

Cover sheet for submission of work for assessment

UNIT DETAILS

Unit name	Data Science Principle	Class day/time	1pm Friday	Office use only	
Unit code	COS10022	Assignment no.	2	Due date	12/11/202
Name of lecturer/teacher	Dang Vi Luan				
Tutor/marker's name				Faculty or school date stamp	

STUDENT(S)

Family Name(s)	Given Name(s)	Student ID Number(s)
----------------	---------------	----------------------

(1)	Dang	Vi Luan	103802759
-----	------	---------	-----------

(2)			
-----	--	--	--

(3)			
-----	--	--	--

(4)			
-----	--	--	--

(5)			
-----	--	--	--

(6)			
-----	--	--	--

DECLARATION AND STATEMENT OF AUTHORSHIP

1. I/we have not impersonated, or allowed myself/ourselves to be impersonated by any person for the purposes of this assessment.
2. This assessment is my/our original work and no part of it has been copied from any other source except where due acknowledgement is made.
3. No part of this assessment has been written for me/us by any other person except where such collaboration has been authorised by the lecturer/teacher concerned.
4. I/we have not previously submitted this work for this or any other course/unit.
5. I/we give permission for my/our assessment response to be reproduced, communicated, compared and archived for plagiarism detection, benchmarking or educational purposes.

I/we understand that:

6. Plagiarism is the presentation of the work, idea or creation of another person as though it is your own. It is a form of cheating and is a very serious academic offence that may lead to exclusion from the University. Plagiarised material can be drawn from, and presented in, written, graphic and visual form, including electronic data and oral presentations. Plagiarism occurs when the origin of the material used is not appropriately cited.

Student signature/s

I/we declare that I/we have read and understood the declaration and statement of authorship.

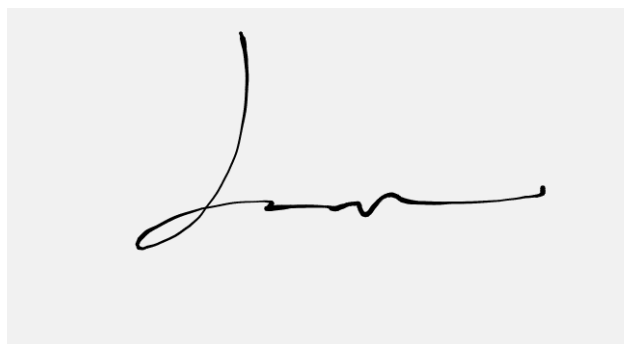
(1) **Nguyen Dinh Nhat Minh (peer reviewer)**

(4)



(2) **Dang Vi Luan (author)**

(5)



(3)

(6)

Further information relating to the penalties for plagiarism, which range from a formal caution to expulsion from the University is contained on the Current Students website at www.swin.edu.au/student/

Copies of this form can be downloaded from the Student Forms web page at www.swinburne.edu.au/studentforms/

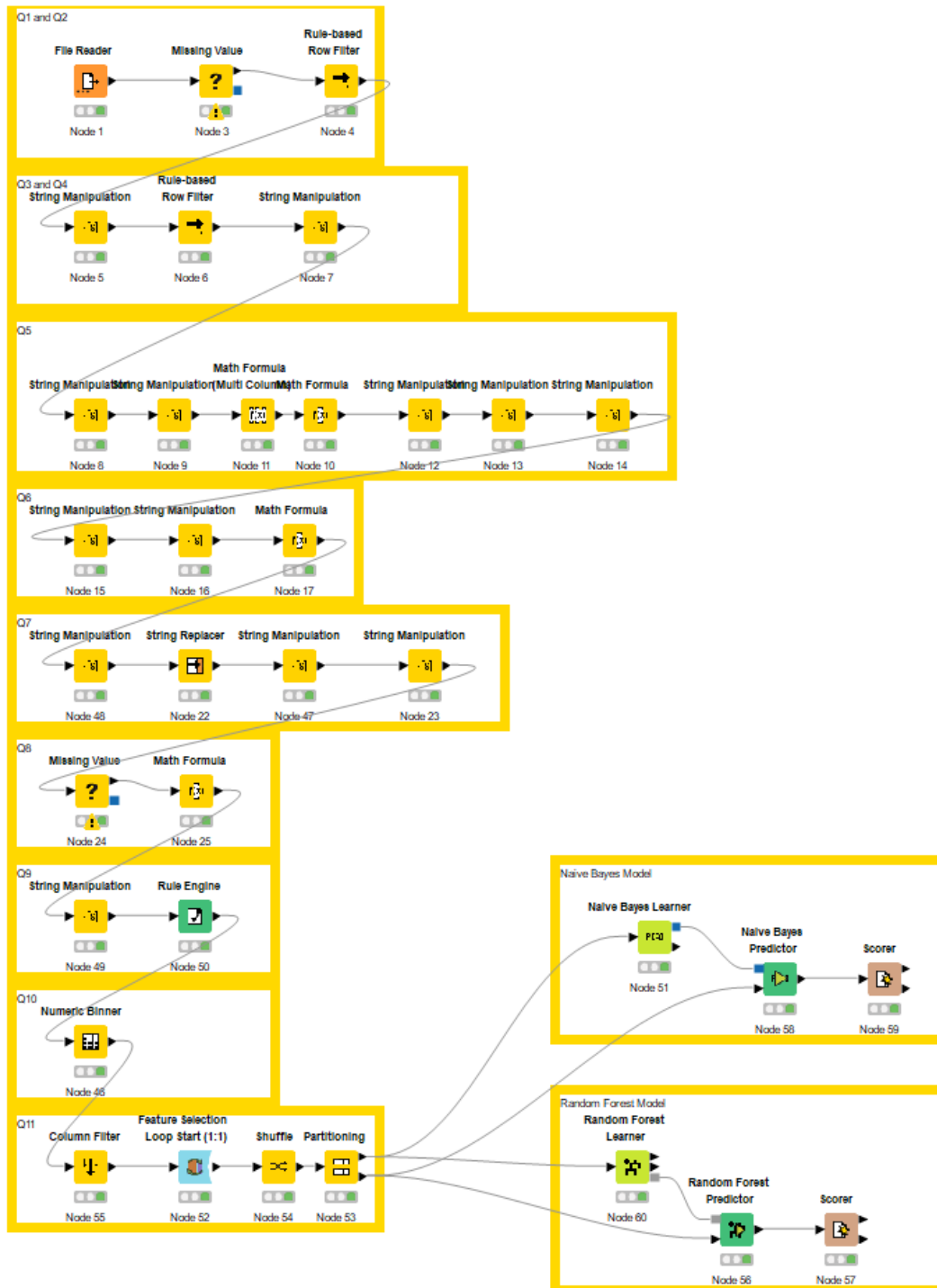
PAGE 1 OF 1

COS10022 Data Science Principles

Assignment 2 – Vi Luan Dang – 103802759

1. Introduction

Proficiency in conducting, utilizing, and elucidating predictive models and their outcomes is crucial for students aspiring to pursue a career in the IT and data science field. In this assignment, I aim to showcase my comprehension of fundamental data science concepts by constructing predictive models. Each section will encompass significant steps involved in this process, and within each section, I will address the assignment's provided questions. The whole workflow for this assignment are as follows:



2. Data Cleaning

Data preparation plays a pivotal role in building predictive models. The quality and reliability of the data directly impact the accuracy and effectiveness of our output, therefore, the very first step would be to clean our data.

In order to process the assignment's input, we will utilize the "data_2023.csv" file by employing the "File Reader" tool. As the provided data contains a wide range of attributes, it is crucial to remove irrelevant attributes to our predicted target – "Credit_Score". Among the attributes, "Names" is irrelevant to predicting "Credit_Score" as it is personal information and does not affect the entity's creditworthiness. Moreover, the requirement states that there is a limit of 600 distinct nominal values per attribute which "Names" clearly violates as there are more than 600 distinct names in the data set.

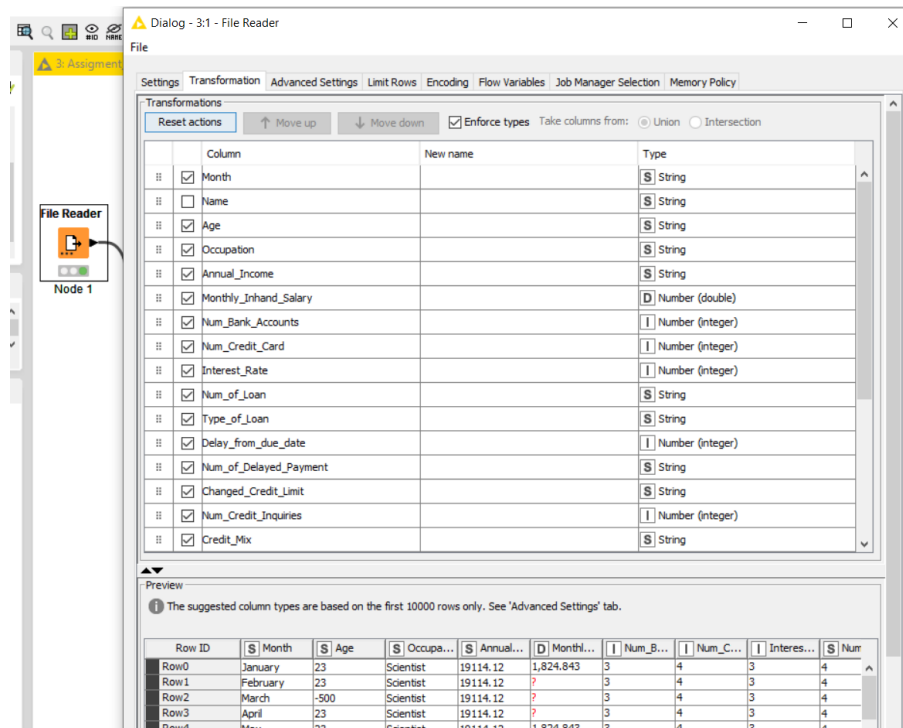


Figure 1: Data Preparation node

Answers to assignment's questions

- 1) The excluded attribute is “Name”. The reason is that does not affect the entity’s creditworthiness and the requirement states that there is a limit of 600 distinct nominal values per attribute which “Names” clearly violates as there are more than 600 distinct names in the data set.
- 2) Remove selected attributes that contain missing values as follows:

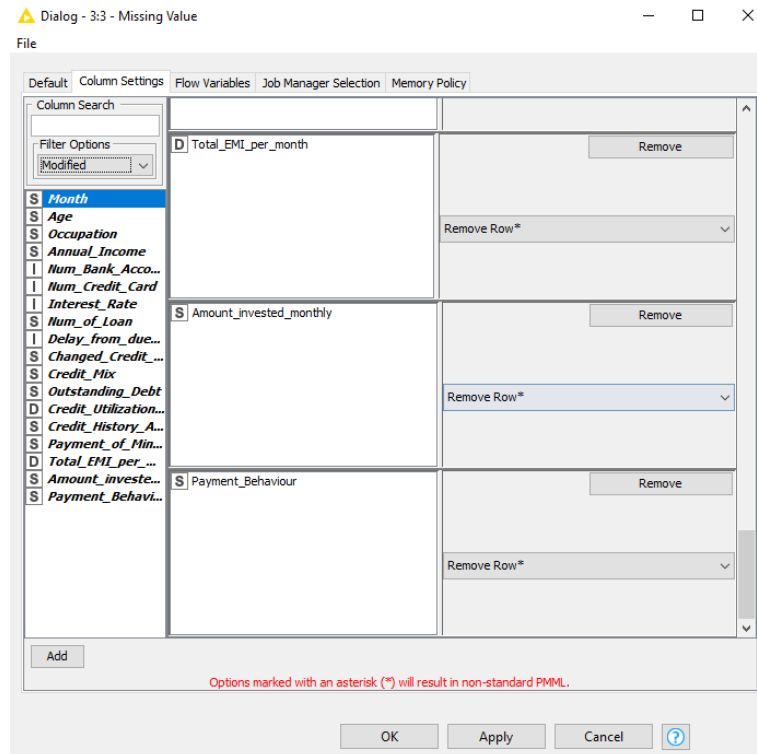


Figure 2: Removed attributes that contain missing values.

All the attributes that contain missing values are shown in the “Modified” filter option of the “Missing Value” node, should there be missing values in these categories, the rows will be eliminated due to “Remove Row*” option. Additionally, infeasible values will also be eliminated using “Rule-based Row Filter” as follows:

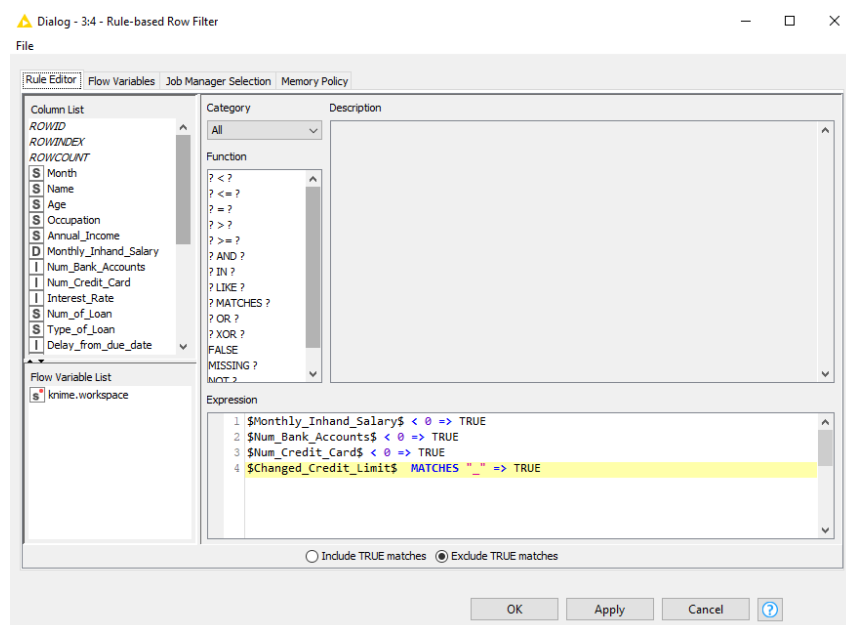


Figure 3: Infeasible values are removed using expression

- 3) **Age Cleansing:** The “Age” attribute is cleansed using regular expression to search for non-numerical data and replace it with an empty string, additionally only values which are greater than 0 and equal or smaller than 120 are accepted. These cleansing rules are enforced using “String Manipulation” node and “Rule-based Row Filter” node.

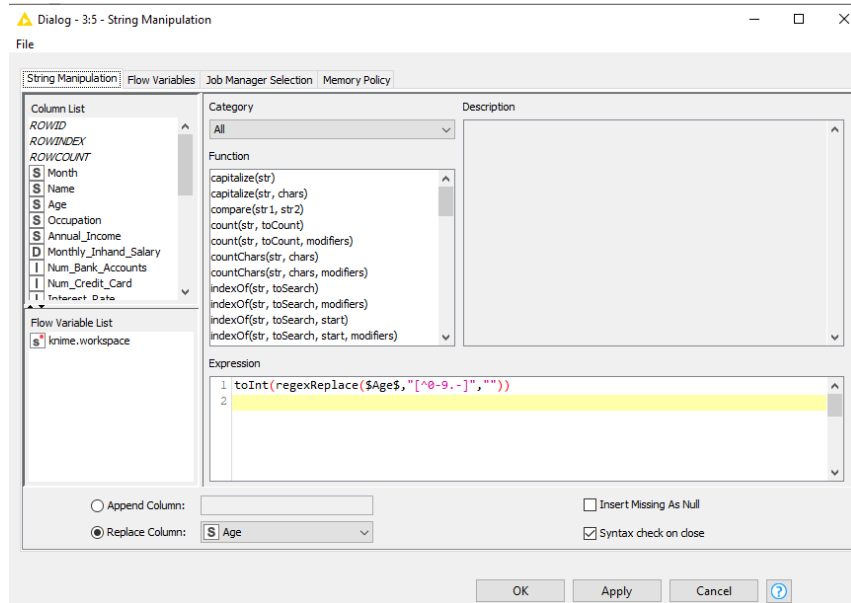


Figure 4: “Age” attribute’s cleansing rules

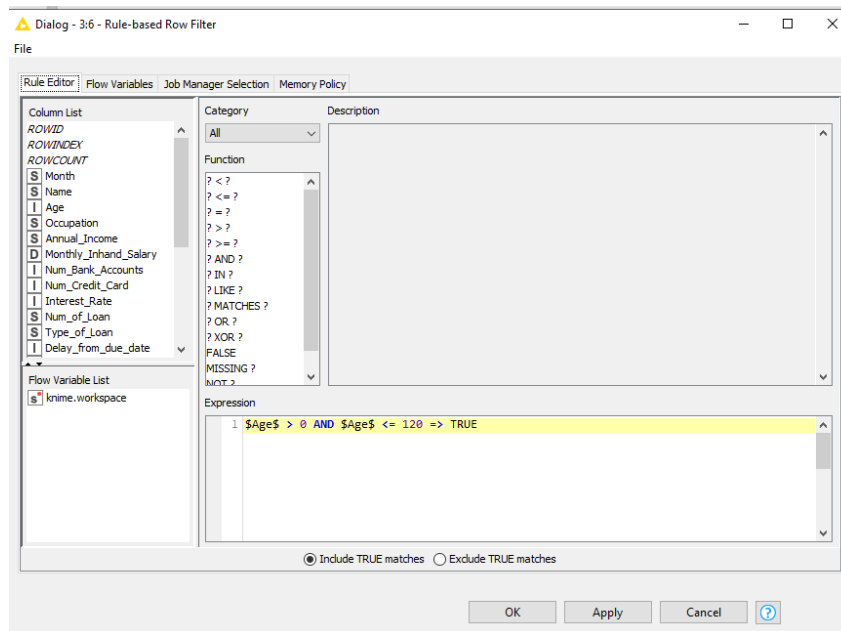


Figure 5: Range of data acceptance for “Age” attribute

- 4) **Annual_Income Cleansing:** Similarly, “Annual_Income” attribute will also need to be removed of any non-numerical value. The same regular expression will be applied for this attribute using “String manipulation” node.

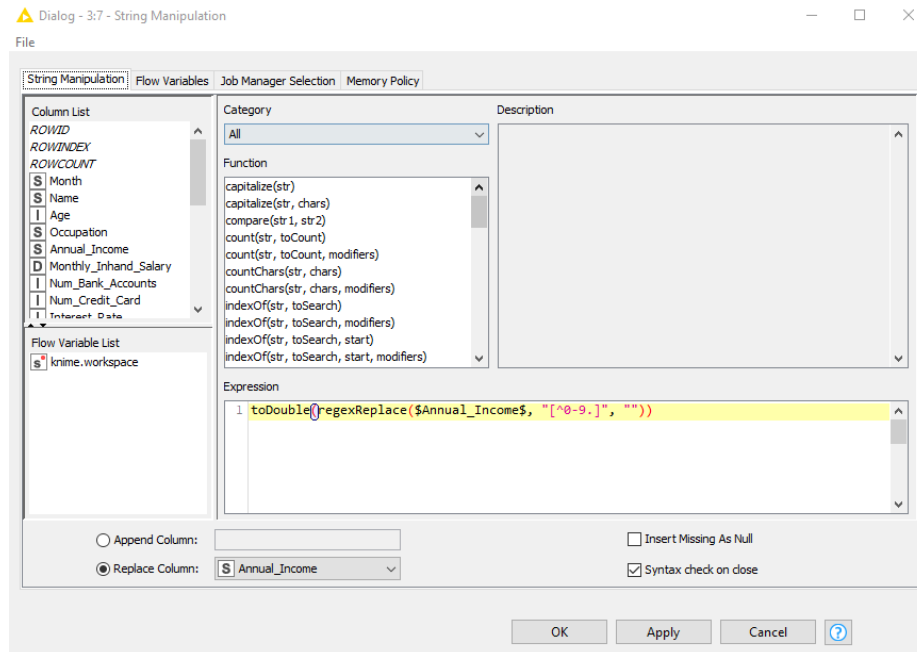


Figure 6: “Annual_Income” attribute’s cleansing rule.

- 5) **Occupation, Num_Bank_Accounts, Num_Credit_Card, Num_of_Loan, Num_of_Delayed_payment, Credit_mix, Outstanding_debt cleansing:**

Firstly, we will replace any value in Occupation which contains “_____” as value using “String Manipulation” node

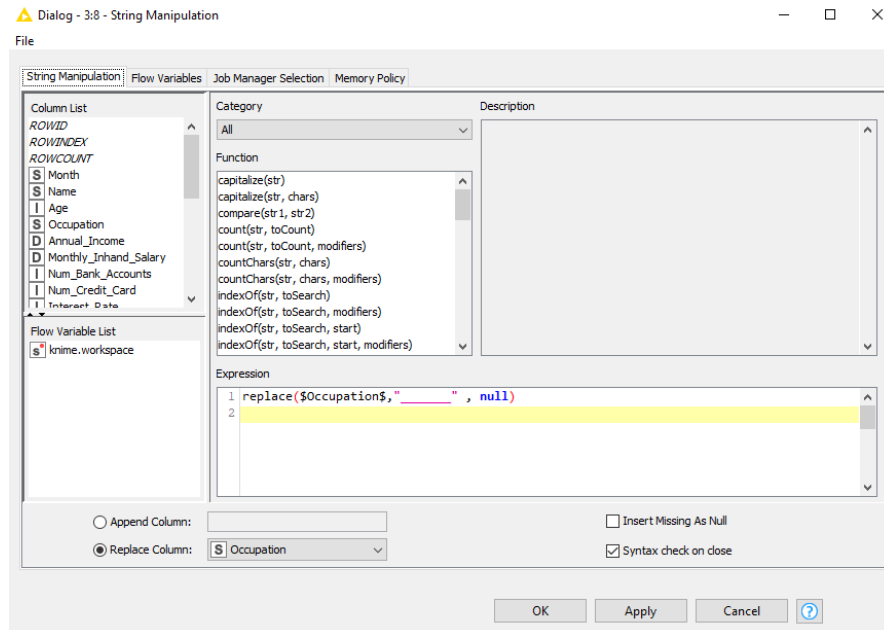


Figure 7: “Occupation” attribute’s cleansing rule

Then we will apply the cleansing rule from question 4 to “Num_of_loan” attribute. We will also applied another expression to check for any “Num_of_Loan” that is negative and converse them accordingly

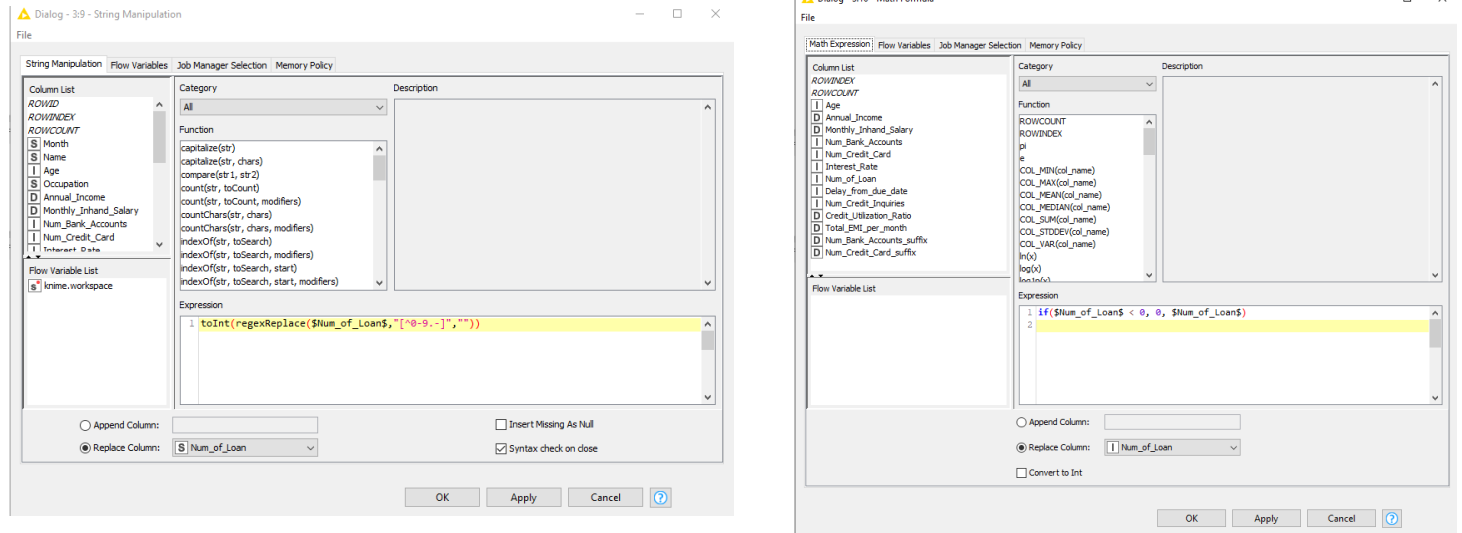


Figure 8: “Num_of_Loan” attribute’s cleansing rule

After that we will converse “Num_Bank_Accounts” and “Num_Credit_Card” value to absolute.

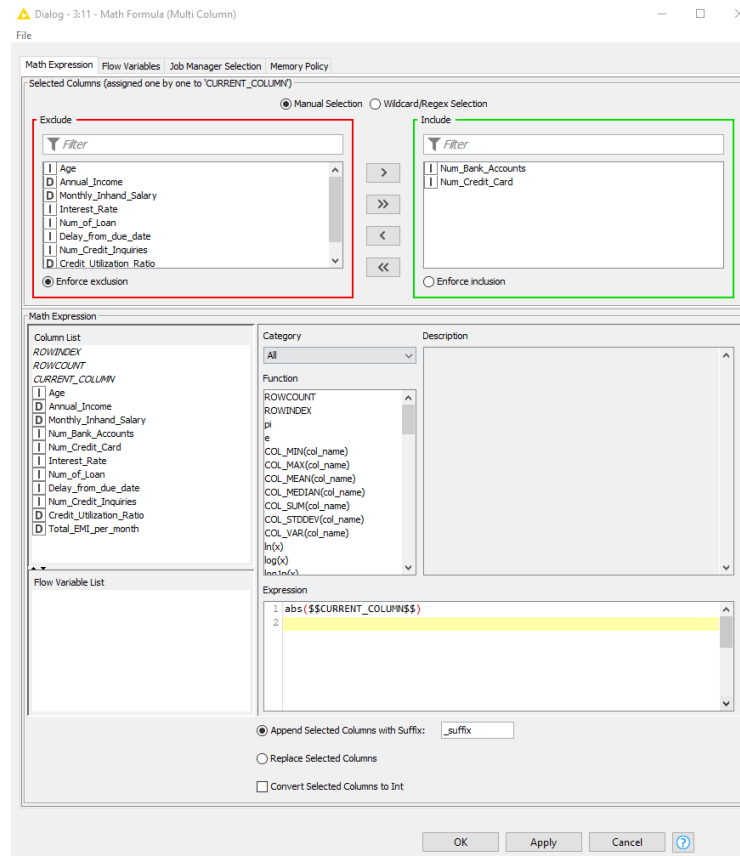


Figure 9: Conversion of “Num_Bank_Accounts” and “Num_Credit_Card” values

“Num_of_Delayed_Payment” and “Outstanding_Debt” should not contains any non-numerical value, we can ensure this using cleansing rule from question 4.

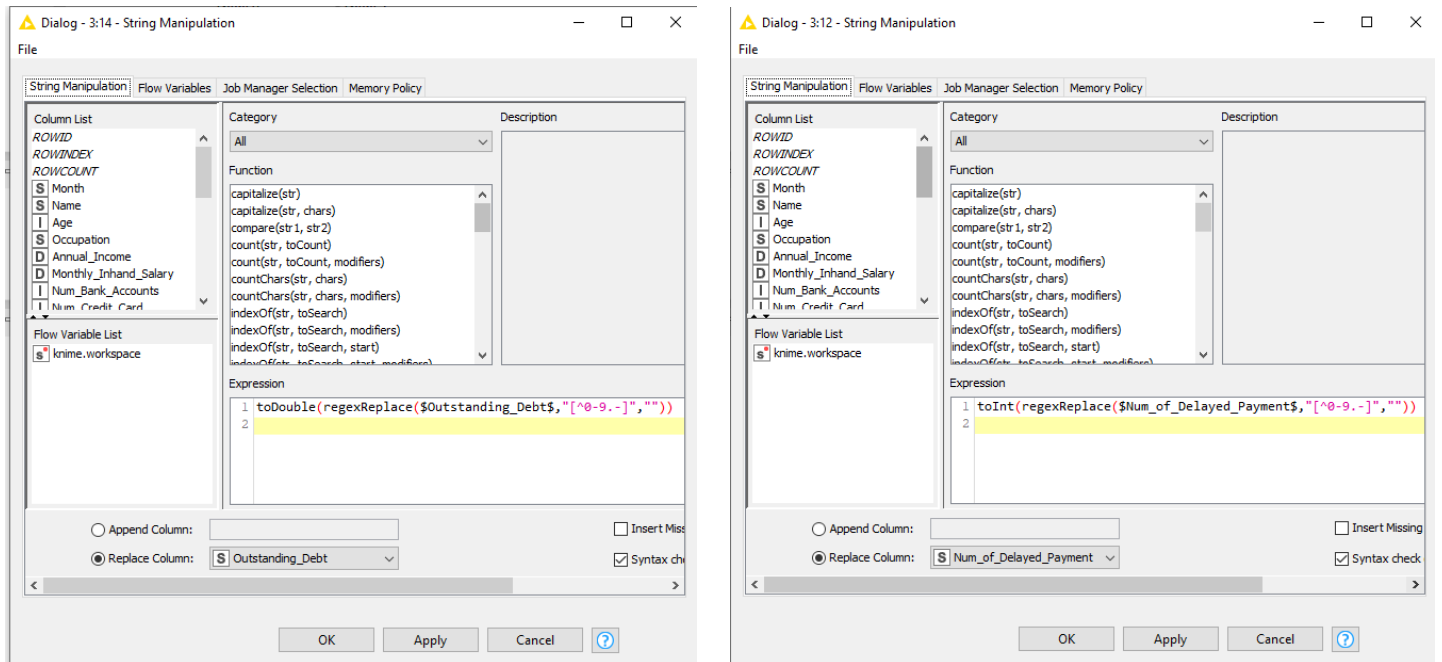


Figure 9: Cleansing rule for “Outstanding_Debt” and “Num_of_Delayed_Payment”

Finally, abnormal value of “Credit_Mix” value should be conversed to “Unknown”

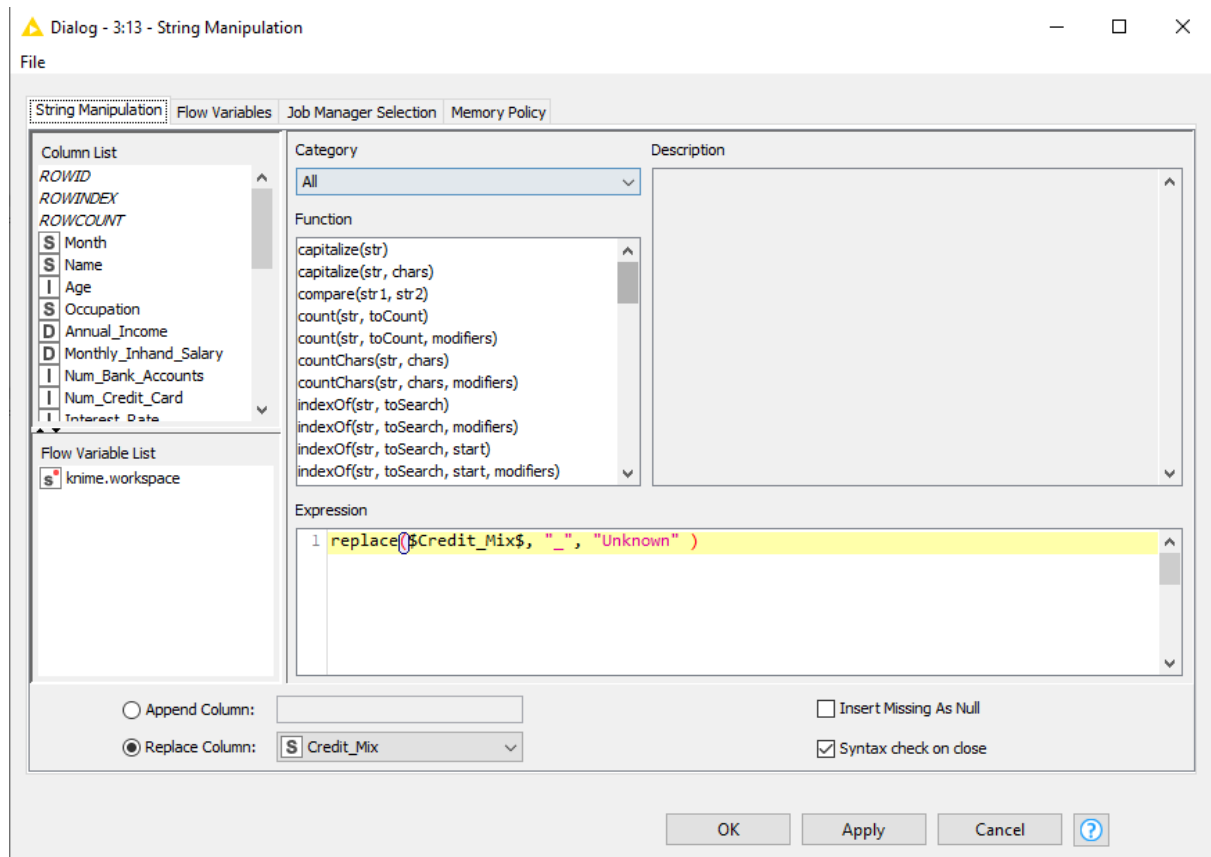


Figure 10: Cleansing rule for “Credit_Mix”

- 6) **Calculation for the “Total_CHA” value:** In order to achieve this, we will need extract a specific portions of the string variable “Credit_History_Age”, remove any leading/trailing whitespace, and convert it to an integer value then perform the calculation based on the years and months of the “Credit_History_Age”.

First we will extract the years from “Credit_History_Age” attribute and convert it to interger.

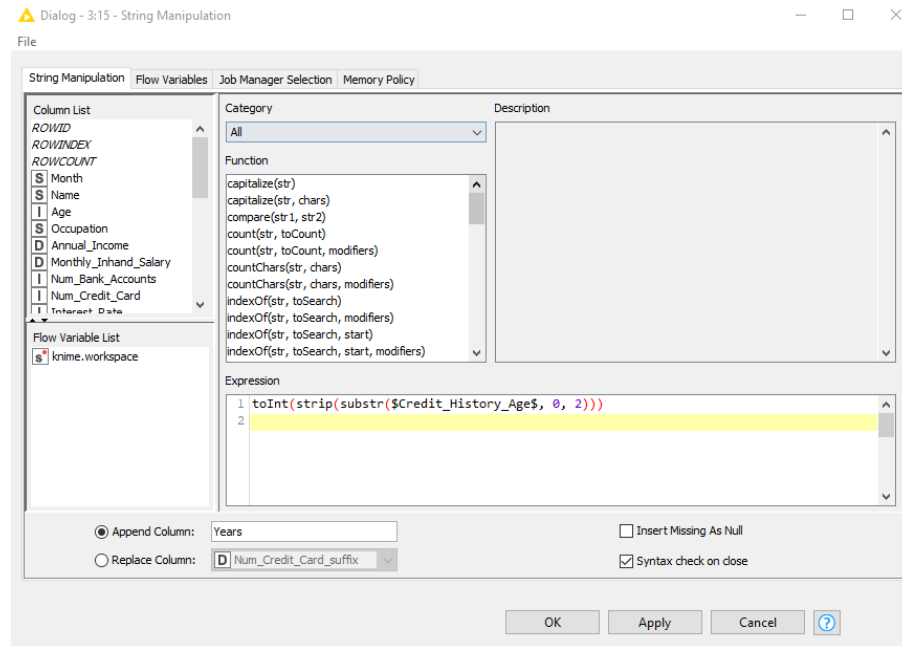


Figure 11: Obtain the “Years” value.

Then we will need to obtain the “month” value of this attribute. As all the values in this field follow the pattern of “and x months” we will spot the first occurrence of the letter “d” and take 3 characters after it for our month (we take 3 characters for fear of any 2-digits month in our data set), of course the value will need to be in integer for future calculation.

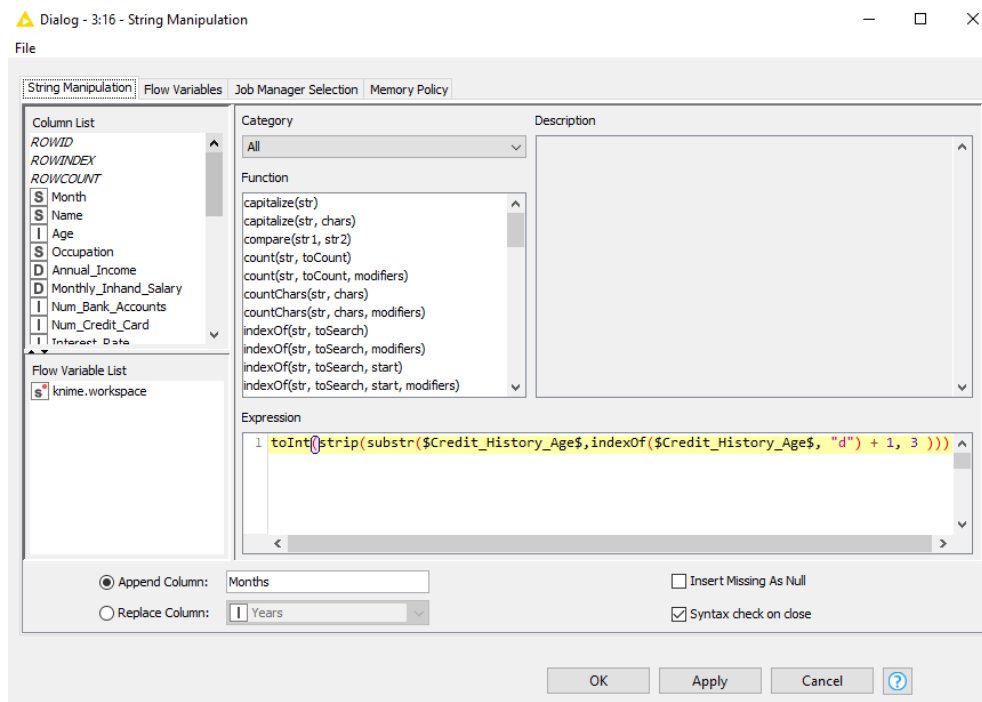


Figure 12: Obtain the “Months” value.

Finally, we will calculate our “Total_CHA” value using formula (Years * 12) + Months.

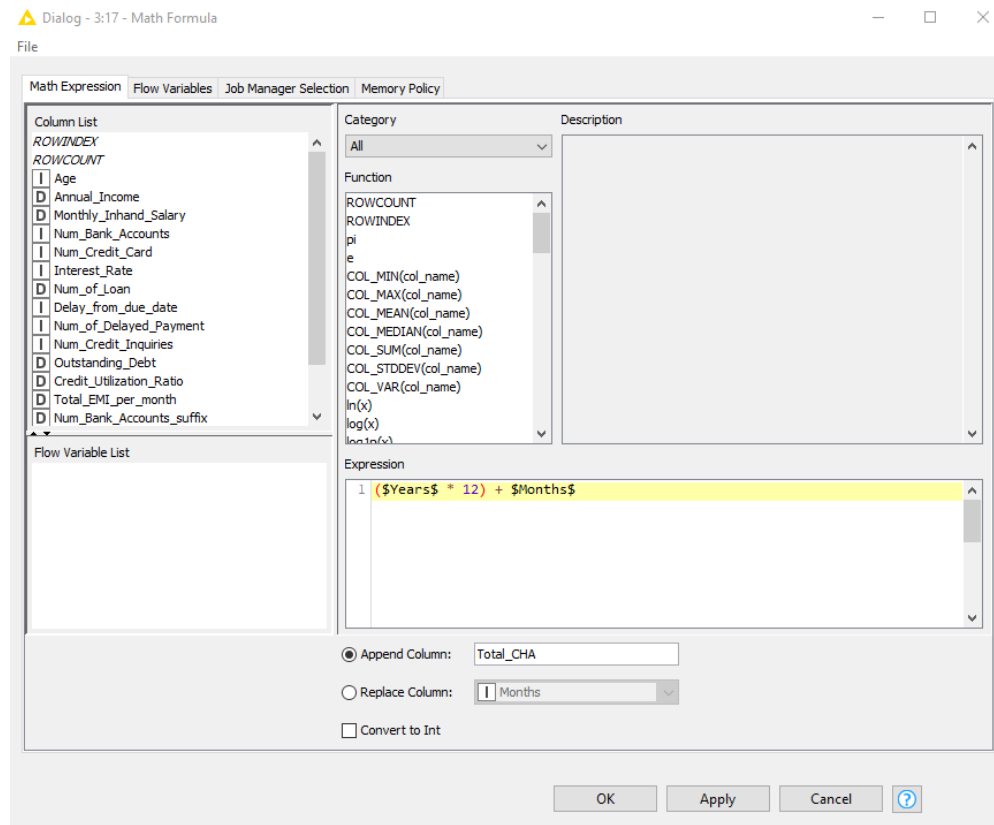


Figure 13: Calculation of “Total_CHA” value.

7) “Amount_invested_monthly”, “Payment_Behaviour”, “Monthly_Balance”, “Changed_Credit_Limit” cleansing rule:

Firstly, we will remove non-numerical value for attributes that need to be convert to double.

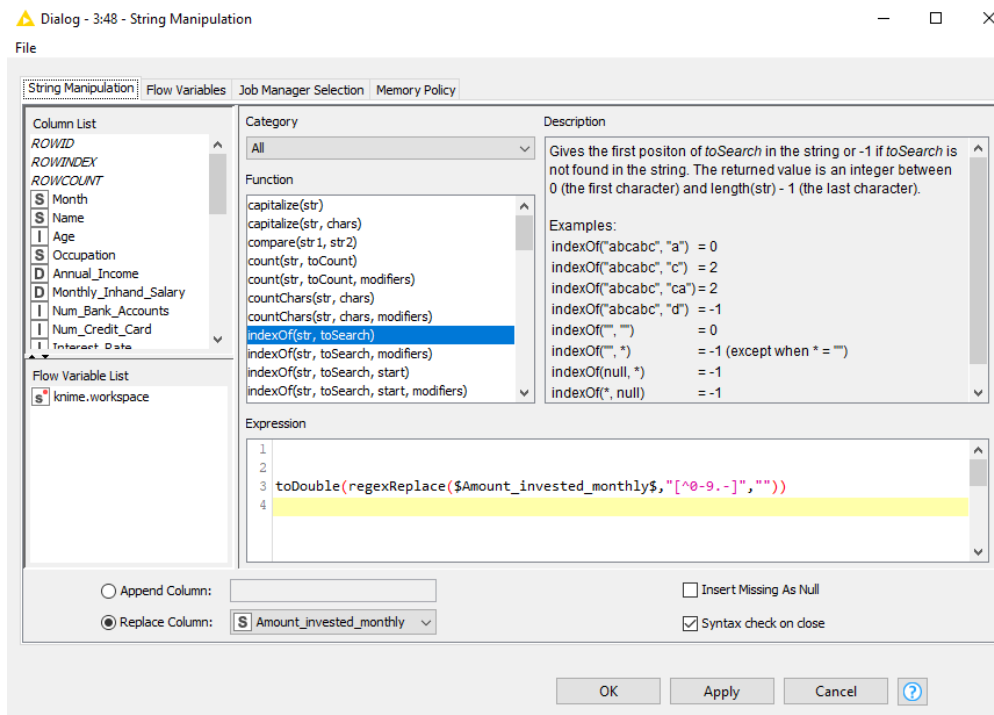


Figure 14: Cleansing rule for “Amount_invested_monthly”

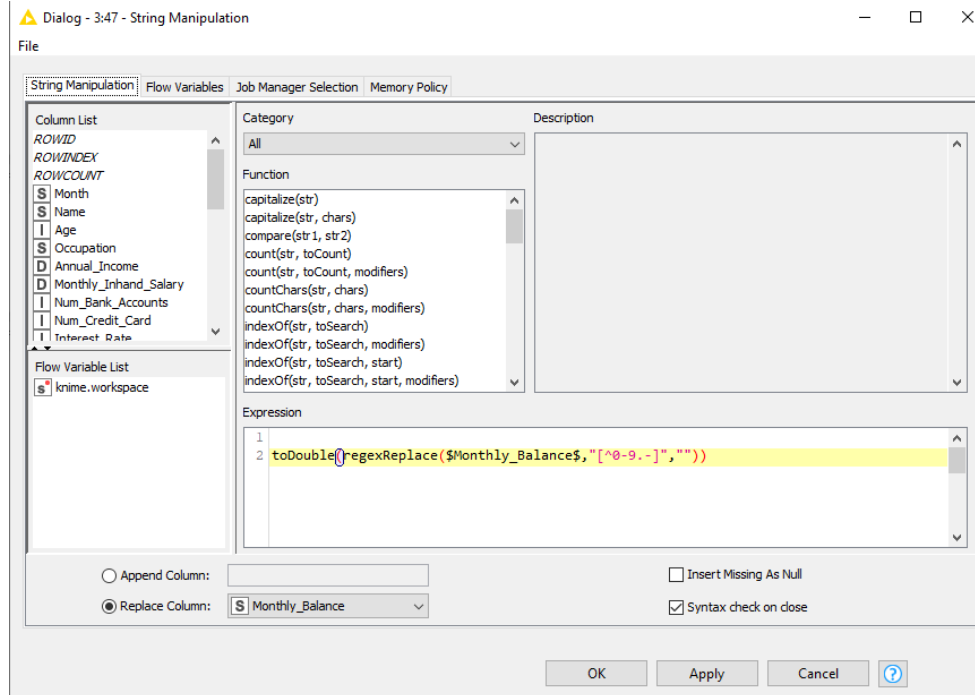


Figure 15: Cleansing rule for “Monthly_Balance”

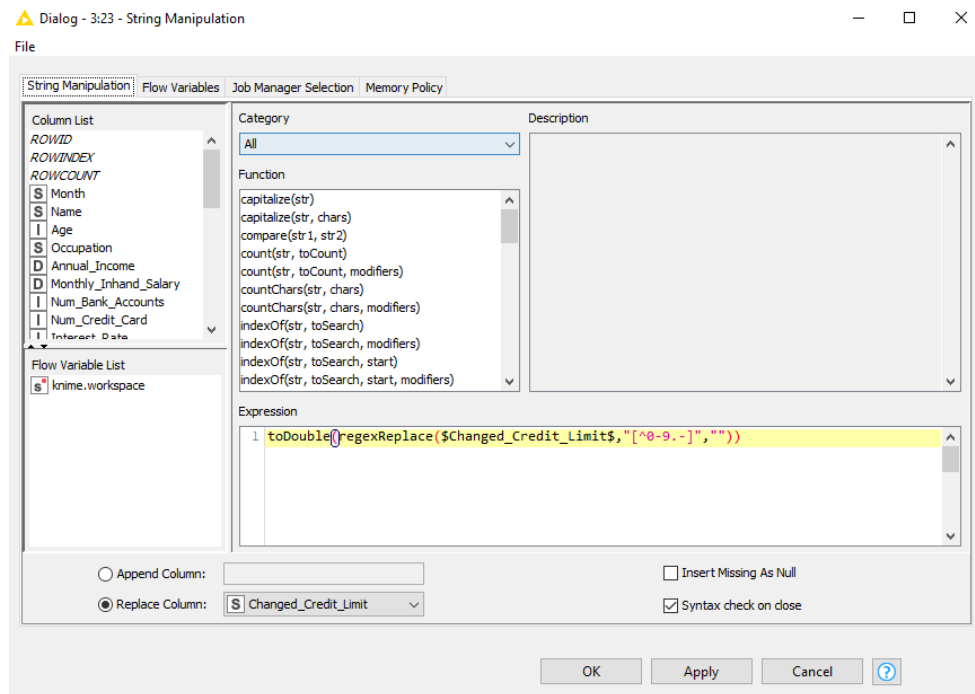


Figure 16: Cleansing rule for “Changed_Credit_Limit”

Finally we will need to omit requested pattern for “Payment_Behaviour” attribute

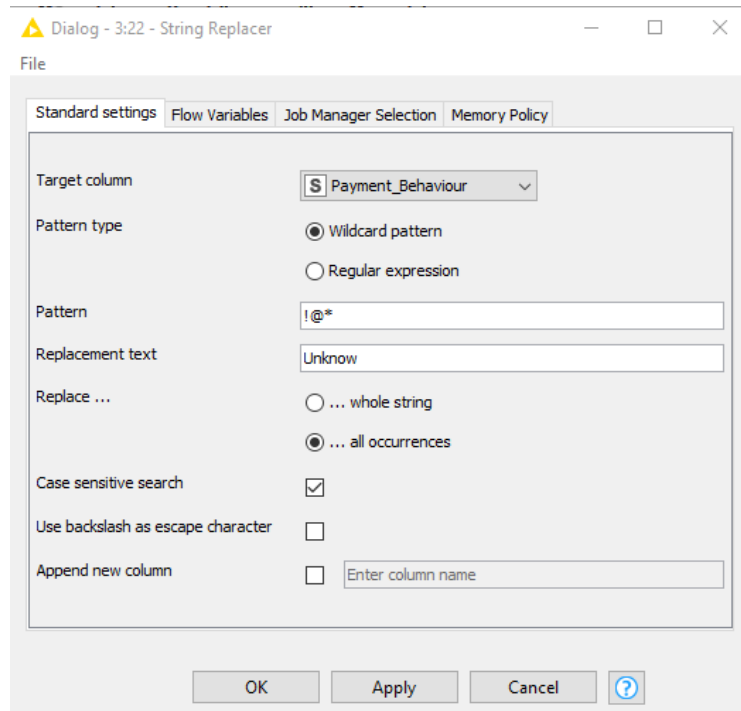


Figure 17: Omit undesired pattern for “Payment_Behaviour” attribute.

- 8) We will replace missing data in all string type values to “Next Value” and all numerical type values to “Previous Value”. Then “Monthly_Balance” will be applied with rule so that negative number will become 0.

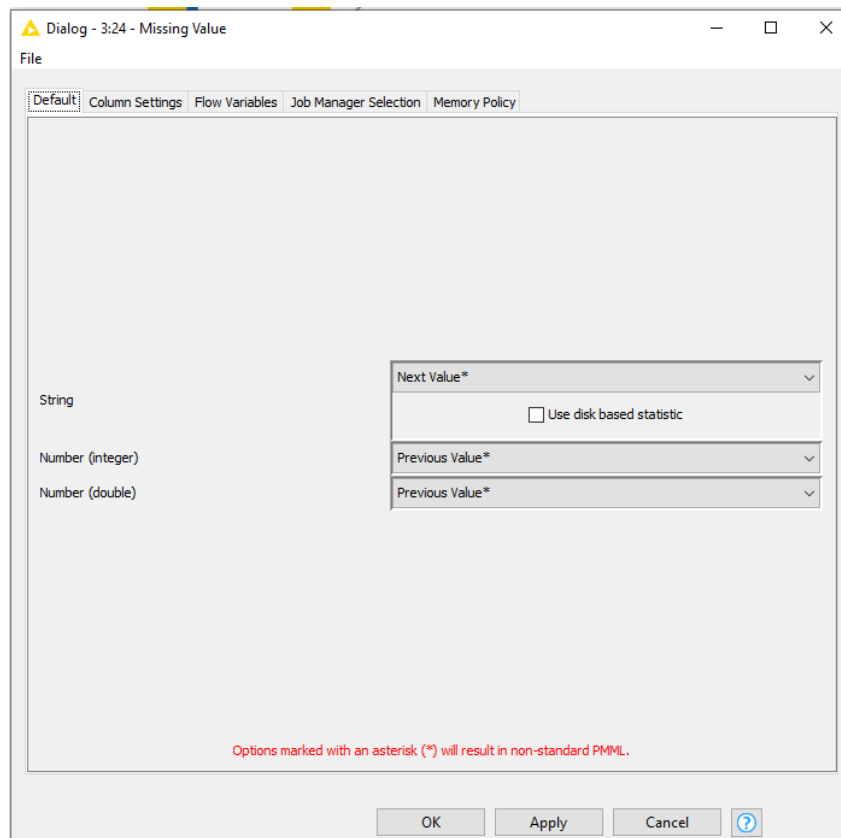


Figure 18: Cleansing rule for all string type value and numerical type value

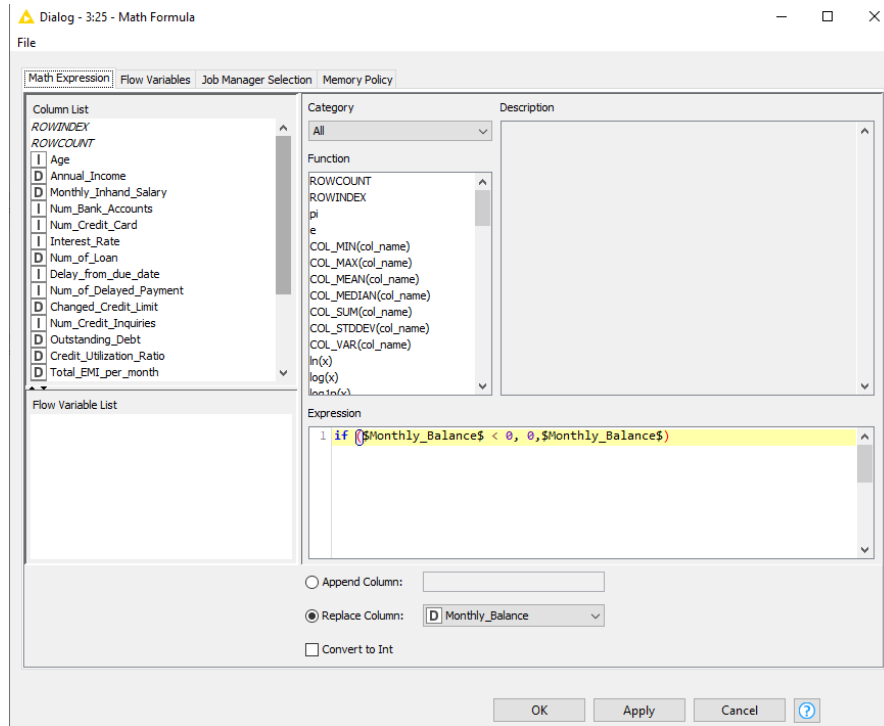


Figure 19: Expression for “Monthly_Balance”

- 9) We will simplify “Type_of_Loan” attribute so that only the first loan type will be kept if there are multiple loan types separated by comma, single loan type will be kept the same.

To achieve this we will need to extract the first loan type using an expression. This expression will firstly discard all single type loan, therefore, we will create a new column as a placeholder for multiple loan types.

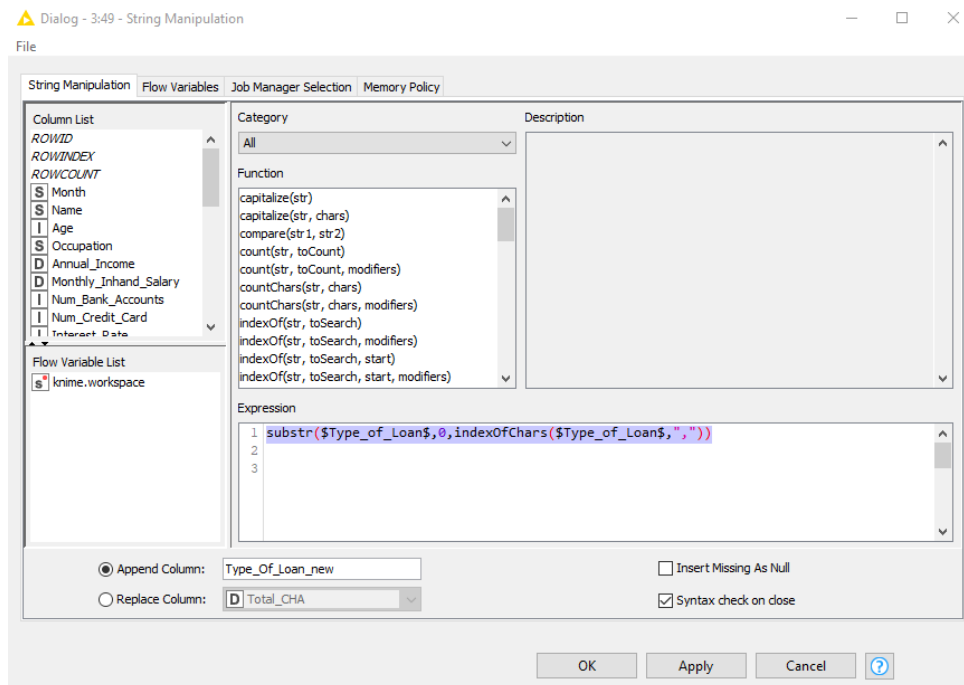


Figure 20: Extract the first loan type.

After that we will use “Rule Engine” node to check whether our loan type is multiple loan types of single loan type, if it is the former, we will take the placeholder attribute as our new value, if it is the latter, we will keep the old value. Finally we will have our simplified loan type as “New_Type_Of_Loan”

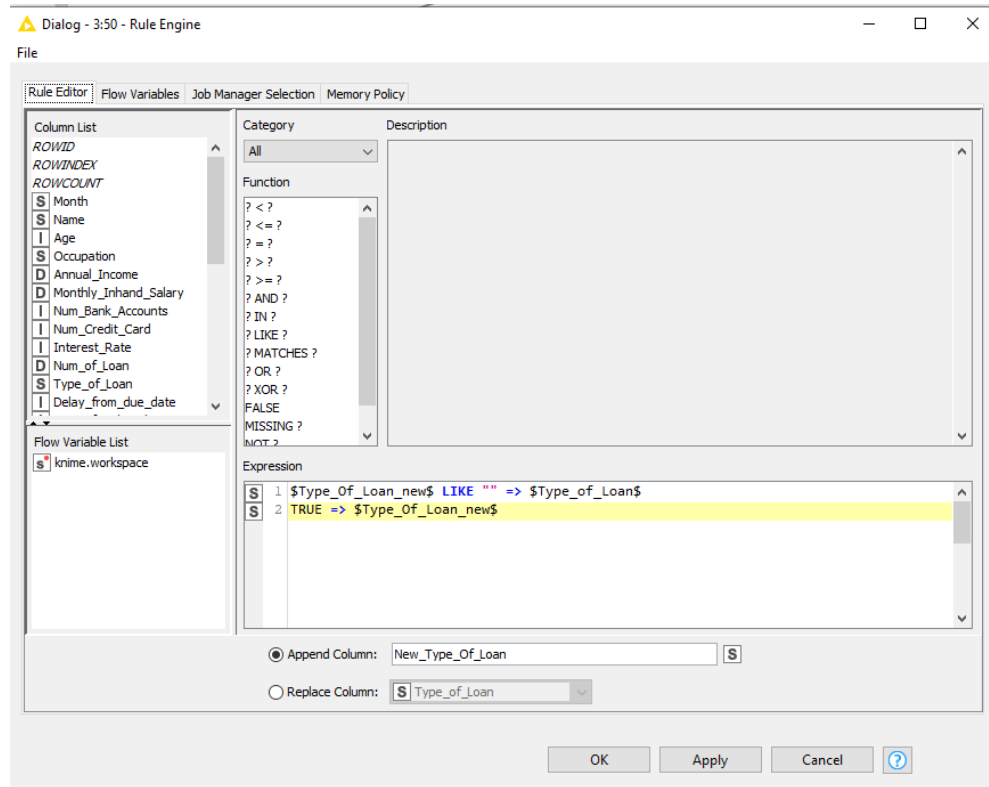


Figure 21: Simplified “Type_of_Loan”

- 10) In this part we will put “Changed_Credit_Limit” attribute into six categories using “Numeric Binner” node.

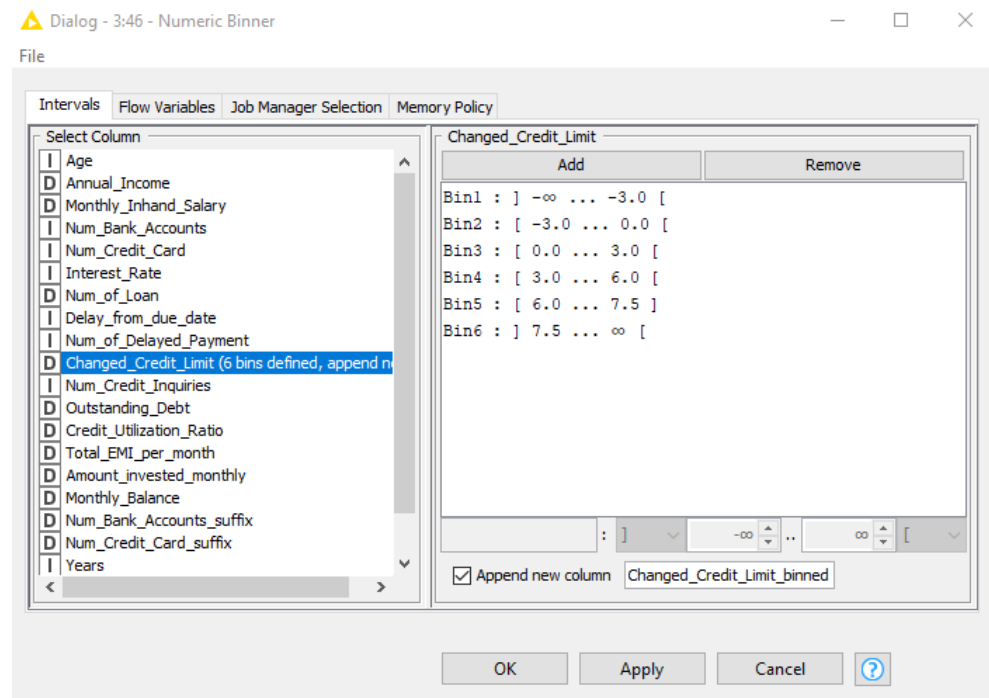


Figure 22: Categorization of “Changed_Credit_Limit” attribute

- 11) As the final part of the data cleansing and preparation process, we will discard unwanted value and partition our dataset for modelling.

Firstly, we will discard unwanted value such as placeholder variables and irrelevant attributes.

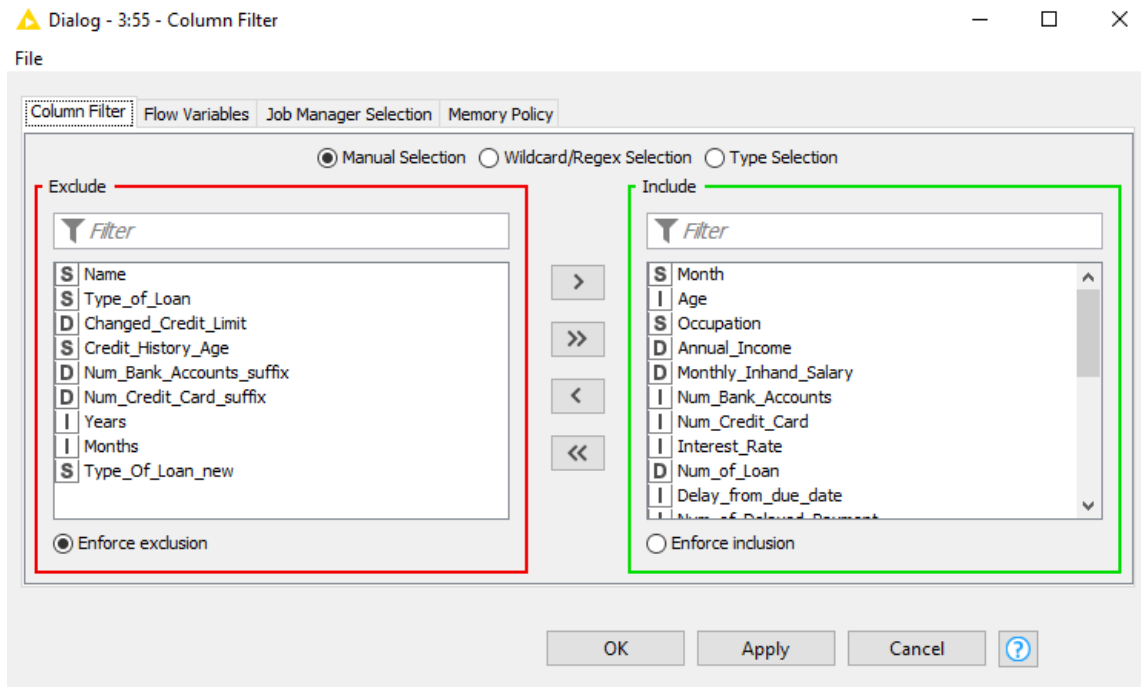


Figure 23: Elimination of undesired attributes.

Then we will select attributes for modelling with the requested seed.

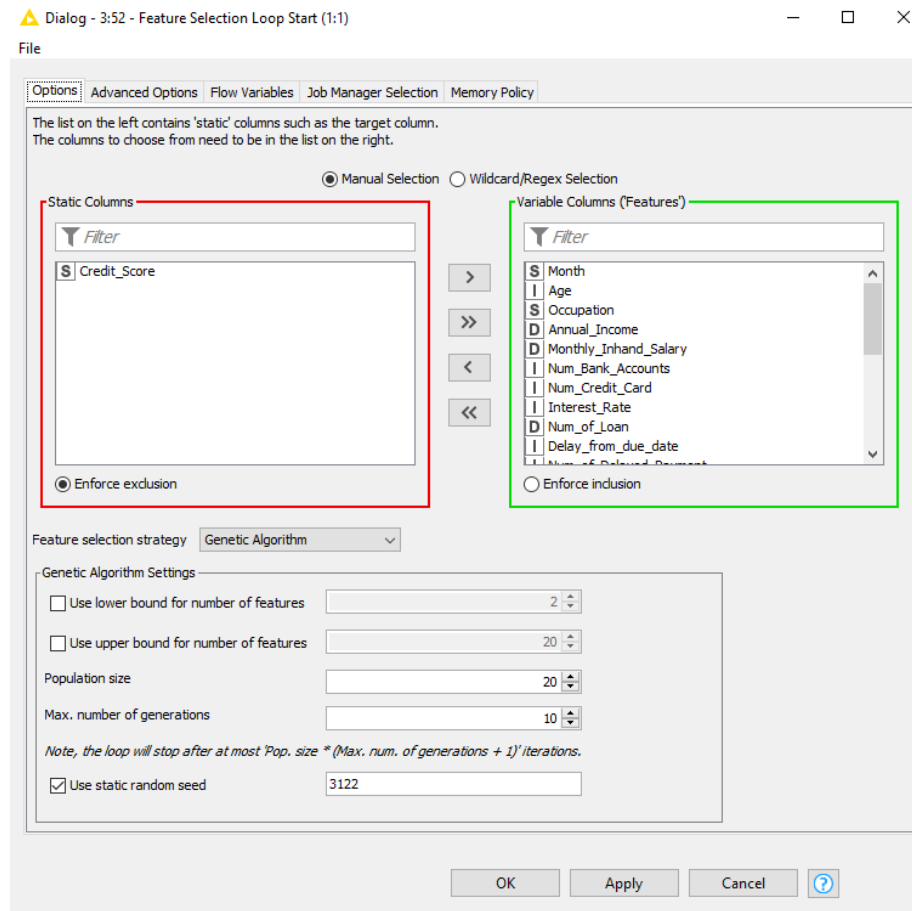


Figure 24: Static column selection

As the final step of our preparation process, we will shuffle and partition our data set with “3122” seed.

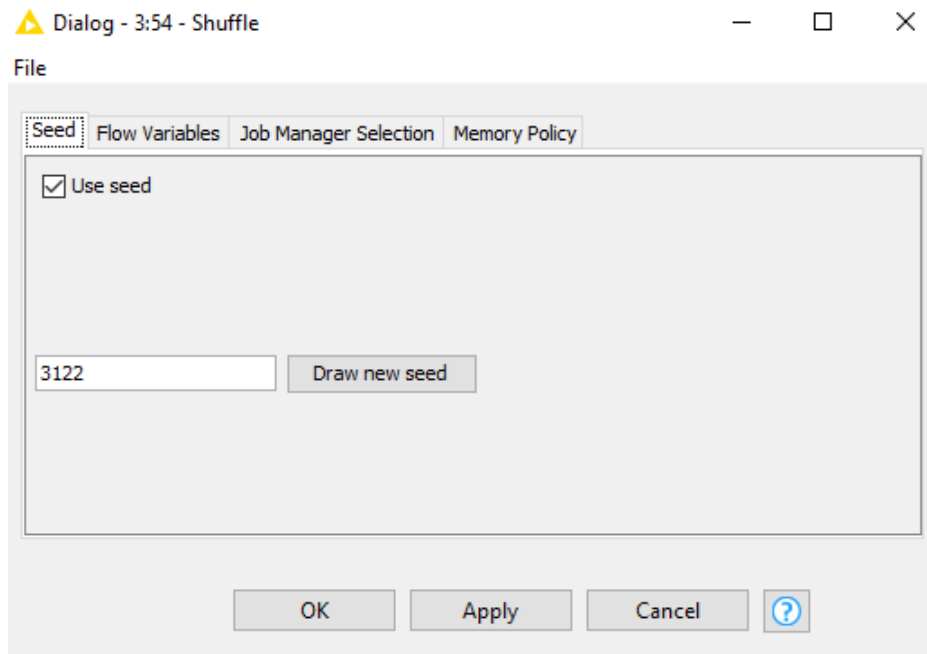


Figure 25: Shuffle the data set.

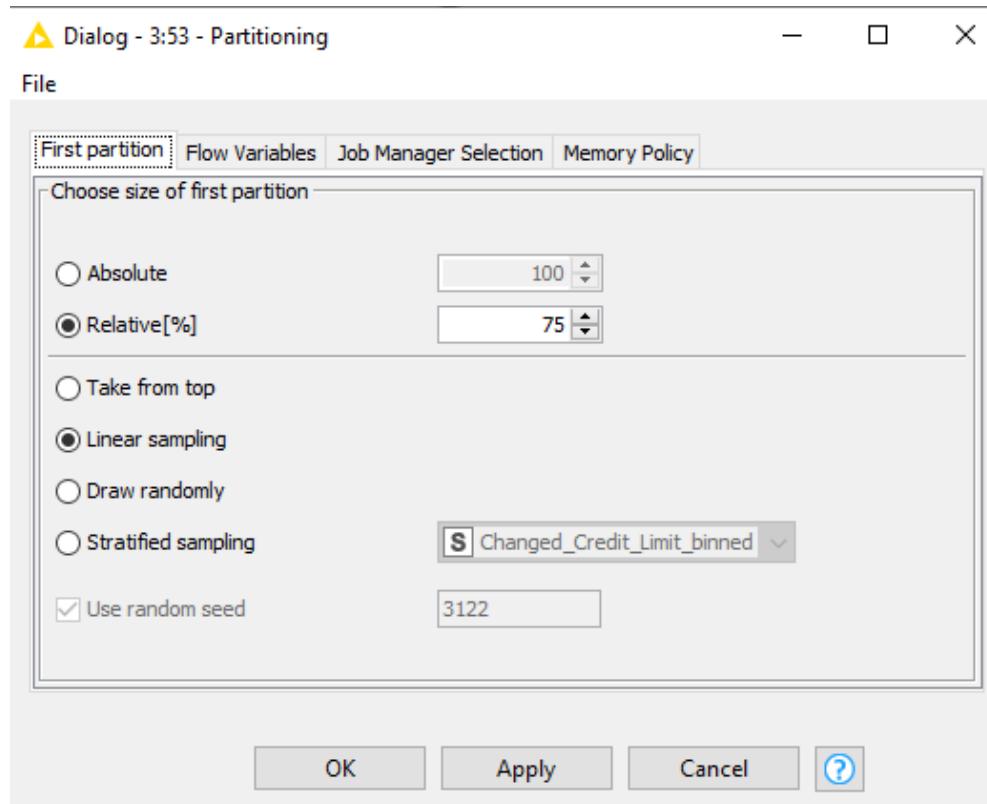


Figure 26: Partition the data set.

3. Naïve Bayes Model

1) The workflow for this model

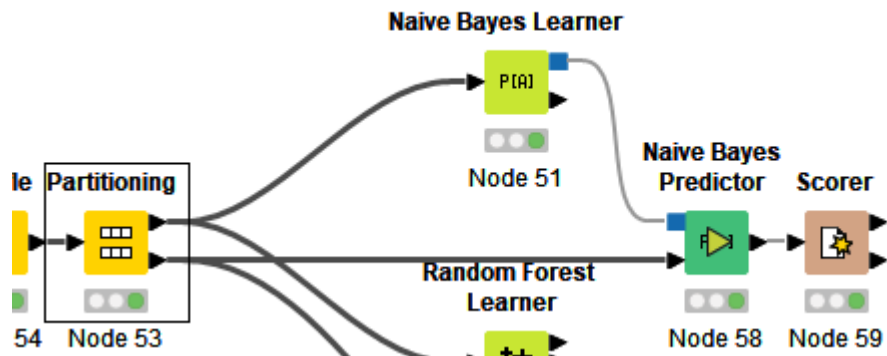


Figure 27: Naïve Bayes Model

2) The configuration for Naïve Bayes Model are as follows:

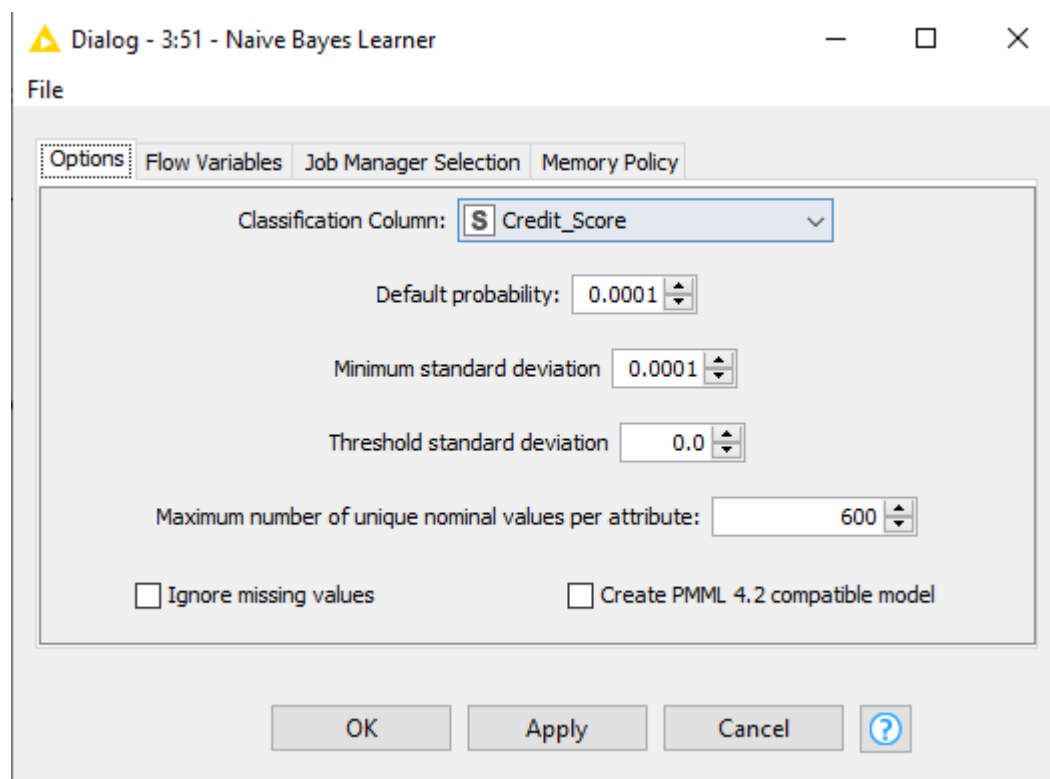


Figure 28: Configuration for Naïve Bayes Model

3) The result of the model are as follows:

Confusion matrix - 3:59 - Scorer

File Edit Hilite Navigation View

Table "spec_name" - Rows: 3 Spec - Columns: 3 Properties Flow V

Row ID	Good	Standard	Poor
Good	3376	620	107
Standard	3067	6653	2363
Poor	1034	1636	3876

Figure 29: Confusion matrix of the model

Row ID	TruePo...	FalsePo...	TrueNe...	FalseN...	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen'...
Good	3376	4101	14528	727	0.823	0.452	0.823	0.78	0.583	?	?
Standard	6653	2256	8393	5430	0.551	0.747	0.551	0.788	0.634	?	?
Poor	3876	2470	13716	2670	0.592	0.611	0.592	0.847	0.601	?	?
Overall	?	?	?	?	?	?	?	?	?	0.612	0.404

Figure 30: Accuracy statistics of the model

Banks can rely on the percentage of “Good” categorized borrowers to lower the risk of lending money to customers who have insufficient ability to repay the debt. However, according to the statistics the precision of identifying the “Good” borrower is relatively low at 0.452. Therefore, the classifier does not perform satisfactorily.

4)

In this context, the best metric to evaluate is the precision of “Good” borrowers as it would reduce the risk of banks lending money to borrowers who might default on their payments.

4. Random Forest Classifier

1) The workflow for the model

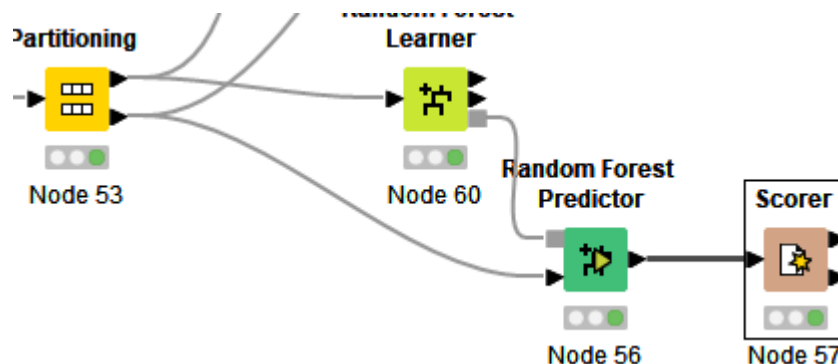


Figure 31: Random Forest Classifier Model

2) Random Forest Model Result

Table "spec_name" - Rows: 3 Spec - Columns: 3 Properties Flow Variables			
Row ID	Good	Standard	Poor
Good	2841	1178	84
Standard	1077	9481	1525
Poor	229	1534	4783

Figure 32: Confusion matrix for the model

Row ID	TruePo...	FalsePo...	TrueNe...	FalseN...	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen'...
Good	2841	1306	17323	1262	0.692	0.685	0.692	0.93	0.689	?	?
Standard	9481	2712	7937	2602	0.785	0.778	0.785	0.745	0.781	?	?
Poor	4783	1609	14577	1763	0.731	0.748	0.731	0.901	0.739	?	?
Overall	?	?	?	?	?	?	?	?	?	0.752	0.588

Figure 33: Accuracy statistics for the model

3) Comparison between Naïve Bayes Model and Random Forest Model

	Naïve Bayes Model	Random Forest Model
Accuracy	0.612	0.752
Precision of "Good" borrowers	0.452	0.685

As the major target of evaluated metric is the precision of "Good" borrowers, the Random Forest Model is more suitable as a predictive model for reducing risk of lending money for banks than Naïve Bayes Model.

- 4) In order to identify which class does the random forest model perform the best, let's take a look at the Recall rate (the rate of which the banks did not lend money to borrowers who can pay back the loan) and the Precision rate (the rate of which the banks lend money to borrowers whose credit are bad). Therefore, these two metrics will be the guidance for us to identify the best-performed class.

	Recall rate	Precision
Good	0.692	0.685
Standard	0.785	0.785
Poor	0.731	0.748

As the statistic suggest, "Standard" is the class in which the model performs the best as the Recall rate and the Precision rate are relatively high, both with the prediction of spotting the rate of which banks missed the chance and the rate of which the banks lend money to borrowers whose credit is bad.