Design Report

COMP 1942 Project Phase 2

- Authors: Group 69, xxsuoaa, yslaiaf

Preprocessing

First, we format both the training data and the test data as a table using the Excel function "Format as Table".

1. Select the entire training data table, including the headers.
2. Find and press "Format as Table", then select any colors.
3. Check "My table contains headers" if it is not checked initially, then click "OK".
4. Do the same for the entire test data table.

Next, one of the discrete variable, `native-country`, has more than 30 distinct values. In particular, the training table has 41 distinct values while the test table has 40 distinct values. This is too many categories for our version of XLMiner to process.

From here on, do not touch the test table.

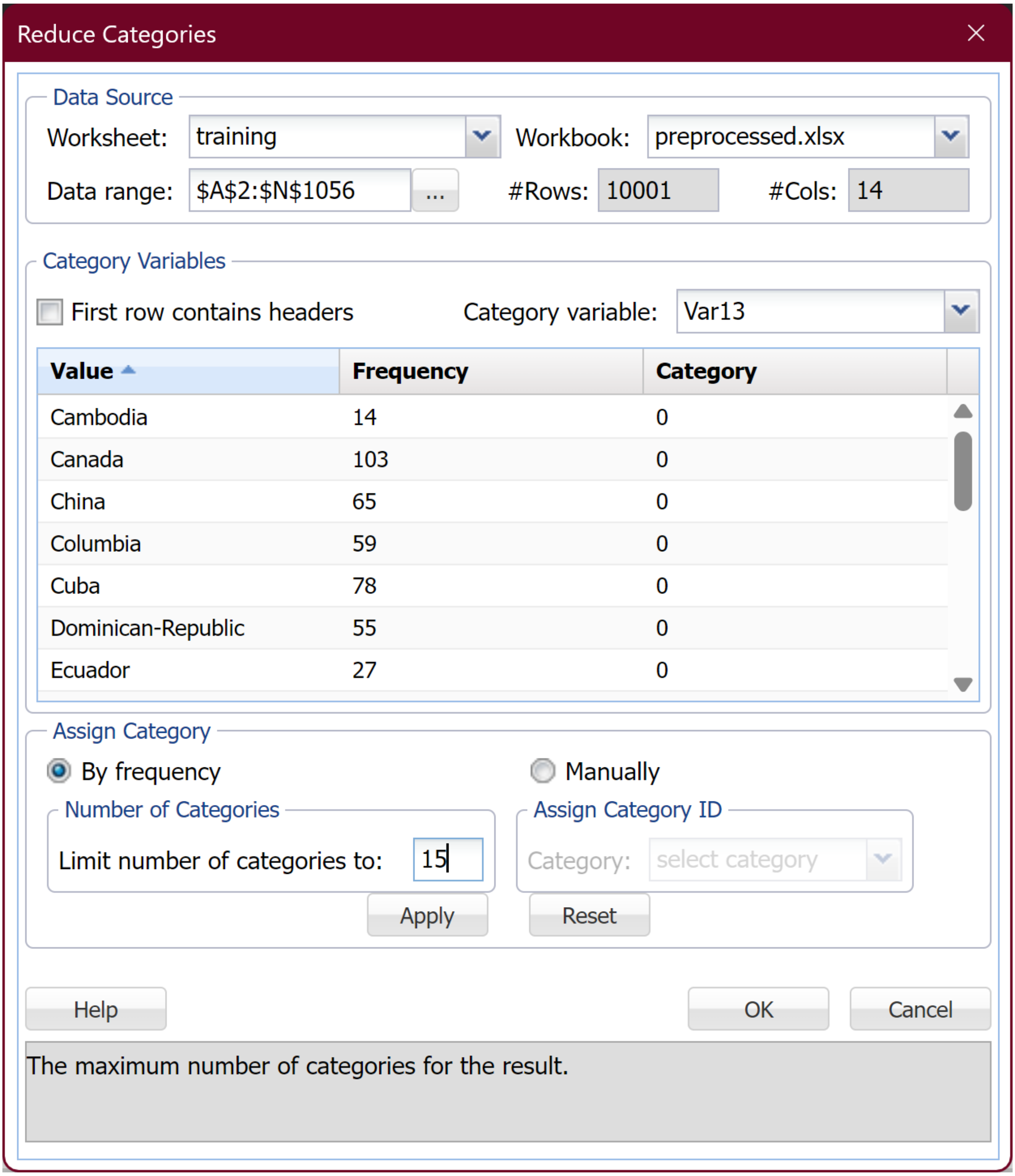
Before we reduce the number of categories, we need to sort the training table by `native-country` in ascending order. This is so that the frequency of the distinct values are somewhat randomized. We also need to record the original distinct values:

1. Create a new sheet named `native-country`.
2. Set value in `A1` as `key`, set value in `B1` as `value`.
3. Set formula in `A2` as `=UNIQUE(training!$M$2:$M$10001)`.
4. Copy `A2:A42`, and then paste (hover over "paste special" and click "values") at the same range. Afterwards, the range should not have any formulas.
5. Format `A1:B42` as a table using "Format as Table".

We need to use the "Reduce Categories" function of XLMiner. However, this function is also limited to 30 distinct values, so instead we need to do 2 passes for the training table. For each pass, perform the following steps:

1. Press on "Reduce Categories".

2. Configure the settings as in Figure 1. In particular, the settings needs to be changed are:
    - Data Range: `A2:N1056` for the 1st pass, `A1057:N10001` for the 2nd pass
    - First row contains headers: `false`
    - Category Variable: Ignore the option values. Choose the 13th option in the dropdown.
    - Limit number of categories to: `15`
    - Others: Check that the number of distinct values in the "Category Variables" table is 21 for the 1st pass and 20 for the 2nd pass.
3. Press on "Apply", and then "OK".
4. A summary sheet will be generated. Copy and paste (paste options: values) the transformed `native-country` column back into the `training` sheet at `O(start row):O(end row)`. Do not copy the meaningless headers.
5. For the 2nd pass of the training table, we additionally need to increment each `native-country` value by 15. This makes the processed `native-country` values unique from that of the 1st pass. This can be done by:
    1. Set `P(start row)` to `=$O(start row)+15`.
    2. Extend `P(start row)` to `P(start row):P(end row)`.
    3. Copy `P(start row):P(end row)`.
    4. Paste (paste options: values) at `O(start row):O(end row)`.
    5. Clear `P(start row):P(end row)`.



Reduce Categories

Data Source

Worksheet: training

Workbook: preprocessed.xlsx

Data range: $A$2:$N$1056

#Rows: 10001

#Cols: 14

Category Variables

□ First row contains headers

Category variable: Var13

Value	Frequency	Category
Cambodia	14	0
Canada	103	0
China	65	0
Columbia	59	0
Cuba	78	0
Dominican-Republic	55	0
Ecuador	27	0

Assign Category

● By frequency

○ Manually

Number of Categories

Limit number of categories to: 15

Assign Category ID

Category: select category

The maximum number of categories for the result.

Figure 1

After that, we need to fill in the `native-country` string-to-number mappings in the `native-country` sheet.training table, fill in the mappings. We can do so easily:

1. Set `B2` to `=VLOOKUP($A2,training!$M$2:$O$10001,3,TRUE)`.
2. Extend `B2` to `B2:B42`.
3. Copy `B2:B42`, and then paste (paste options: values) at the same range. Afterwards, the range should not have any formulas.

For reference, our mapping is:

key	**value**
Cambodia	15
Canada	3
China	7
Columbia	9
Cuba	5
Dominican-Republic	10
Ecuador	14
El-Salvador	4
England	6
France	15
Germany	1
Greece	13
Guatemala	8
Haiti	11
Holand-Netherlands	15
Honduras	15
Hong	15
Hungary	15
India	2
Iran	12
Ireland	15
Italy	22
Jamaica	21
Japan	23
Laos	30
Mexico	17
Nicaragua	29
Outlying-US(Guam-USVI-etc)	30
Peru	28
Philippines	18
Poland	25
Portugal	26
Puerto-Rico	19
Scotland	30

key	value
South	20
Taiwan	27
Thailand	30
Trinadad&Tobago	30
United-States	16
Vietnam	24
Yugoslavia	30

Afterwards, in the `training` sheet, cut and paste `O2:O10001` to `M2:M10001`.

Now we can finally touch the test table. Map the `native-country` column of the test table. To do so:

1. Set `N2` to `=VLOOKUP($M2,'native-country'!$A$2:$B$42,2,TRUE)`.
2. Extend `N2` to `N2:N8001`.
3. Copy `N2:N8001`, and then paste (paste options: values) at `M2:M8001`.
4. Clear `N2:N8001`.

Finally, sort both tables by all columns, left columns first. This can be done by starting with the rightmost column and sort by ascending. Then, go to the column on the immediate left and sort by ascending. Repeat this until the leftmost column is reached and sorted by ascending.

The above is not strictly necessarily. It is only for standardizing the results. For reference, the resulting workbook is available as `preprocessed.xlsx`.

Models

We will be testing 5 models. All models below use `preprocessed.xlsx` as the source workbook.

Model 1: *k*-Nearest Neighbors

Note that we only use continuous variables here. This is because *k*-nearest neighbors uses distance for classification, and distance cannot be meaningfully defined for discrete variables.

Press on "Data Science > Classify > *k*-Nearest Neighbors". Then, configure the model as follows:

Data	
Workbook	model1.xlsx
Worksheet	training
Partitioning Method	Random Partition
Seed Value	12345
# Records in the training data	6000
# Records in the validation data	4000

Variables	
# Variables	6
Scale Variables	age, education-num, capital-gain, capital-loss, hours-per-week
Output Variable	income

Rescaling: Fitting Parameters	
Rescale Data?	TRUE
Technique	STANDARDIZATION

Nearest-Neighbors: Fitting Parameters	
# Nearest neighbors (K)	10

Nearest-Neighbors Classification: Fitting Parameters	
Prior Probability Calculation	EMPIRICAL

Nearest-Neighbors Classification: Model Parameters	
# Classes	2
Success Class	>50K
Success Probability	0.5

Nearest-Neighbors: Reporting Parameters	
Search for best K?	FALSE

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

Model 2: Classification Tree

Before we can use the classification tree model, note that the model in XLMiner requires distinct variables to have 15 or fewer distinct values. Unfortunately, the `native-country` column has 30 distinct values. To fix this, we need to use the "Reduce Categories" function of XLMiner:

1. Press on "Reduce Categories".
2. Configure the settings as in Figure 1. In particular, the settings needs to be changed are:
    - Data Range: `A1:N10001`
    - First row contains headers: `true`
    - Category Variable: `native-country`
    - Limit number of categories to: `15`
3. Press on "Apply", and then "OK".
4. A summary sheet will be generated. Copy and paste (paste options: values) the transformed table back into the `training` sheet at `A2:N10001`. Do not copy the headers.

After doing so, we can finally use the classification tree model.

Note that we do not use the `education` column because `education-num` is the continuous version of `education`, so we only need to choose one of them.

Press on "Data Science > Classify > Classification Tree". Then, configure the model as follows:

Data	
Workbook	model2.xlsx
Worksheet	training
Partitioning Method	Random Partition
Seed Value	12345
# Records in the training data	6000
# Records in the validation data	4000

Variables	
# Variables	12
Scale Variables	age, education-num, capital-gain, capital-loss, hours-per-week
Categorical Variables	workclass, marital-status, occupation, relationship, race, sex, native-country

Variables	
Output Variable	income

Rescaling: Fitting Parameters	
Rescale Data?	TRUE
Technique	STANDARDIZATION

Decision Tree Classification: Fitting Parameters	
Prior Probability Calculation	EMPIRICAL

Decision Tree: Model Parameters	
Prune?	TRUE
Scoring tree type	Best pruned

Decision Tree Classification: Model Parameters	
# Classes	2
Success Class	>50K
Success Probability	0.5

Decision Tree: Reporting Parameters	
Trees to draw	Fully grown, Best pruned, Min error
# Max level to display	7
Show feature importance?	TRUE

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

Model 3: Naive Bayes

Note that we do not use the `education` column because `education-num` is the continuous version of `education`, so we only need to choose one of them.

Also note that we do not partition the training-validation data into training data and validation data. This is because naive bayes classifiers require each distinct value to appear at least once in the training data, and the entire training-validation data is not large enough to ensure all possible distinct values appear in the training data at least once.

Press on "Data Science > Classify > Naive Bayes". Then, configure the model as follows:

Data	
Workbook	model3.xlsx
Worksheet	training
Data Range	*A*$1 :N$10001
# Records	10000

Variables	
# Variables	12
Scale Variables	age, workclass, education-num, martial-status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week, native-country
Output Variable	income

Naive Bayes: Fitting Parameters	
Laplace smoothing	TRUE
Smoothing alpha	1
Prior Probability Calculation	EMPIRICAL

Naive Bayes: Model Parameters	
# Classes	2
Success Classes	>50K
Success Probability	0.5

Naive Bayes: Reporting Parameters	
Show prior conditional probability	TRUE
Show log-density	TRUE

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data

Output Options
Frequency chart on training data

Model 4: Neural Network

Note that we do not use the `education` column because `education-num` is the continuous version of `education`, so we only need to choose one of them.

Press on "Data Science > Classify > Neural Network > Manual Network". Then, configure the model as follows:

Data	
Workbook	model4.xlsx
Worksheet	training
Partitioning Method	Random Partition
Seed Value	12345
# Records in the training data	6000
# Records in the validation data	4000

Variables	
# Variables	12
Scale Variables	age, education-num, capital-gain, capital-loss, hours-per-week
Categorical Variables	workclass, martial-status, relationship, race, sex, native-country
Output Variable	income

Rescaling: Fitting Parameters	
Rescale Data?	TRUE
Technique	STANDARDIZATION

Neural Network: Fitting Parameters	
Random seed for initial weights	12345
# Hidden Layers	0
Learning rate	0.1
Weight change momentum	0.6
Error tolerance	0.01
Weight decay	0
Cost function	Cross Entropy

Neural Network: Fitting Parameters	
Hidden layer activation function	ReLU
Output layer activation function	SOFTMAX
Learning order	Random
Learning order: random seed	12345
Response correction	0.01
Data for error computation	TRAINING ONLY
Maximum number of epochs	1000
Maximum number of epochs without improvement	5
Maximum training time	3600
Minimum relative change in error	0.0001
Minimum relative change in error compared to null model	0.001

Neural Network Classification: Fitting Parameters	
Prior Probability Calculation	EMPIRICAL

Neural Network Classification: Model Parameters	
# Classes	2
Success Class	>50K
Success Probability	0.5

Neural Network: Reporting Parameters	
Search for best architecture	FALSE
Show neural network weights?	TRUE

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

Model 5: Neural Network

The difference between this model and model 4 is that this model has an additional hidden layer of 64 neurons. We want to see if the hidden layer can improve the accuracy of the model.

Note that we do not use the `education` column because `education-num` is the continuous version of `education`, so we only need to choose one of them.

Press on "Data Science > Classify > Neural Network > Manual Network". Then, configure the model as follows:

Data	
Workbook	model5.xlsx
Worksheet	training
Partitioning Method	Random Partition
Seed Value	12345
# Records in the training data	6000
# Records in the validation data	4000

Variables	
# Variables	12
Scale Variables	age, education-num, capital-gain, capital-loss, hours-per-week
Categorical Variables	workclass, martial-status, relationship, race, sex, native-country
Output Variable	income

Rescaling: Fitting Parameters	
Rescale Data?	TRUE
Technique	STANDARDIZATION

Neural Network: Fitting Parameters	
Random seed for initial weights	12345
# Hidden Layers	1
# Nodes in Hidden Layer 1	64
Learning rate	0.1
Weight change momentum	0.6
Error tolerance	0.01
Weight decay	0
Cost function	Cross Entropy

Neural Network: Fitting Parameters	
Hidden layer activation function	ReLU
Output layer activation function	SOFTMAX
Learning order	Random
Learning order: random seed	12345
Response correction	0.01
Data for error computation	TRAINING ONLY
Maximum number of epochs	1000
Maximum number of epochs without improvement	5
Maximum training time	3600
Minimum relative change in error	0.0001
Minimum relative change in error compared to null model	0.001

Neural Network Classification: Fitting Parameters	
Prior Probability Calculation	EMPIRICAL

Neural Network Classification: Model Parameters	
# Classes	2
Success Class	>50K
Success Probability	0.5

Neural Network: Reporting Parameters	
Search for best architecture	FALSE
Show neural network weights?	TRUE

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data