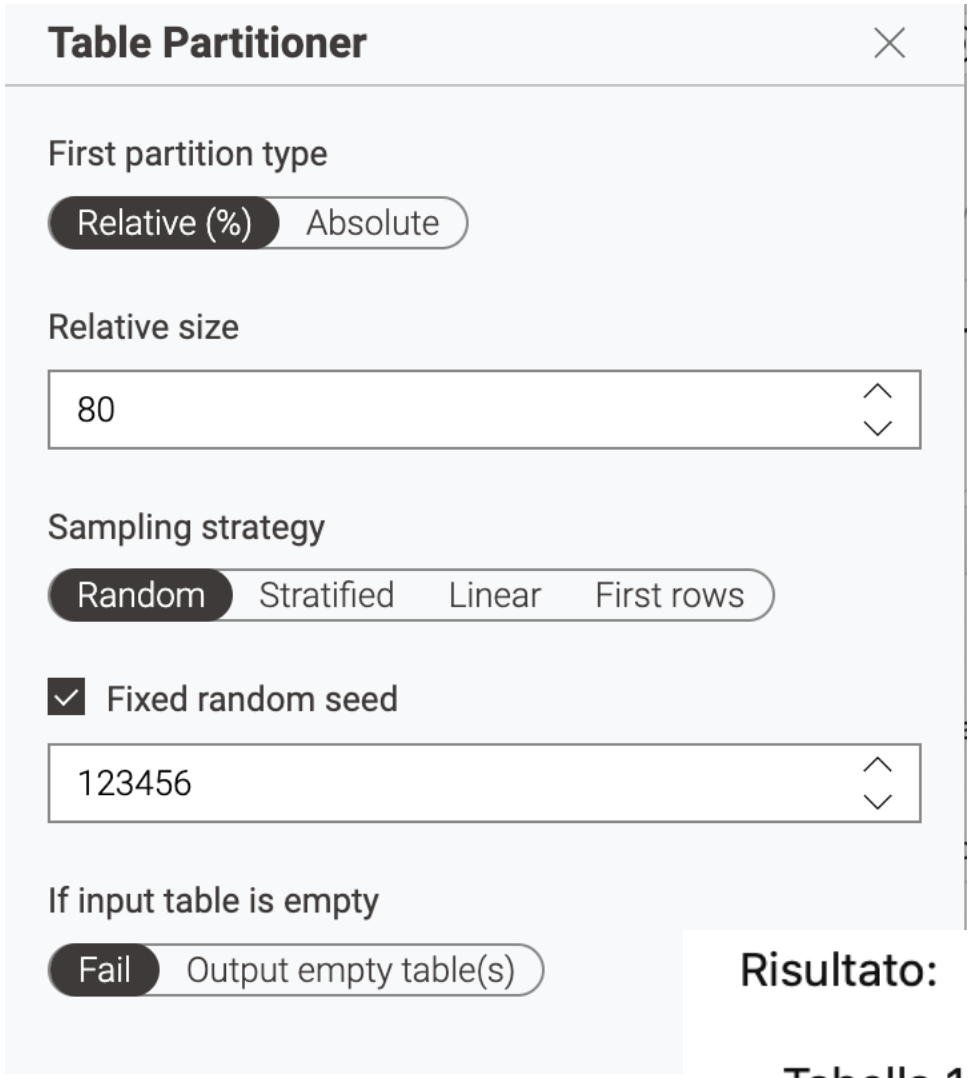
Random seed

» **Perché usarlo?**

- Rende *riproducibili* i risultati
- "Fissa" il comportamento casuale del nodo
- Stesso input + stesso seed → stesso partizionamento

» Per poter mantenere un partizionamento costante impostiamo il **random seed** con un numero a nostro piacimento:

```
Random Seed = 123456
```



Risultato:

- Tabella 1 → 80% degli immobili (training)
- Tabella 2 → 20% degli immobili (test)

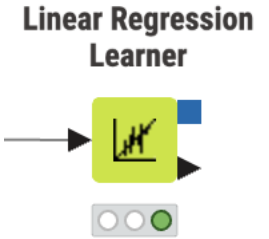
Valutazione del modello

» **Ci concentriamo su: Supervised learning**

- **Regressione** → target numerico
- **Classificazione** → target categoriale

→ la colonna *SalePrice* è il risultato noto che la macchina deve imparare a predire.

Linear Regression Learner



» **A cosa serve?**

- Addestrare un modello che predice una **variabile numerica** attraverso la **regressione lineare multipla**.

» **Input richiesto**

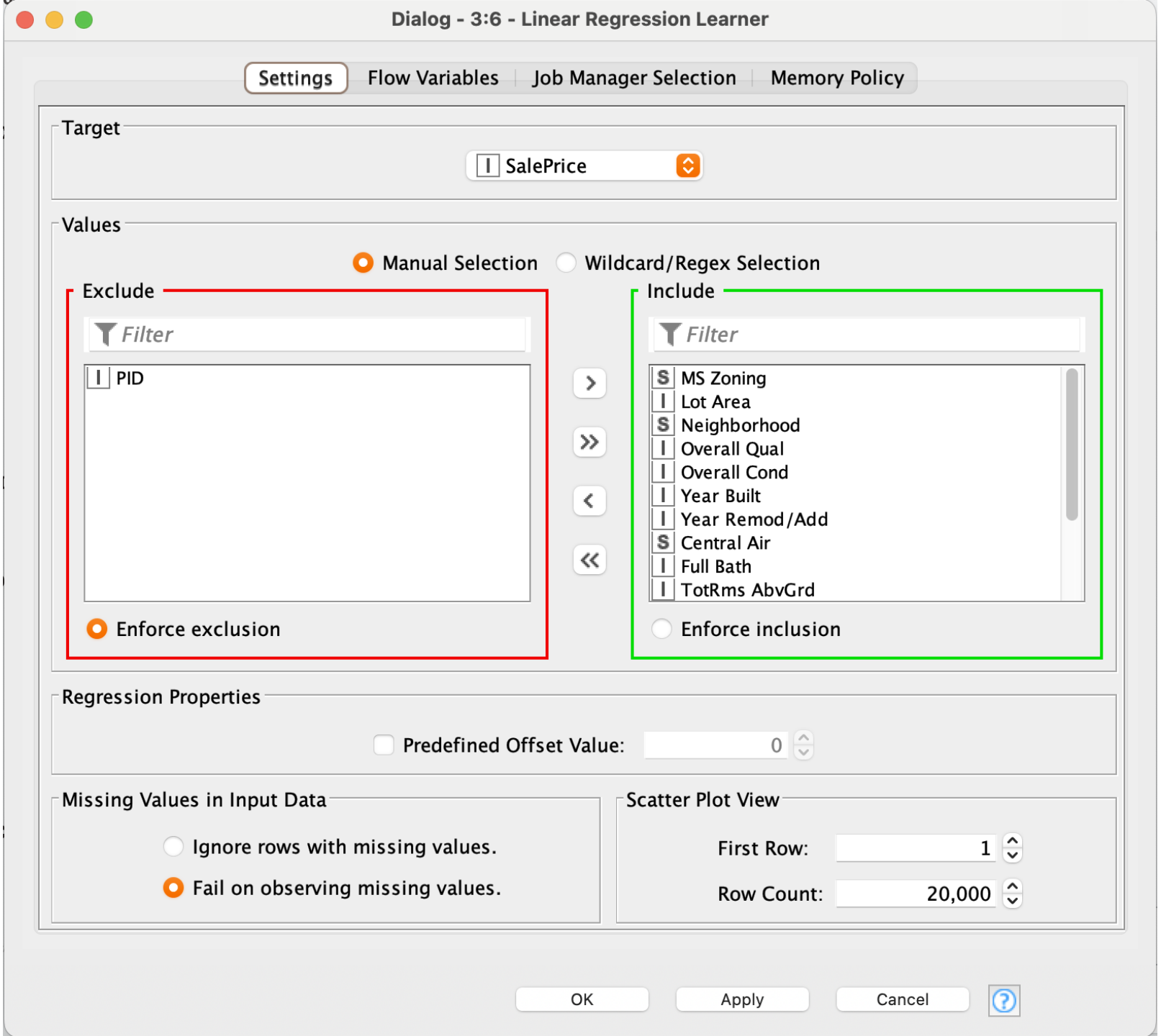
- Un dataset **numerico** (training set)
- La colonna **target** da prevedere (es. *SalePrice*)

Configurazione iniziale

1. **Selezionare la colonna target**
   - Deve essere numerica
   - Esempio: **SalePrice**

2. **Selezionare i predittori**
   - Colonne numeriche o categoriche
   - Le variabili categoriche verranno **convertite automaticamente in dummy**

3. **Escludere colonne irrilevanti**
   - Da escludere: **Id** (non ha relazione con il prezzo)


Dialog - 3:6 - Linear Regression Learner
Settings
Flow Variables
Job Manager Selection
Memory Policy
Target
SalePrice
Values
Manual Selection
Wildcard/Regex Selection
Exclude
Filter
PID
Enforce exclusion
Include
Filter
MS Zoning
Lot Area
Neighborhood
Overall Qual
Overall Cond
Year Built
Year Remod/Add
Central Air
Full Bath
TotRms AbvGrd
Enforce inclusion
Regression Properties
Predefined Offset Value:
0
Missing Values in Input Data
Ignore rows with missing values.
Fail on observing missing values.
Scatter Plot View
First Row:
1
Row Count:
20,000
OK
Apply
Cancel

Definizione di intercetta (bias)

» L'**intercetta** (o **bias**) è il valore dell'output previsto quando **tutti i predittori sono pari a zero**.

👉 È il **punto di partenza** del modello.

» **Perché è importante?**

- Fornisce un **livello base** della predizione
- Permette al modello di **non passare per lo zero**
- Migliora la **flessibilità** e la capacità di adattarsi ai dati
- Presente in regressione lineare, logistica e reti neurali

$$y = b_0 + b_1 x_1 + b_2 x_2 + \ldots$$

b_0 = intercetta

» 👉 *Il bias regola l'altezza del modello e consente predizioni realistiche anche quando le variabili esplicative valgono zero.*

Configurazione dell'Intercetta (Offset Value)

» **Predefined Offset Value**
  - Possibilità di **fissare l'intercetta a 0**
  - Vantaggi:
    » modello più semplice e interpretabile
    » ogni coefficiente mostra **di quanto il predittore contribuisce** al prezzo

» **Quando usarlo?**
  - Quando la spiegabilità è più importante della precisione marginale

Output del nodo

» Il nodo produce:

1. Modello appreso
   - Da inviare al nodo *Predictor*
2. Tabella dei Coefficienti e Statistiche
   - Coefficienti $b_0 \ldots b_N$
   - Satistiche di significatività
     - »t-value
       - **quanto il coefficiente stimato sia diverso da zero**, in rapporto alla sua variabilità (errore standard)
     - »p-value (**P>|t|**)
       - **probabilità che il coefficiente osservato sia dovuto al caso**
   - Variabili dummy incluse automaticamente

Come interpretare p-value e significatività

» **Regola generale**
- **p-value < 0.05** → predittore **significativo**
- **p-value > 0.05** → predittore poco utile → valutare rimozione

» **Significato**
- Misura quanto un predittore spiega la variabilità della target
- Utile per selezionare i predittori più rilevanti

Analisi della tabella Coefficients & Statistics

» **Ordinare i predittori**
- Clic sulla colonna **P>|t|**, poi → *Sort Descending*
- Vengono mostrati dal più significativo al meno significativo

» **Predittori molto significativi (dataset Ames)**
- Numero dei vani (**TotRmsAbvGrd**)
- Qualità generale (**OverallQual**)
- Dimensione lotto (**LotArea**)
- Anno di costruzione (**YearBuilt**)
- Quartiere **NoRidge**

Attenzione alla collinearità

» **Cosa succede?**
  - Due predittori sono fortemente correlati

  → uno dei due diventa statisticamente “inutile”

» **Esempio:**
  - *GarageCars* correlato con *GarageArea*
  - *GarageArea* più significativo → *GarageCars* perde significatività

» **Come risolvere?**
  - Rimuovere una delle due colonne prima del training
  - Evitare predittori duplicati o quasi-equivalenti

Come interpretare i coefficienti

» **I coefficienti (Coeff.) indicano:**

- La variazione del prezzo per ogni unità del predittore

» **Esempi (Ames):**

- 1 punto in più di **OverallQual** → +19.940 $
- 1 vano in più **TotRmsAbvGrd** → +7.659 $
- Quartiere **NoRidge** → +90.308 $
- Quartiere **Veenker** → +35.629 $

Spiegabilità del modello (**Explainability**)

» Il modello lineare è **completamente interpretabile**.

» **Per stimare il prezzo di un immobile:**

1. Moltiplica ogni variabile per il suo coefficiente
2. Somma i contributi
3. Aggiungi l'effetto del quartiere (variabili dummy)
4. Ottieni la stima finale

» Esempio concettuale:

- Prezzo = 19.940 × OverallQual
- 7.659 × TotRmsAbvGrd
- valore dummy del quartiere
- … (altri predittori)

Regression Predictor

» Applica il **modello di regressione** (lineare multipla o polinomiale) ai dati del **test set**.

» Produce una **nuova colonna** contenente le predizioni (Prediction(SalePrice)).

» **Input richiesti**

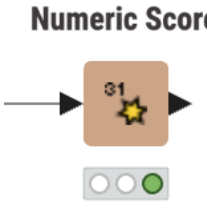
- **Modello** dal Linear Regression Learner
- **Test set** dal nodo Partitioning

» **Operazione**

- Somma dei prodotti: **coefficiente × valore del predittore**

→ predizione del prezzo per ogni immobile.

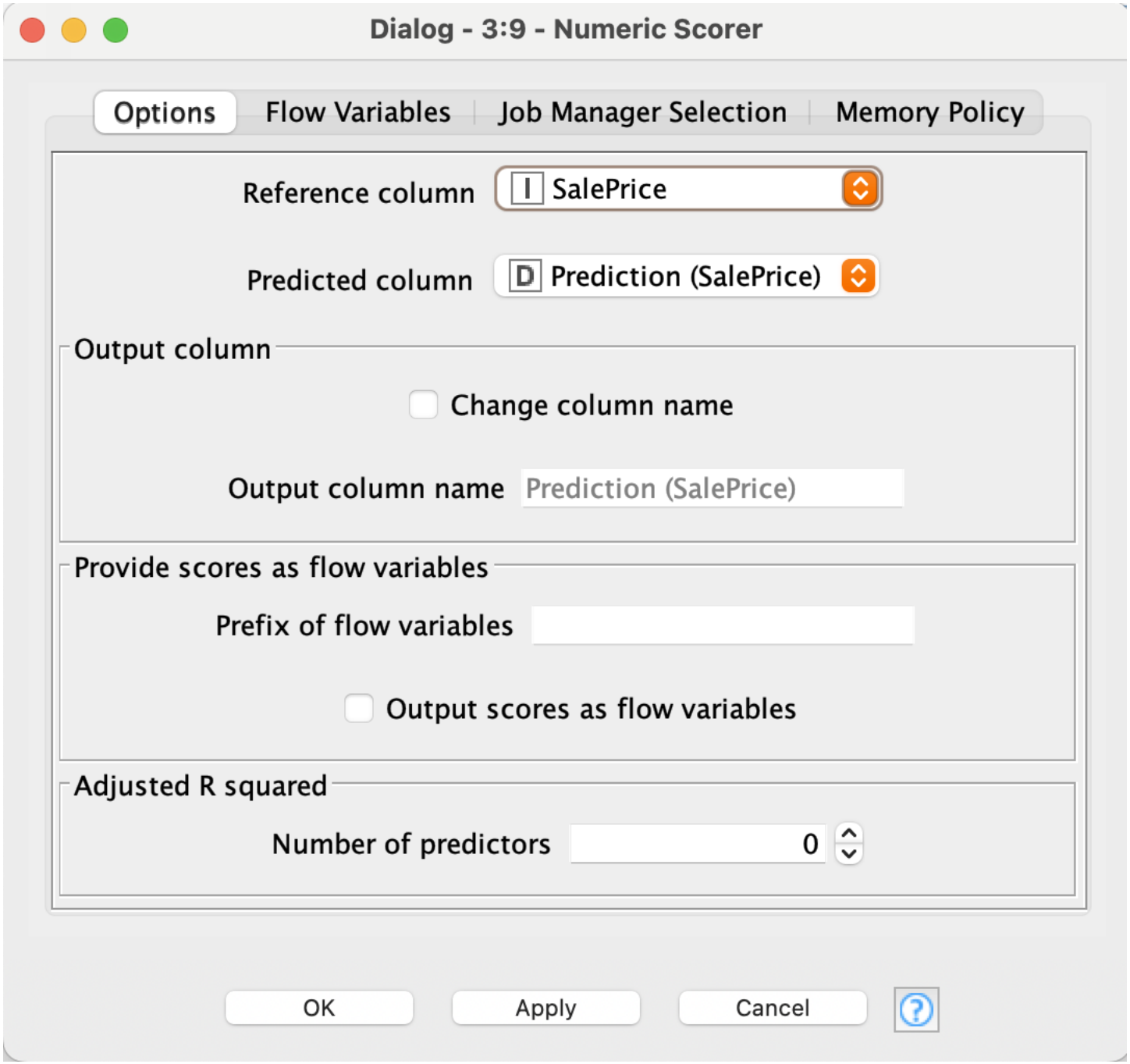
Numeric Scorer



» Valuta **quanto bene** il modello ha imparato a predire il target.

» **Funzioni principali**

- Confronta:
  - » **Reference column** → valori reali
  - » **Predicted column** → valori stimati
- Calcola metriche di performance (R^2, MAE, MSE, RMSE, MSD).


Dialog - 3:9 - Numeric Scorer
Options
Flow Variables
Job Manager Selection
Memory Policy
Reference column
SalePrice
Predicted column
Prediction (SalePrice)
Output column
Change column name
Output column name
Prediction (SalePrice)
Provide scores as flow variables
Prefix of flow variables
Output scores as flow variables
Adjusted R squared
Number of predictors
0
OK
Apply
Cancel

Metriche prodotte dal Numeric Scorer

» **R^2 — Coefficiente di determinazione**
  - Percentuale di variabilità spiegata dal modello.

» **MAE — Mean Absolute Error**
  - Media degli errori assoluti.

» **MSE — Mean Squared Error**
  - Media degli errori quadratici.

» **RMSE — Root Mean Squared Error**
  - Errore medio in unità reali (es. $).

» **MSD — Mean Signed Difference**
  - Media degli scarti (positivi e negativi).

Risultati del modello (dopo lo Scorer)

» **Performance ottenute**
- **$R^2 = 0.844$** → il modello spiega ~84% della variabilità dei prezzi
- **RMSE ≈ 30.000 $** → errore medio previsto nei 2/3 dei casi

» **Interpretazione business**
- Le stime rientrano nel margine di negoziazione tipico del settore immobiliare.
- Il modello è **accurato**, **affidabile** e utilizzabile dagli agenti.

Necessità di semplificare il modello

» Perché semplificare?
  - Troppi predittori → difficili da spiegare
  - Alcuni hanno **p-value alto** → scarsa utilità

» **Variabili eliminate (non significative)**
  - GarageCars
  - SaleType
  - SaleCondition
  - MSZoning
  - CentralAir = Y

» **Risultato dopo la pulizia**
  - **$R^2 = 0.836$** → Accuratezza quasi invariata, modello più semplice.

Introduzione della variabile “Età dell’immobile”

» **Problema**

- Usare direttamente YearBuilt e YearRemodAdd produce formule poco intuitive.

» **Soluzione (Math Formula)**

- **AgeFromBuilt = 2010 - YearBuilt**
- **AgeFromRemod = 2010 - YearRemodAdd**

» **Risultato**

- Modello più **interpretabile**
- R^2 migliora leggermente → **0.845**

Ulteriore semplificazione del modello

» **Analisi dei nuovi parametri**
  - Ogni anno di età → –343 $
  - Ogni anno dall'ultima ristrutturazione → –304 $

» **Eliminata un'altra variabile non significativa**
  - **FullBath** (p-value = 0.136)

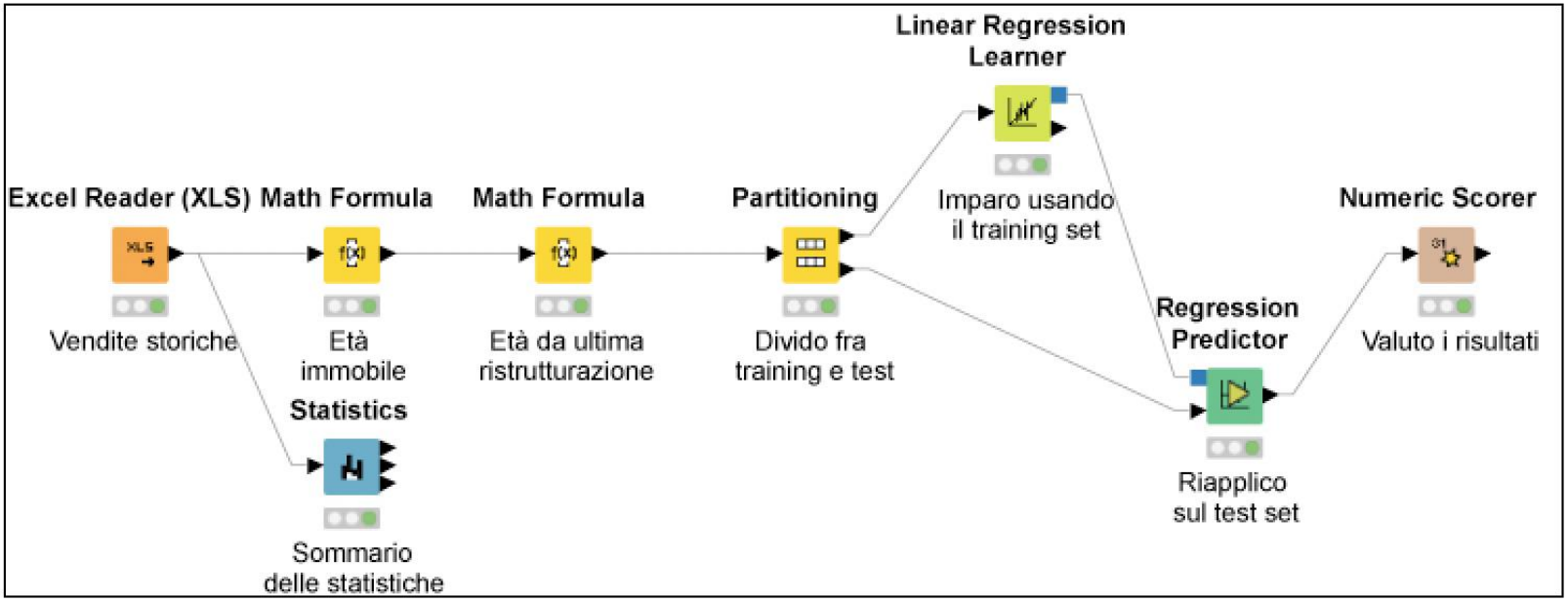
» **R^2 finale**
  - **$R^2 = 0.846$** → modello *ancora più semplice* e *ancora accurato*.

Workflow finale del modello

» **Componenti principali:**

- Excel Reader (input storico)
- Math Formula (età, anni da ristrutturazione)
- Partitioning (train/test)
- Linear Regression Learner
- Regression Predictor
- Numeric Scorer

» Il modello ora predice i prezzi con buona accuratezza ed è completamente spiegabile.


Linear Regression Learner
Excel Reader (XLS)
Math Formula
Math Formula
Partitioning
Imparo usando il training set
Numeric Scorer
Vendite storiche
Età immobile
Età da ultima ristrutturazione
Divido fra training e test
Regression Predictor
Valuto i risultati
Statistics
Riapplico sul test set
Sommario delle statistiche

Come usare il modello nella pratica?

» Proposte al committente:
  - Condividere workflow KNIME con gli agenti
  - Creare una pagina web con form e stima automatica
  - Creare file Excel con formule incorporate
  - Condividere una tabella dei coefficienti per calcolo manuale

→ Strumenti per autonomia totale degli agenti.