2.1 Data Preparation

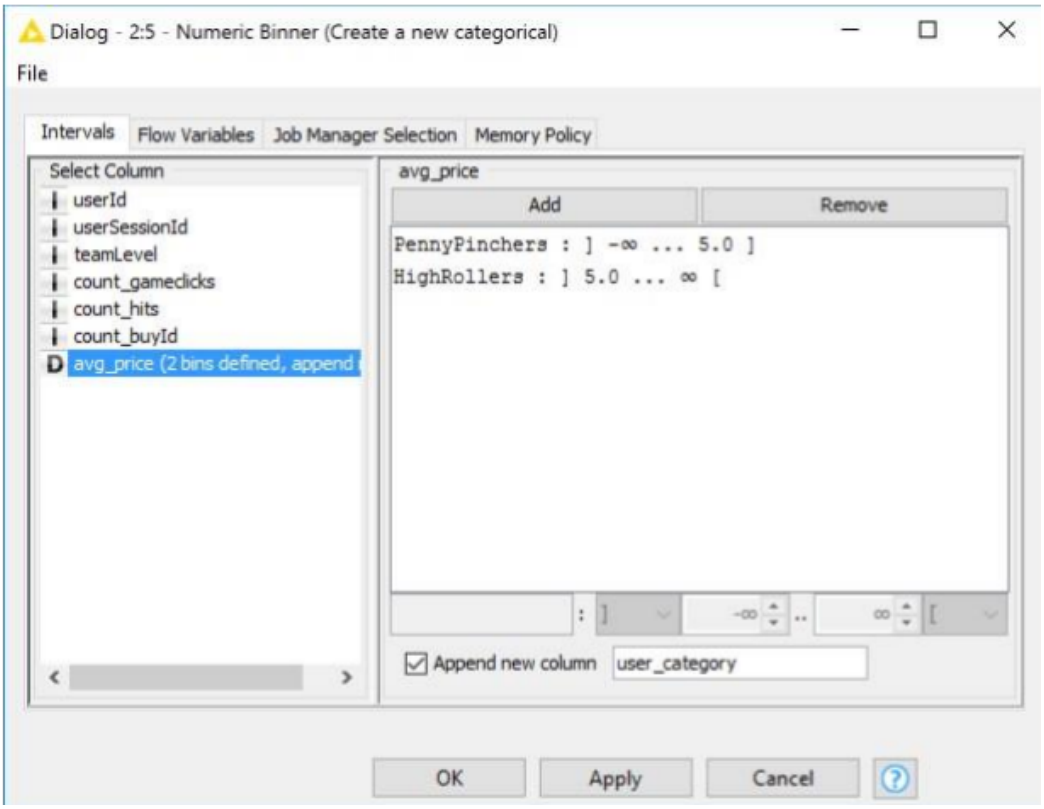
Analysis of combined_data.csv to investigate the classification of users.

2.1.1 Sample Selection

Item	Amount
# of Samples	4619
# of Samples with Purchases	1411

2.1.2 Attribute Creation

A new categorial attribute was created to enable analysis of players as broken into 2 categories (HighRollers and PennyPinchers). A screenshot of the attribute follows:



A new categorical attribute, named “user_category”, is created by the Numeric Binner node. As presented in the instruction, we need to define two categories for price which we will use to distinguish between HighRollers(buyers of items that cost more than $5.00) and PennyPinchers (buyers of items that cost $5.00 or less), so as we see in the screenshot above, the user who costs $5.00 or less is defined as “PennyPinchers”, the user who costs more than $5.00 is defined as “HighRollers”.

The creation of this new categorical attribute was necessary because it can facilitate the classification of users and contribute to the following steps.

2.1.3 Attribute Selection

The following attributes were filtered from the dataset for the following reasons:

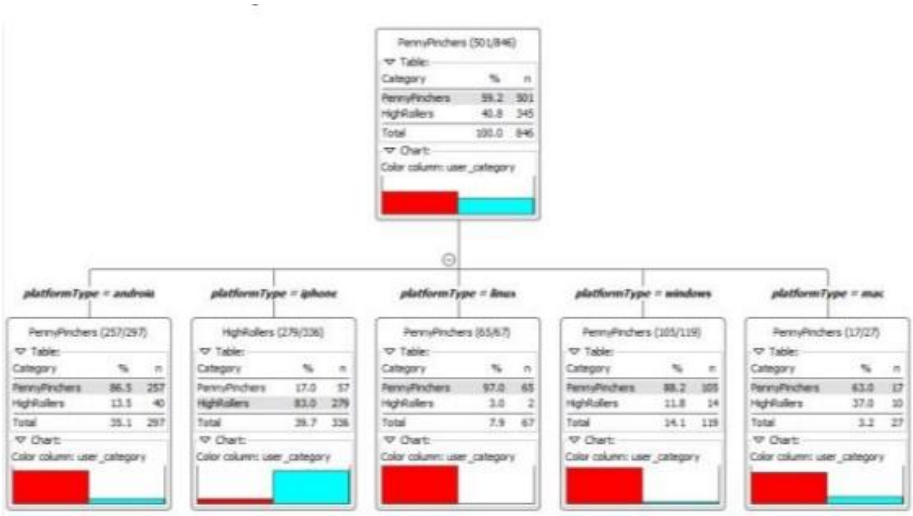
Attribute	Rationale for Filtering
userId	Since the objective is to predict which user is likely to purchase big-ticket items, and the attribute “userId” has no effect on it, so it’s removed.
userSessionId	Since the objective is to predict which user is likely to purchase big-ticket items, and the attribute “userSessionId” has no effect on it, so it’s removed.
avg_price	Since a new attribute “user_category” has been created, which was generated from the attribute “avg_price”, so we can remove it.

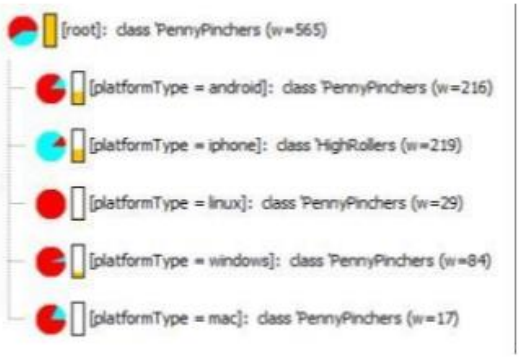
2.2 Data Partitioning and Modeling

The data was partitioned into train and test datasets. The training data set was used to create the decision tree model. The trained model was then applied to the test dataset. This is important because train data set is used in creating the decision tree model, the apply the model to the test data set, which is not used to train the mode then we can see the accuracy of the model.

When partitioning the data using sampling, it is important to set the random seed because it can get the same data partitions every time the node is executed.

A screenshot of the resulting decision tree can be seen below:





Step 3: Evaluation

A screenshot of the confusion matrix can be seen below:

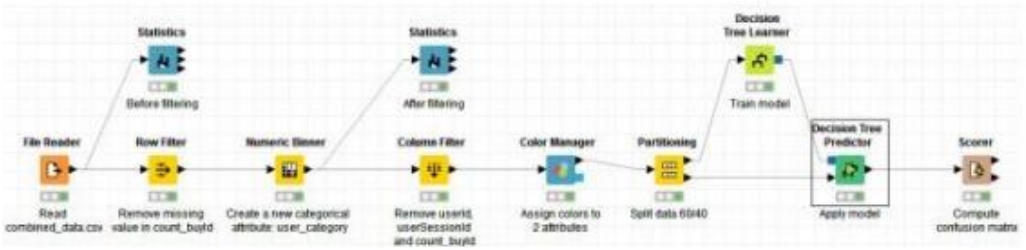
As seen in the screenshot above, the overall accuracy of the model is 88.496%.

• “308” & “38”: according to the model, we predict that 348(308+38) users are PennyPinchers, but among them 308 users are truly predicted, which means among these 348 users, 308 users are exactly PennyPinchers, 38 HighRollers are incorrectly predicted as PennyPinchers.

• “192” & “27”: according to the model, we predict that 219(192+27) users are HighRollers, but among them 192 users are truly predicted, which means among these 219 users, 192 users are exactly HighRollers, 27 PennyPinchers are incorrectly predicted as HighRollers.

4. Analysis Conclusions

The final KNIME workflow is shown below:



According to the resulting decision tree, it obviously shows that the predicted user_category is different in various platforms, the users on the platform android, linux, windows and mac are almost PennyPincher, however, most users which on the platform iphone are HighRoller.

Specific Recommendations to Increase Revenue
1. Offer more products to iPhone users.
2. Offer some promotions to PennyPinchers for attracting their consommation.