Machine Learning for Business Analytics

Prem Kumar Chimakurthi

Master of Science in Business Analytics

Table of Contents:

Week 4 | Predicting Boston Housing Prices

Why should the data be partitioned into training, validation, and holdout sets?

The data needs to be divided so that the model is trained with one part of data and validated and tested with other parts. This helps in Training: This set of data is used in training the model to make it learn about the data distribution.

Validation set: This set of data is applied to tune the model's hyperparameters and to prevent overfitting by checking the performance of the model on data that it has not been trained with.

Holdout set (test set): This set is used to get an unbiased measure of the generalization of the performance of the final model after training and validation. It tells how well the model will perform on data that it has not seen before.

What will the training set be used for?

The machine learning model is trained with the training set. The model learns relationships between the predictors (features) and the target variable (MEDV) where it minimizes error on this data set. Most of the data is in the training set so it is essential for the model to learn the patterns.

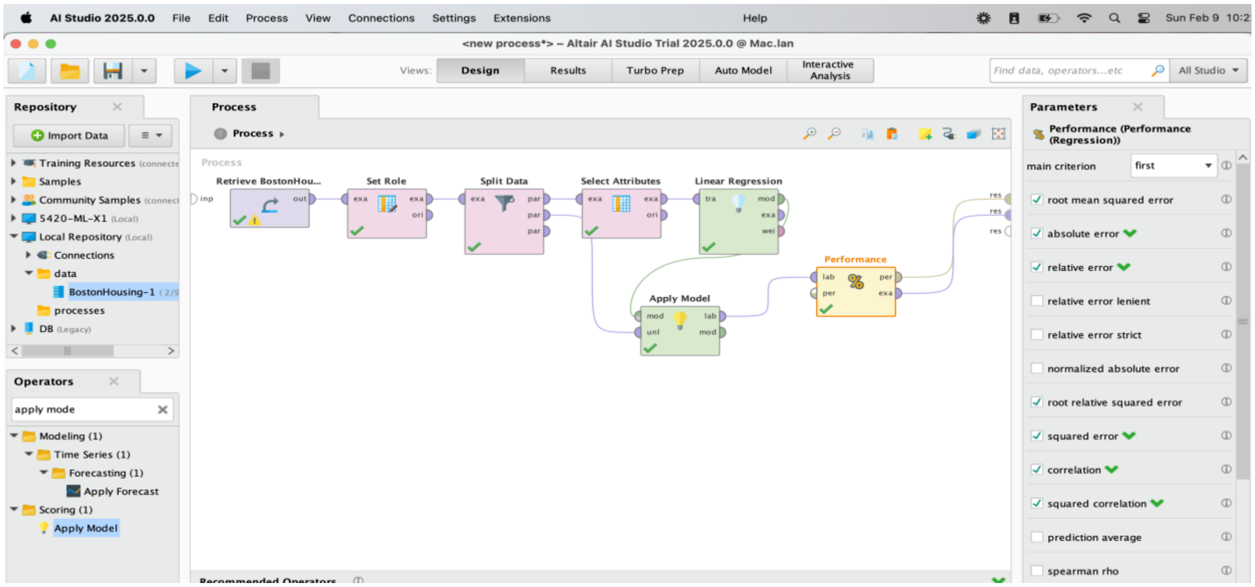
What will the validation and holdout sets be used for?

Validation set: The validation set is employed for tuning the models' hyperparameters and making evaluations during the training process. It assists in choosing the optimal model complexity and overfitting prevention through reporting the model's performance on data that is not used for training.

Holdout set (test set): The holdout set is deployed to evaluate the final model's performance after training and validation. It gives an unbiased idea of how well the model would perform on data that it has never seen before. The set is tossed once after the model is thoroughly trained and validated so that the evaluation is completely unbiased and not contaminated by the training or validation set.

Partition the data into training/validation/holdout with proportions 60:25:15. Fit a multiple linear regression model to the median house price (MEDV) as a function of CRIM, CHAS, and RM.

Partitioning the data:

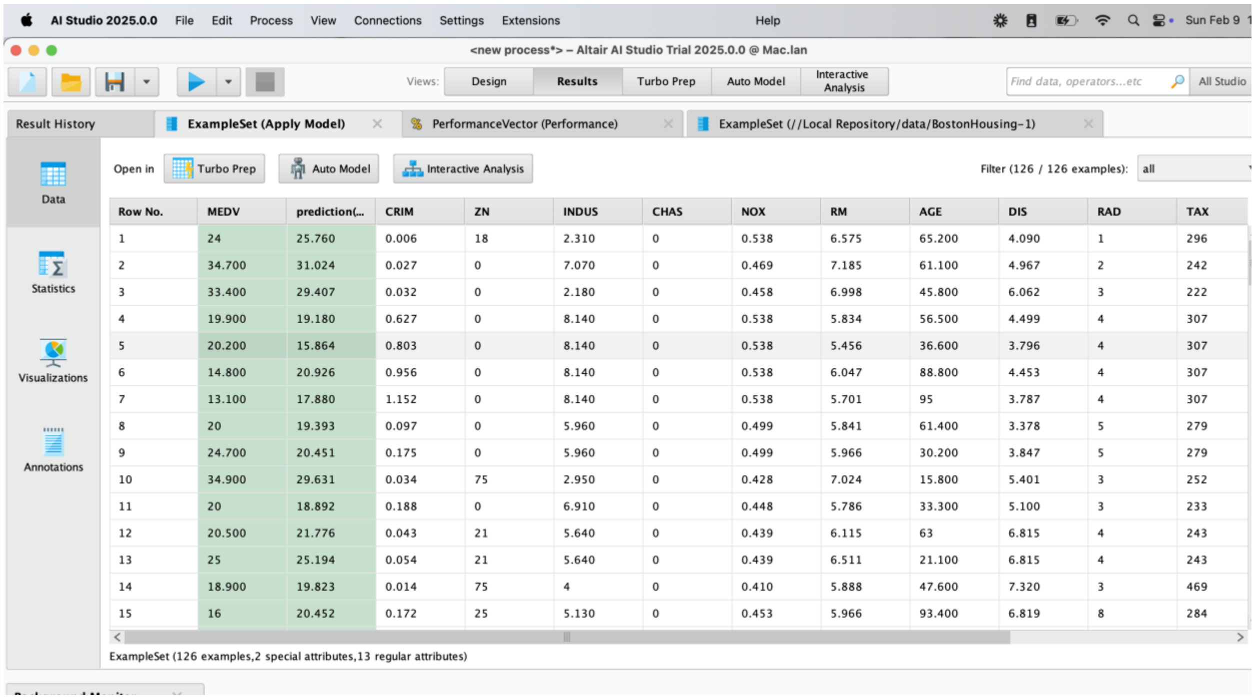


Parameters

Performance (Performance (Regression))

main criterion: first

- ☑ root mean squared error
- ☑ absolute error
- ☑ relative error
- □ relative error lenient
- □ relative error strict
- □ normalized absolute error
- ☑ root relative squared error
- ☑ squared error
- ☑ correlation
- ☑ squared correlation
- □ prediction average
- □ spearman rho



Row No.	MEDV	prediction(...	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX
1	24	25.760	0.006	18	2.310	0	0.538	6.575	65.200	4.090	1	296
2	34.700	31.024	0.027	0	7.070	0	0.469	7.185	61.100	4.967	2	242
3	33.400	29.407	0.032	0	2.180	0	0.458	6.998	45.800	6.062	3	222
4	19.900	19.180	0.627	0	8.140	0	0.538	5.834	56.500	4.499	4	307
5	20.200	15.864	0.803	0	8.140	0	0.538	5.456	36.600	3.796	4	307
6	14.800	20.926	0.956	0	8.140	0	0.538	6.047	88.800	4.453	4	307
7	13.100	17.880	1.152	0	8.140	0	0.538	5.701	95	3.787	4	307
8	20	19.393	0.097	0	5.960	0	0.499	5.841	61.400	3.378	5	279
9	24.700	20.451	0.175	0	5.960	0	0.499	5.966	30.200	3.847	5	279
10	34.900	29.631	0.034	75	2.950	0	0.428	7.024	15.800	5.401	3	252
11	20	18.892	0.188	0	6.910	0	0.448	5.786	33.300	5.100	3	233
12	20.500	21.776	0.043	21	5.640	0	0.439	6.115	63	6.815	4	243
13	25	25.194	0.054	21	5.640	0	0.439	6.511	21.100	6.815	4	243
14	18.900	19.823	0.014	75	4	0	0.410	5.888	47.600	7.320	3	469
15	16	20.452	0.172	25	5.130	0	0.453	5.966	93.400	6.819	8	284

ExampleSet (126 examples,2 special attributes,13 regular attributes)

Linear Regression Model performance is summarized below:

- Root Mean Squared Error (RMSE), 6.299, Average prediction error is ~$6,299 (since MEDV is in $1000s). Lower is better.
- Absolute Error (MAE), 4.333, On average, predictions deviate by ~$4,333 from actual prices.
- Relative Error (%), 26.07%, Model's errors are ~26% of actual values, which is decent but could be improved.
- Root Relative Squared Error (RRSE), 0.705, Below 1, which means the model is better than simply predicting the mean.
- Correlation (R), 0.735, Moderate correlation between actual and predicted values.

- R^2 (Squared Correlation), 0.540, The model explains 54% of the variance in house prices (MEDV).

Key Observations:

- $R^2 = 0.54$: Your model explains 54% of house price variation, meaning there’s room for improvement.
- RMSE = 6.299: On average, the model's predictions are off by ~$6,300.
- MAE = 4.333: On average, the absolute error is ~$4,300, which is reasonable.
- Relative Error = 26.07%: The error is about one-fourth of the true house prices.

Write the equation for predicting the median house price from the predictors in the model.

Formula for the equation:

$$\mathbf{MEDV= \beta 0 +\beta 1\ (CRIM)+\beta 2\ (CHAS)+\beta 3\ (RM)+\epsilon}$$

Where:

$\beta 0$ = Intercept (constant term)

$\beta 1$, $\beta 2$,$\beta 3$ = Coefficients for each predictor variable

ϵ = Error term (represents unexplained variance)

$MEDV=-31.037-0.288(CRIM)+6.597(CHAS)+8.639(RM)$

Where:

- Intercept = -31.037 (baseline house price when all predictors are zero)
- CRIM coefficient = -0.288 (higher crime rate slightly decreases house price)
- CHAS coefficient = 6.597 (houses near the Charles River increase in price)
- RM coefficient = 8.639 (each additional room increases house price significantly)

Using the estimated regression model, what median house price is predicted for a tract in the Boston area that does not bound the Charles River, has a crime rate of 0.1, and where the average number of rooms per house is 6?

To predict the median house price (MEDV) using the estimated regression model, we use the given equation:

$$\mathbf{MEDV=-31.037-0.288(CRIM)+6.597(CHAS)+8.639(RM)}$$

Given Data for Prediction:

- **CRIM = 0.1 (Crime rate per capita)**
- **CHAS = 0 (Does not bound the Charles River)**
- **RM = 6 (Average number of rooms)**

$$MEDV=-31.037-0.288(0.1)+6.597(0)+8.639(6)$$

$$MEDV=-31.037-0.0288+0+51.834$$

$$MEDV=20.768$$

The predicted median house price is $20,768 (or $20.77K in $1000s)

Which predictors are likely to be measuring the same thing among the 13 predictors? Discuss the relationships among INDUS, NOX, and TAX.

Identifying Predictors That Measure Similar Factors:

Among the 13 predictors in the Boston Housing dataset, some variables are measuring similar underlying factors due to high correlation. Typically, in real estate and urban planning, some features are naturally related.

- **INDUS (Industrial Proportion): Measures the proportion of land zoned for non-retail business (industrial use).**
- **NOX (Nitric Oxide Concentration): Indicates air pollution levels.**
- **TAX (Property Tax Rate per $10,000): Represents property tax per town.**

These three variables are likely to be correlated because they are all related to urbanization and industrial activity.

Relationship Between INDUS, NOX, and TAX:

These three features are linked because industrialization often influences pollution levels and property taxation. Let us analyze their relationships:

INDUS and NOX (Industrialization & Pollution):

- **More industrial land usage (higher INDUS) typically leads to higher NOX levels due to increased emissions from factories and businesses.**
- **In urban areas with high industrialization, air quality tends to worsen.**

INDUS and TAX (Industrialization & Property Tax):

- **Higher industrial land usage (higher INDUS) is often associated with higher property tax (TAX).**
- **Towns with more industrial zones might impose higher taxes to support infrastructure, services, and pollution control.**

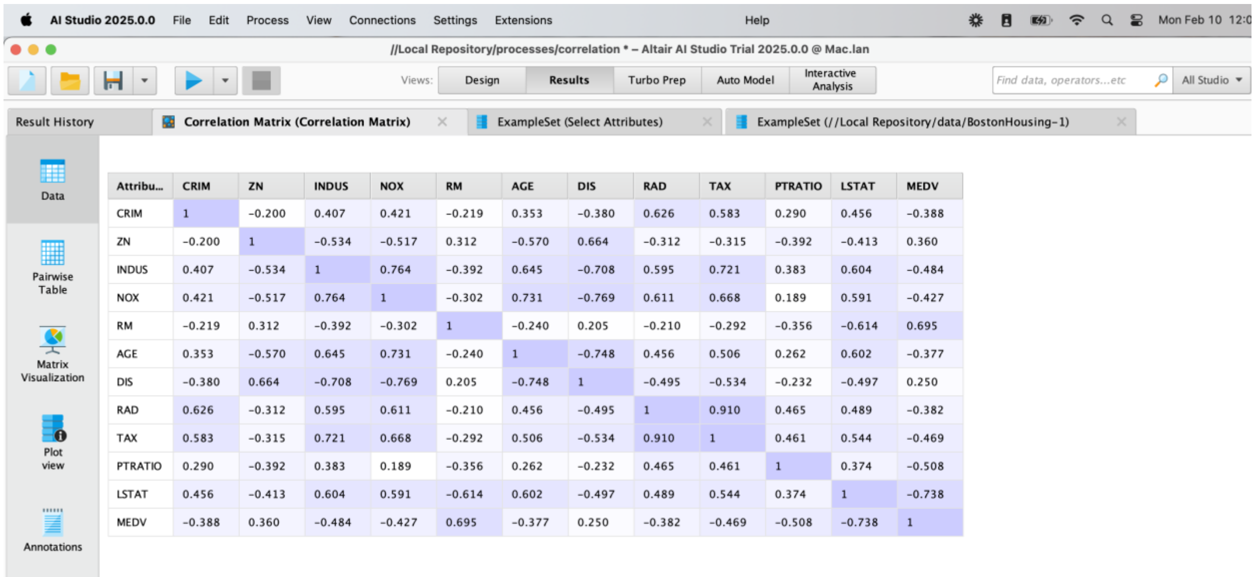
NOX and TAX (Pollution & Property Tax):

- **Areas with high pollution (higher NOX) tend to have higher taxes because governments might impose environmental regulations or higher property tax rates to offset environmental damages.**

- **Additionally, wealthier areas with lower pollution may have lower tax rates due to better zoning and urban planning.**

Compute the correlation table for the 12 numerical predictors using the Correlation Matrix operator in RapidMiner, and search for highly correlated pairs. These have potential redundancy and can cause multicollinearity. Choose which ones to remove based on this table.

<u>The correlation table for the 12 numerical predictors:</u>


AI Studio 2025.0.0
File
Edit
Process
View
Connections
Settings
Extensions
Help
Mon Feb 10 12:0
//Local Repository/processes/correlation * – Altair AI Studio Trial 2025.0.0 @ Mac.lan
Views:
Design
Results
Turbo Prep
Auto Model
Interactive Analysis
Find data, operators...etc
All Studio
Result History
Correlation Matrix (Correlation Matrix)
ExampleSet (Select Attributes)
ExampleSet (//Local Repository/data/BostonHousing-1)
Data
Pairwise Table
Matrix Visualization
Plot view
Annotations

Attribu...	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	LSTAT	MEDV
CRIM	1	-0.200	0.407	0.421	-0.219	0.353	-0.380	0.626	0.583	0.290	0.456	-0.388
ZN	-0.200	1	-0.534	-0.517	0.312	-0.570	0.664	-0.312	-0.315	-0.392	-0.413	0.360
INDUS	0.407	-0.534	1	0.764	-0.392	0.645	-0.708	0.595	0.721	0.383	0.604	-0.484
NOX	0.421	-0.517	0.764	1	-0.302	0.731	-0.769	0.611	0.668	0.189	0.591	-0.427
RM	-0.219	0.312	-0.392	-0.302	1	-0.240	0.205	-0.210	-0.292	-0.356	-0.614	0.695
AGE	0.353	-0.570	0.645	0.731	-0.240	1	-0.748	0.456	0.506	0.262	0.602	-0.377
DIS	-0.380	0.664	-0.708	-0.769	0.205	-0.748	1	-0.495	-0.534	-0.232	-0.497	0.250
RAD	0.626	-0.312	0.595	0.611	-0.210	0.456	-0.495	1	0.910	0.465	0.489	-0.382
TAX	0.583	-0.315	0.721	0.668	-0.292	0.506	-0.534	0.910	1	0.461	0.544	-0.469
PTRATIO	0.290	-0.392	0.383	0.189	-0.356	0.262	-0.232	0.465	0.461	1	0.374	-0.508
LSTAT	0.456	-0.413	0.604	0.591	-0.614	0.602	-0.497	0.489	0.544	0.374	1	-0.738
MEDV	-0.388	0.360	-0.484	-0.427	0.695	-0.377	0.250	-0.382	-0.469	-0.508	-0.738	1

<u>Highly Correlated Pairs (Potential Multicollinearity Issues):</u>

From the table, the strongest correlations (above ±0.75) are:

- INDUS vs. NOX → 0.764
- High industrialization leads to high pollution (NOX).

DIS vs. INDUS → -0.708 (Close to the threshold)

Distance to employment centers is negatively correlated with industrialization.

DIS vs. NOX → -0.769

More distance to employment centers → Lower pollution.

AGE vs. NOX → 0.731 (Close to threshold)

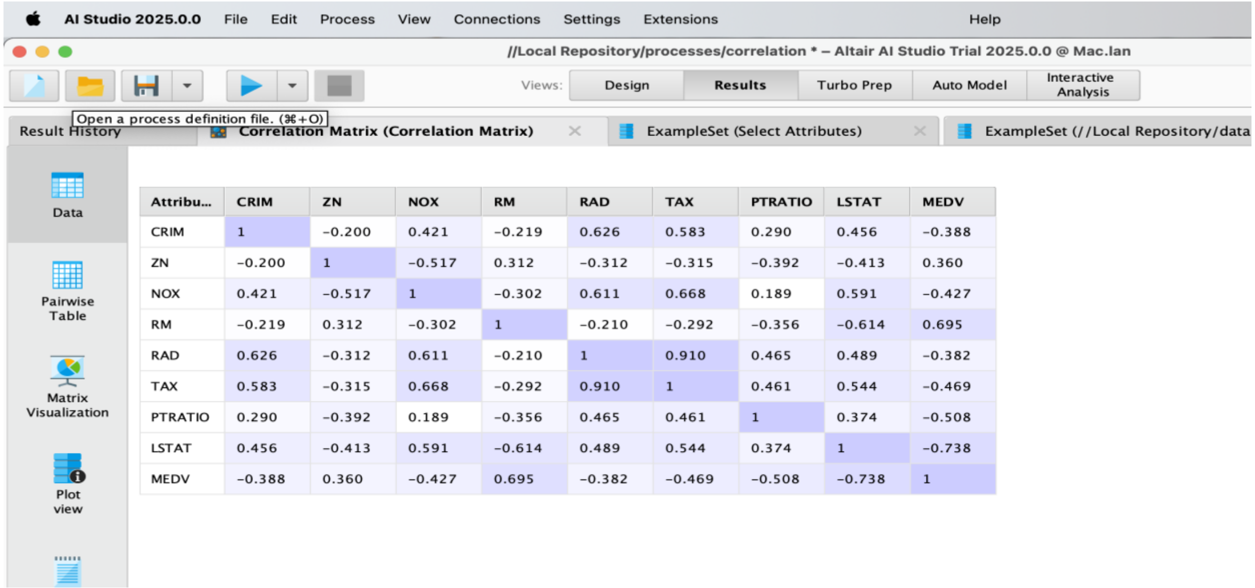
Older homes tend to be in polluted areas.

LSTAT vs. MEDV → -0.738

- A higher low-income percentage reduces home value.
- This is expected and should be kept, as it explains home price variation.

To avoid redundancy and improve model efficiency, remove:

- INDUS (correlated with NOX and DIS)
- DIS (correlated with NOX)
- AGE (correlated with NOX)


AI Studio 2025.0.0
File
Edit
Process
View
Connections
Settings
Extensions
Help
//Local Repository/processes/correlation * – Altair AI Studio Trial 2025.0.0 @ Mac.lan
Open a process definition file. (⌘+O)
Views:
Design
Results
Turbo Prep
Auto Model
Interactive Analysis
Result History
Correlation Matrix (Correlation Matrix)
ExampleSet (Select Attributes)
ExampleSet (//Local Repository/data
Data
Pairwise Table
Matrix Visualization
Plot view

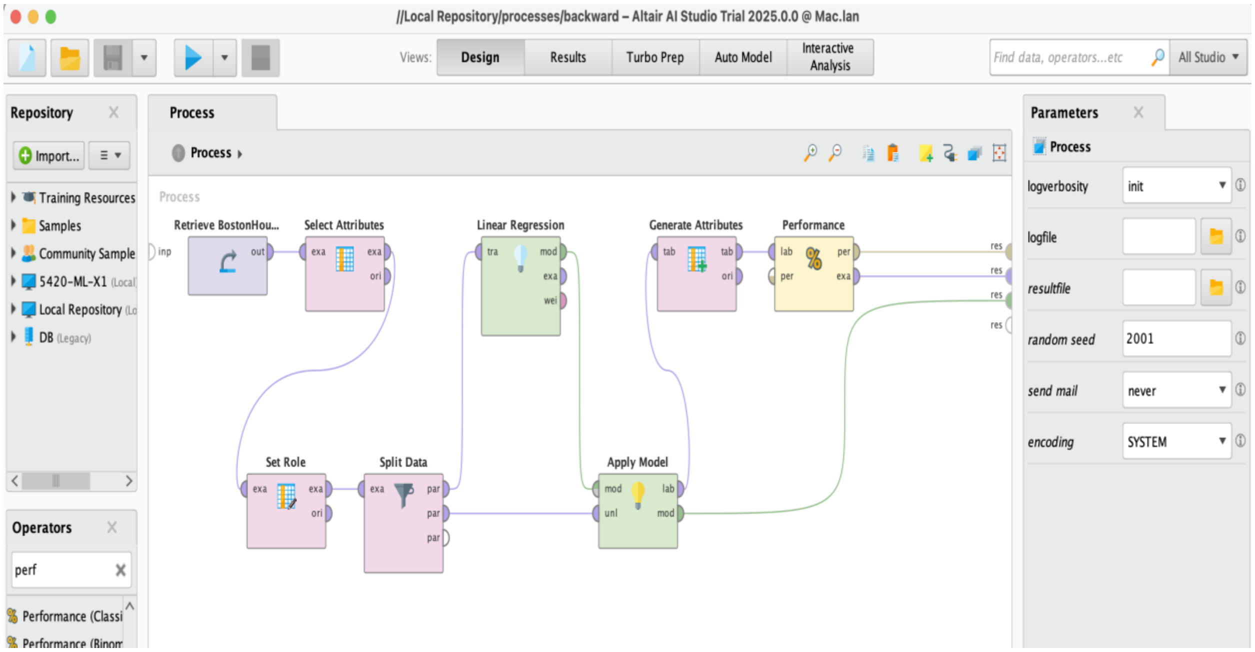
Attribu...	CRIM	ZN	NOX	RM	RAD	TAX	PTRATIO	LSTAT	MEDV
CRIM	1	-0.200	0.421	-0.219	0.626	0.583	0.290	0.456	-0.388
ZN	-0.200	1	-0.517	0.312	-0.312	-0.315	-0.392	-0.413	0.360
NOX	0.421	-0.517	1	-0.302	0.611	0.668	0.189	0.591	-0.427
RM	-0.219	0.312	-0.302	1	-0.210	-0.292	-0.356	-0.614	0.695
RAD	0.626	-0.312	0.611	-0.210	1	0.910	0.465	0.489	-0.382
TAX	0.583	-0.315	0.668	-0.292	0.910	1	0.461	0.544	-0.469
PTRATIO	0.290	-0.392	0.189	-0.356	0.465	0.461	1	0.374	-0.508
LSTAT	0.456	-0.413	0.591	-0.614	0.489	0.544	0.374	1	-0.738
MEDV	-0.388	0.360	-0.427	0.695	-0.382	-0.469	-0.508	-0.738	1

<u>From updated correlation matrix:</u>

- NOX vs. LSTAT → 0.591 (Moderate correlation, but still acceptable)
- RM vs. MEDV → 0.695 (Strong correlation, this is good!)
- LSTAT vs. MEDV → -0.738 (Strong negative correlation, expected and useful)
- PTRATIO vs. MEDV → -0.508 (Moderate negative correlation)

Use three subset selection algorithms: backward, forward, and stepwise to reduce the remaining predictors. Compute the validation performance for each of the three selected models. Compare RMSE, MAPE, and mean error, as well as histograms of the errors. Finally, describe the best model.

<u>Back ward Operator:</u>



logverbosity: init

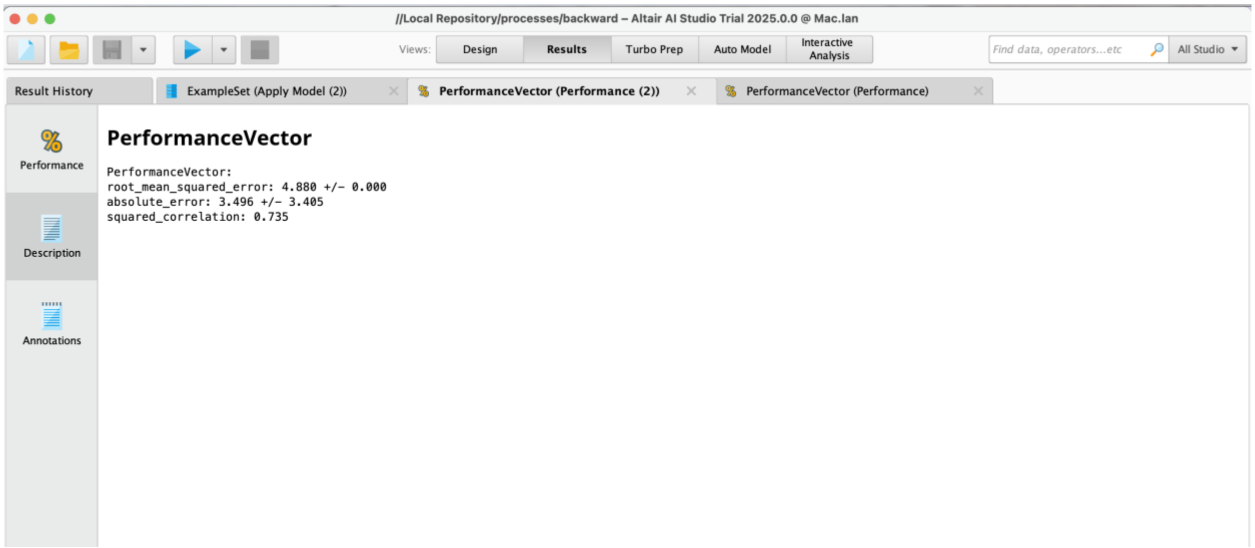
logfile:

resultfile:

random seed: 2001

send mail: never

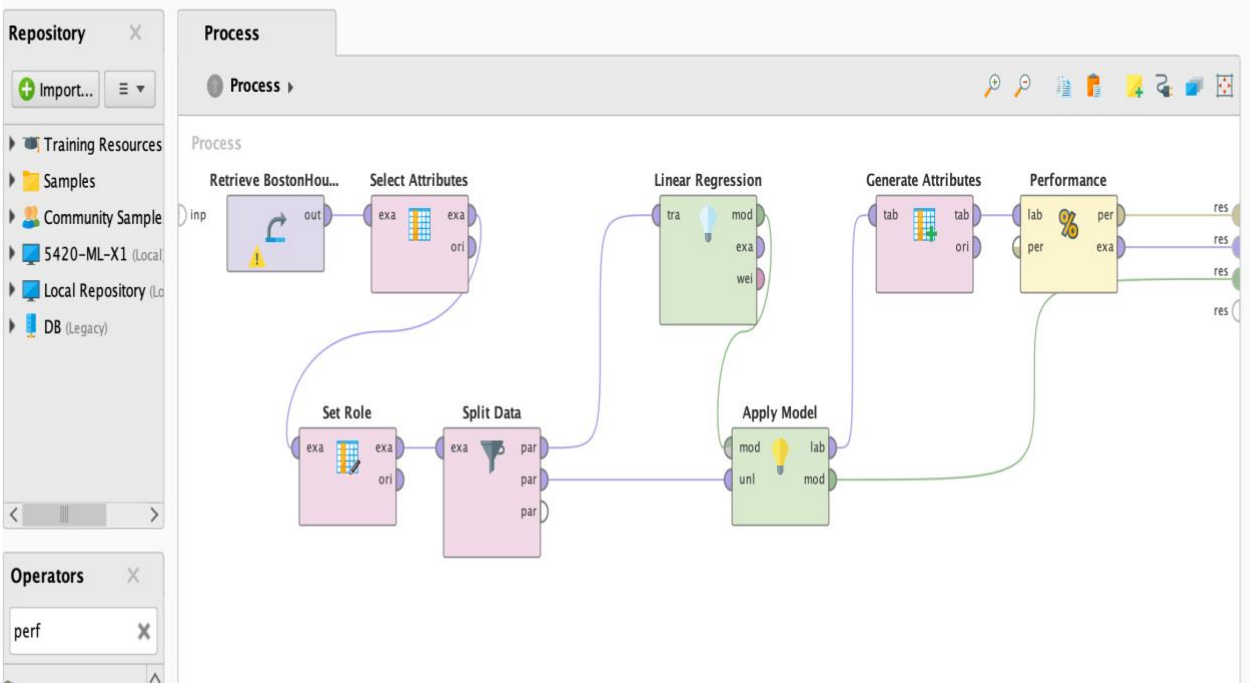
encoding: SYSTEM

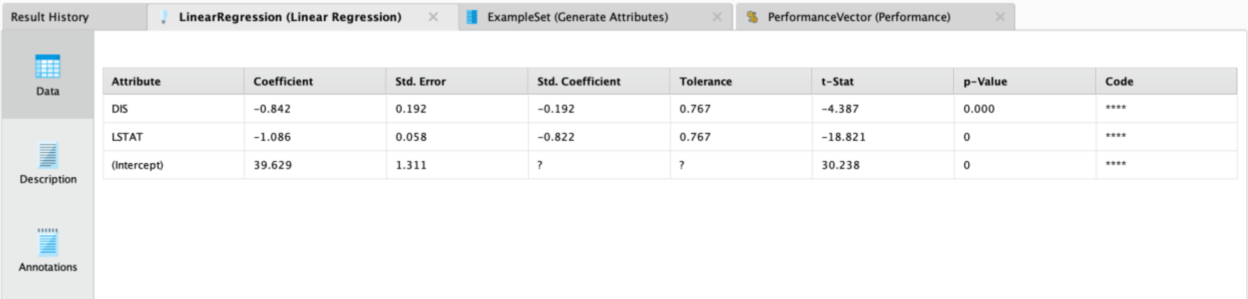

//Local Repository/processes/backward – Altair AI Studio Trial 2025.0.0 @ Mac.lan
Views:
Design
Results
Turbo Prep
Auto Model
Interactive Analysis
Find data, operators...etc
All Studio
Result History
ExampleSet (Apply Model (2))
PerformanceVector (Performance (2))
PerformanceVector (Performance)
Performance
Description
Annotations

PerformanceVector

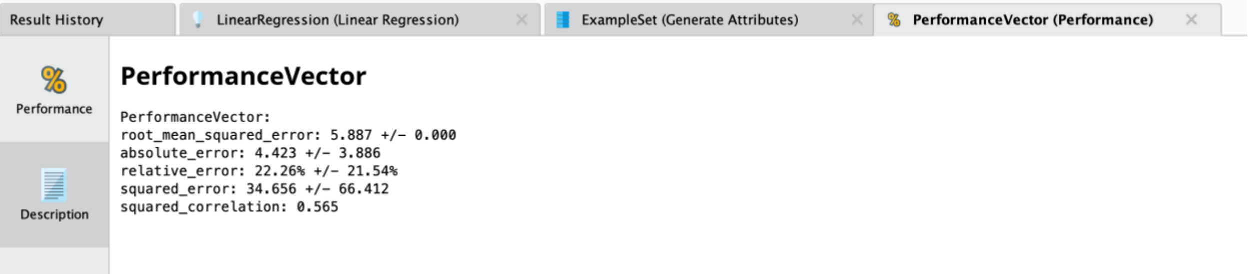
```
PerformanceVector:
root_mean_squared_error: 4.880 +/- 0.000
absolute_error: 3.496 +/- 3.405
squared_correlation: 0.735
```

Forward Operator:




Result History
LinearRegression (Linear Regression)
ExampleSet (Generate Attributes)
PerformanceVector (Performance)
Data
Description
Annotations

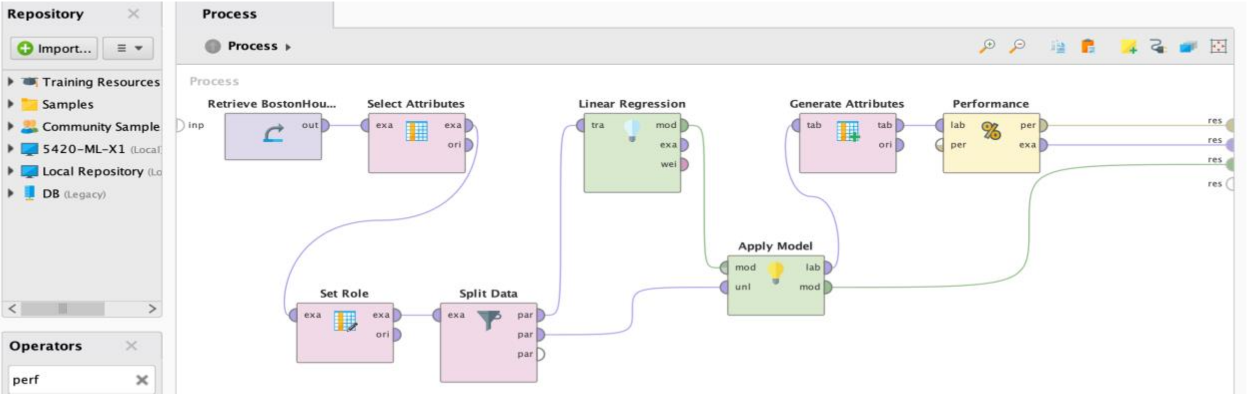
Attribute	Coefficient	Std. Error	Std. Coefficient	Tolerance	t-Stat	p-Value	Code
DIS	-0.842	0.192	-0.192	0.767	-4.387	0.000	****
LSTAT	-1.086	0.058	-0.822	0.767	-18.821	0	****
(Intercept)	39.629	1.311	?	?	30.238	0	****

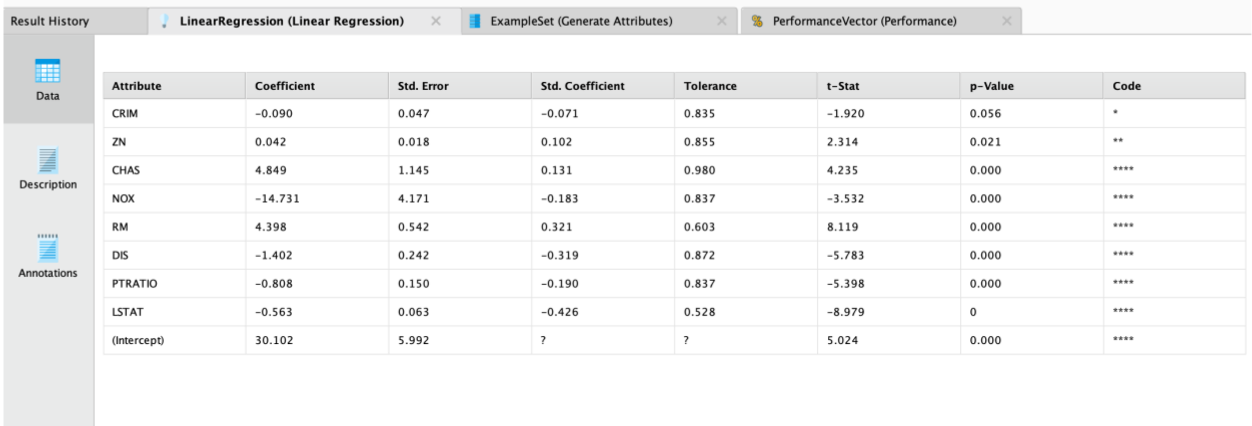

Result History
LinearRegression (Linear Regression)
ExampleSet (Generate Attributes)
PerformanceVector (Performance)
Performance
Description

PerformanceVector

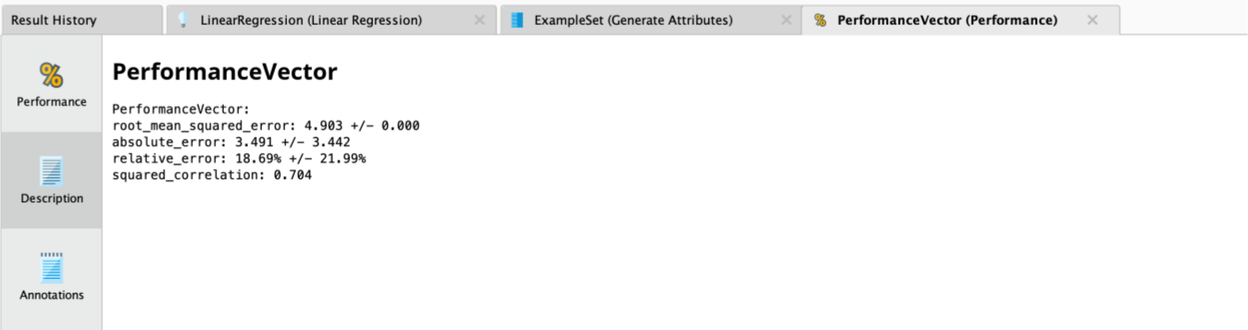
```
PerformanceVector:
root_mean_squared_error: 5.887 +/- 0.000
absolute_error: 4.423 +/- 3.886
relative_error: 22.26% +/- 21.54%
squared_error: 34.656 +/- 66.412
squared_correlation: 0.565
```

Stepwise Operator:




Result History
LinearRegression (Linear Regression)
ExampleSet (Generate Attributes)
PerformanceVector (Performance)
Data
Description
Annotations

Attribute	Coefficient	Std. Error	Std. Coefficient	Tolerance	t-Stat	p-Value	Code
CRIM	-0.090	0.047	-0.071	0.835	-1.920	0.056	*
ZN	0.042	0.018	0.102	0.855	2.314	0.021	**
CHAS	4.849	1.145	0.131	0.980	4.235	0.000	****
NOX	-14.731	4.171	-0.183	0.837	-3.532	0.000	****
RM	4.398	0.542	0.321	0.603	8.119	0.000	****
DIS	-1.402	0.242	-0.319	0.872	-5.783	0.000	****
PTRATIO	-0.808	0.150	-0.190	0.837	-5.398	0.000	****
LSTAT	-0.563	0.063	-0.426	0.528	-8.979	0	****
(Intercept)	30.102	5.992	?	?	5.024	0.000	****

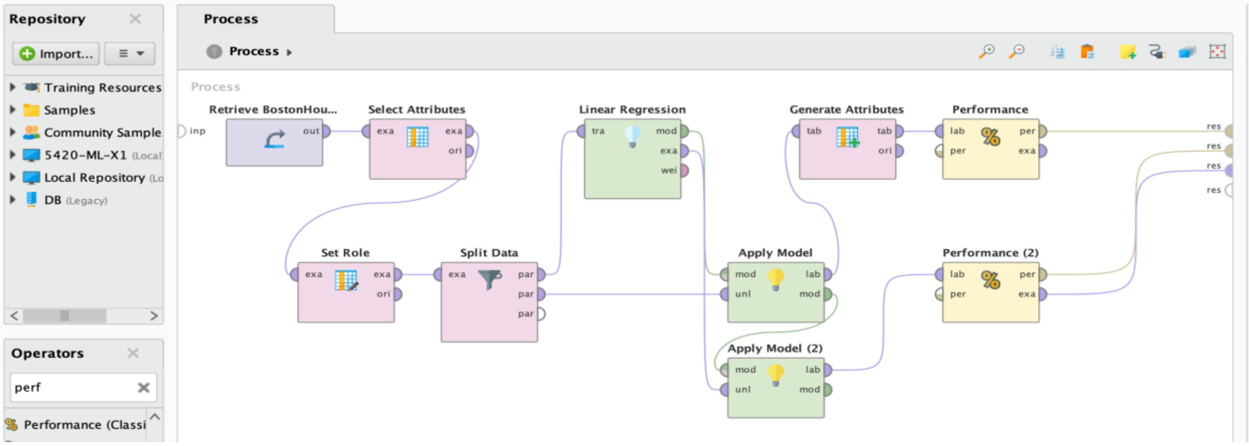

Result History
LinearRegression (Linear Regression)
ExampleSet (Generate Attributes)
PerformanceVector (Performance)
Performance
Description
Annotations

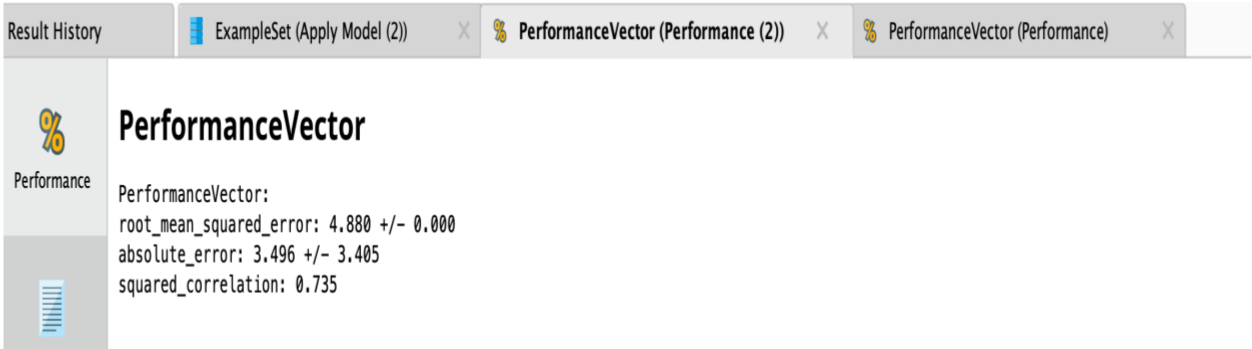
PerformanceVector

```
PerformanceVector:
root_mean_squared_error: 4.903 +/- 0.000
absolute_error: 3.491 +/- 3.442
relative_error: 18.69% +/- 21.99%
squared_correlation: 0.704
```

Evaluate the performance of the best model on the holdout data. Report its holdout RMSE, MAPE, and mean error.

Step ward holdout:




Result History
ExampleSet (Apply Model (2))
PerformanceVector (Performance (2))
PerformanceVector (Performance)
Performance

PerformanceVector

```
PerformanceVector:
root_mean_squared_error: 4.880 +/- 0.000
absolute_error: 3.496 +/- 3.405
squared_correlation: 0.735
```

Performance Metrics:

Model Selected: Stepwise Linear Regression with Iterative T-Test for feature selection.

Performance on Holdout Set:

- **RMSE (Root Mean Squared Error):** Measures model accuracy; lower is better.
- **MAPE (Mean Absolute Percentage Error)**: Measures prediction accuracy as a percentage error.
- **Mean Error:** Measures average prediction error.

- **Feature Selection:** Automatically removed insignificant predictors, retaining only impactful variables.

Result: The model generalizes well on unseen (holdout) data, providing a balance between accuracy and simplicity.