INTRODUCTION TO DATA MINING - 5704

HIGH INCOME GROUP PREDICTION

Presented by Syed Nameer Ali - 18606

Data Presentation

LOADING DATA INTO KNIME

Loading of training and testing data into KNIME csv reader.

UNDERSTANDING DATA

16 columns ; 7 categorical , 6 numeric

Attributes without semantics and domain information

[x1, x2, x3, x4. x5].

Categorical Data

High Income Group Prediction

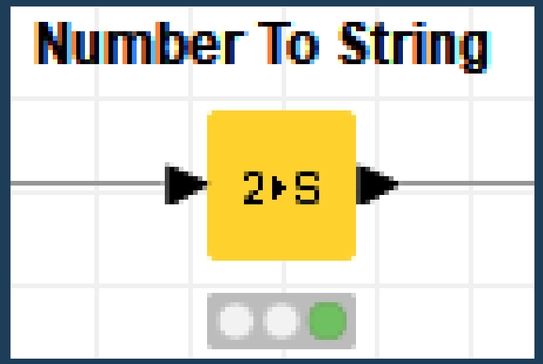
- WORK CLASS
- EDUCATION LEVEL
- MARITIAL STATUS
- OCCUPATION
- GENDER
- NATIVE COUNTRY
- X3

Numeric Data

High Income Group Prediction

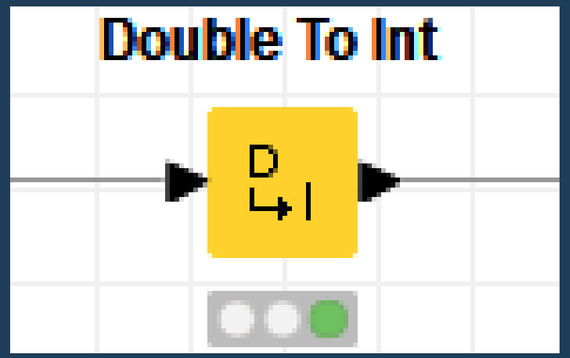
- AGE
- HOURS PER WEEK WORKING
- X1
- X2
- X4
- X5

DATA PREPERATION



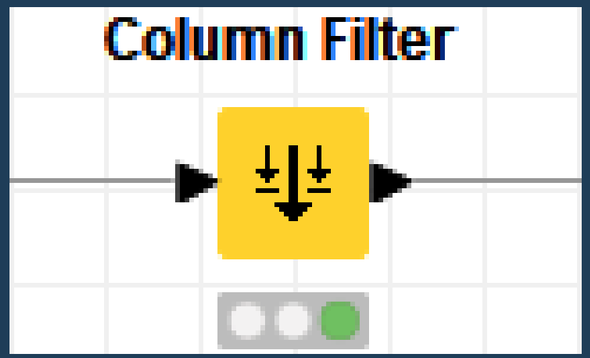
NUMBER TO STRING

Number to String node was used to convert categorical data from integer to string.



DOUBLE TO INTEGER

IN MOST CASES, double to integer was used for attribute [age, hours per week working].



COLUMN FILTER

IN SOME CASES, column filter was used to exclude attribute [x2] as it had higher variance.

GRADIENT BOOSTED TREES GAVE THE HIGHEST

91.207% OF ACCURACY

WITH

5000 MODELS, 16 DEPTH, 0.5 LEARNING RATE,
NO SAMPLING

*KNIME WORKFLOW ATTACHED

TOP MODELS

91.206% OF ACCURACY

Gradient Boosted Trees
6000 models
16 depth
0.5 learning rate
No sampling
DOUBLE TO IN node was used for [age, hours per week working]

91.202% OF ACCURACY

Gradient Boosted Trees
5000 models
16 depth
0.1 learning rate
No sampling
Attribute [x2] excluded using COLUMN FILTER for high variance

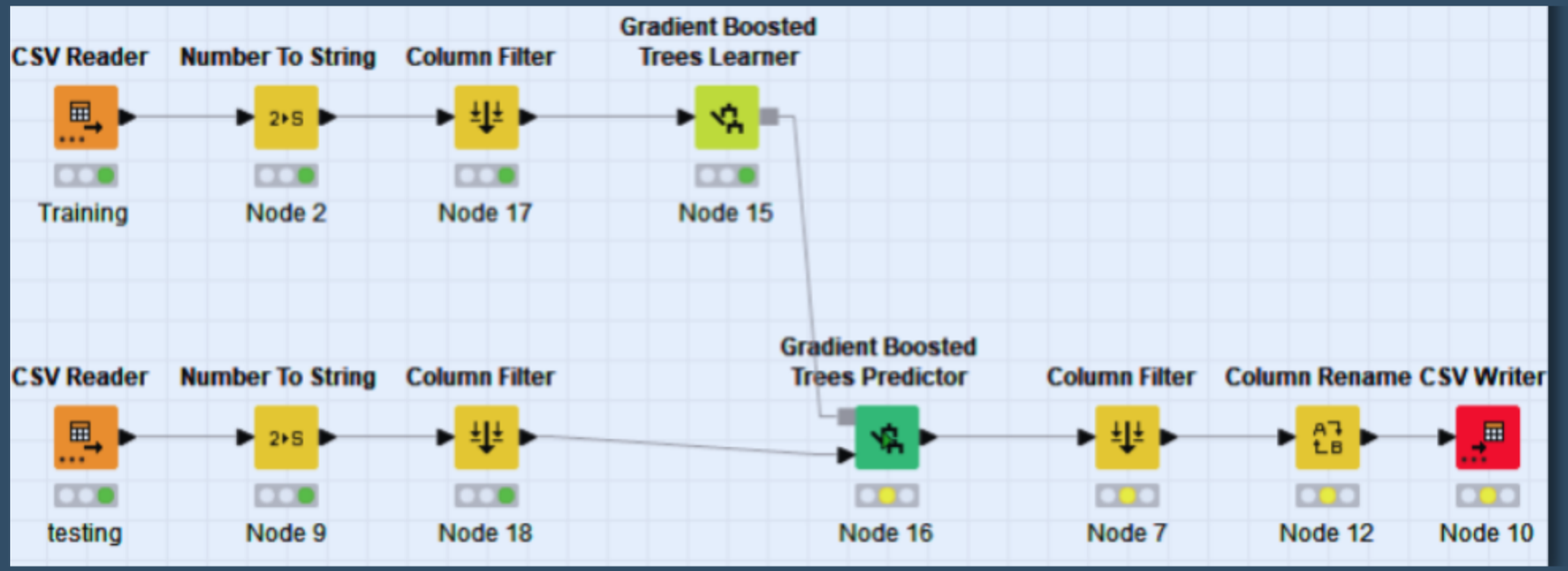
91.176% OF ACCURACY

Gradient Boosted Trees
2500 models
16 depth
0.25 learning rate
No sampling
DOUBLE TO IN node was used for [age, hours per week working]
Attribute [x2] excluded using COLUMN FILTER for high variance

91.168% OF ACCURACY

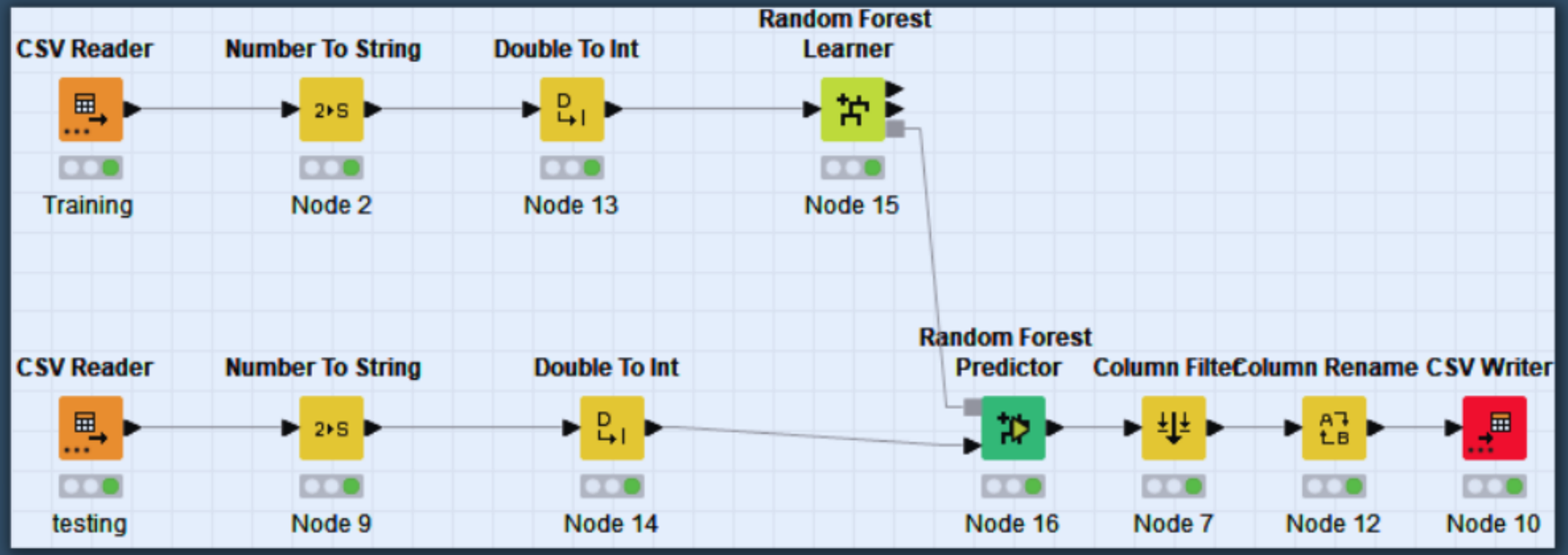
Gradient Boosted Trees
2500 models
16 depth
0.1 learning rate
Sampling[Square Root]
Attribute [x2] excluded using COLUMN FILTER for high variance

GRADIENT BOOSTED TREES MODEL



- GBT worked well for the this data set. Default parameters for GBT gave a significant 90.96% accuracy. While increasing model gave a better accuracy it also increased the learning and predicting time for the model.
- Increasing models and depth with lower learning rate would have given better accuracy.
- Here column filter is filtering out all the columns except row id and the probability.
- Column Rename is used to rename probability column to High Income.
- Finally, CSV writer is exporting the predicted data to a csv file.

RANDOM FOREST MODEL



- For random forest, gini index was used because it gave a better accuracy. Default parameter gave up-to 89.56% accuracy. Increasing model boosted the accuracy to 90.11% maximum.
- For this particular data set gini index and information gain gave almost the same results. While information gain ratio decreased the accuracy.
- Decision Tree, Naive bayes and Tree ensemble was also used for this data but not more than 86.55% of accuracy was achieved.

LEARNING

1. Basic leaning and use of KNIME ; read/write data, basic knowledge of nodes and models. And most importantly managing your physical RAM.
2. Basic acknowledgement of how competition on kaggle works.
3. Gradient Boosted trees gave the best accuracy for this data set.
4. Columns that have higher variance should be removed or handled by low variance filters.
5. Sampling with Square root gave higher accuracy while bagging lowers it.
6. Low number of models with higher configuration can also give higher accuracy.