Data Mining III – Lesson 2

Tamara B. Sipes, Ph.D.

Lesson 2 Overview

- Mining of the CARS dataset, beginning to end:
  - Problem definition
  - Data Preparation
  - Data Mining
  - Evaluation
  - Presentation

Instructions

- Hands-on lessons
- Be prepared to have both the Lesson window opened and the weka window as well
- Have a notepad ready

- Let's get started!

Dataset

- CARS1.csv can be found under the Class Resources
- TO DO:
  - Open weka 3.5.7 or similar
  - Choose Explorer from the Applications menu
  - Read in CARS1.csv (yes, .csv just like .arff is a valid input format)

Novice's Effort

- "But, I have the tool!!"
- The file is read in, I will just run it…
  - Go to Classify tab, and choose a classifier
  - Click on Choose/Tree/RepTree method
  - Results: 75.00 %
  - Not bad. Kind of.
  - I solved this!

RepTree "Default Run" Model

Model = chevrolet_chevelle malibu : US

Model = buick_skylark 320 : US

Model = plymouth_satellite : US

Model = amc_rebel sst : US

Model = ford_torino : USA

Model = ford_galaxie 500 : US

Model = chevrolet_impala : US

Model = plymouth_fury iii : US

Model = pontiac_catalina : US

Model = amc_ambassador dpl : USA

Model = dodge_challenger se : USA

…

RepTree "Default Run" Output

Correctly Classified Instances	303	75	%
Incorrectly Classified Instances	101	25	%
Kappa statistic	0.4329		
Mean absolute error	0.1778		
Root mean squared error	0.3023		
Relative absolute error	64.3739 %		
Root relative squared error	81.5139 %		
Total Number of Instances	404		

RepTree "Default Run" Output

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.996	0.649	0.713	0.996	0.831	0.785	US
0	0.003	0	0	0	0.516	USA
0.38	0	1	0.38	0.55	0.813	Japan
0.353	0	1	0.353	0.522	0.781	Europe

=== Confusion Matrix ===

a	b	c	d		<-- classified as
249	1	0	0	|	a = US
7	0	0	0	|	b = USA
49	0	30	0	|	c = Japan
44	0	0	24	|	d = Europe

Oopps!

- Predicting US and USA…
- Count of: US 250, USA 7
- Let's correct that!
- So, the Novice opens the file in Excel, and corrects the 7 USA entries by hand (we will see how to do that in weka)
- And, reruns the RepTree model

RepTree "Default Run 2" Model

Brand = chevrolet : US

Brand = buick : US

Brand = plymouth : US

Brand = amc : US

Brand = ford : US

Brand = pontiac : US

Brand = dodge : US

Brand = toyota : Japan

Brand = datsun : Japan

Brand = vw : Europe

Brand = peugeot : Europe

…

RepTree “Default Run 2” Output

Correctly Classified Instances	402	99.505 %
Incorrectly Classified Instances	2	0.495 %
Kappa statistic	0.9906	
Mean absolute error	0.0039	
Root mean squared error	0.0464	
Relative absolute error	1.1053 %	
Root relative squared error	11.0513 %	
Total Number of Instances	404	

RepTree "Default Run 2" Output

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
1	0.014	0.992	1	0.996	1	US
0.987	0	1	0.987	0.994	1	Japan
0.985	0	1	0.985	0.993	1	Europe

=== Confusion Matrix ===

```
  a   b   c   <-- classified as
257   0   0 |   a = US
  1  78   0 |   b = Japan
  1   0  67 |   c = Europe
```

Novice’s Conclusion

- I solved this data mining problem!!
- My scores are as high as 99.5% correctly classified instances
- Can’t beat that!

The Meaning of the Model

- What is the meaning of this model?
- Is this what we need?
- The scores are great, but…

- Let’s do this properly!

The Correct Way: First Impressions

- Let's analyze the Preprocess tab
- 404 instances
- 10 input variables:
  - Cyl
  - CuPerIn
  - Hpwr
  - Wt_Lbs
  - Acc_0-60
  - Year
  - Brand
  - Model
  - MiPerGal
  - Origin (default class variable, as it is the last one)

The Beginning: Problem Definition

- First things first:
  - What is the problem definition?
  - How do we find out?
  - What are we trying to accomplish?
  - Let's find out.

Problem Definition

- Talking to the client (or, problem owner), we find out:
  - The company needed a model of the different cars' consumption (mileage per gallon information). In other words, we really need to predict MiPerGal.
  - (the Novice never took the time to figure out what was really needed! and was predicting on the wrong variable)

Next step

- Prepare the data with respect to the problem definition
- First, just for comparison reasons, let's run the dataset through without any data preparation!

Predicting Miles Per Gallon

- Under the Classify tab, choose "MiPerGal" to predict on, instead of "Origin"
- Run RepTree method again
- Result:

Correctly Classified Instances	35	8.6634 %
Incorrectly Classified Instances	369	91.3366 %
Kappa statistic	0.0397	
Mean absolute error	0.0146	
Root mean squared error	0.0919	
Relative absolute error	94.9165 %	
Root relative squared error	104.9749 %	
Total Number of Instances	404	

Data Preparation of CARS1.csv

- Let’s make MiPerGal our class variable first:
- Preprocess tab/Edit
- Right-click on MiPerGal and choose “Attribute as class”
- Click OK at the bottom
- Return to Preprocess tab and double-check your attribute manipulation: MiPerGal should be listed as the last attribute

Individual variables

- The next step is to check the individual variables
- We want to check for missing values, inconsistencies, duplicates etc.
- Let's start with the Origin as we know there is an inconsistency in labeling US and USA
- Let's merge US and USA values

Merging Nominal Values

- Preprocess/Filter/Choose/ (gives the Filter Tree)
  - Choose: filters/unsupervised/attribute/ MergeTwoValues
  - Click on “MergeTwoValues” to set the parameters:
    - attributeIndex: 9
    - fistValueIndex: 1
    - secondValueIndex: 2
    - Click “Apply”!
  - Calls: MergeTwoValues –C 9 –F 1 –S 2
  - Check the statistics window for Origin variable (by clicking on it)

Brand: Inconsistencies

- chevy vs. chevrolet
- M-benz vs. mercedes
- Let's merge these values:
  - MergeTwoValues –C 7 –F 1 –S 2 (merges chevy and chevrolet) – renames it "chevrolet_chevy" (check)
  - MergeTwoValues –C 7 –F 1 –S 2 (merges M-benz and mercedes) – "m-benz_mercedes" (check to make sure it was applied)

Brand: Incorrect values?

- Brand contains 2 values which are erroneous (Capri and Hi), and so those should be deleted:
  - Choose: filters/unsupervised/instance/RemoveWithValues
  - Click on "RemoveWithValues" to set the parameters:
    - attributeIndex: 7
    - invertSelection: False
    - matchMissingValues: False
    - modifyHeader: True
    - nominalIndices: 15, 26 (for "hi", and for "capri")
    - splitPoint: 0.0
    - Click "Apply"!
  - Calls: RemoveWithValues –S 0.0 –C 7 –L 15,26 -H
  - Check to make sure that Brand now has 28 distinct values

Cyl: Inconsistencies

- The Cyl variable contains values which might be inaccurate
- While 4, 6, and 8 are common values for cylinder counts, 3 and 5 are unusual
- A review of technical specifications via Google shows that the 4 cars with 3-cylinder listings were in fact rotary engines (without cylinders), while the 3 cars with 5-cylinder listings were valid
- This means that the four 3-cylinder entries do not correspond to real-world values, and in order to preserve the validity of the numeric data, these values should be deleted

Cyl: Delete Incorrect Values

- Choose: filters/unsupervised/instance/RemoveWithValues
- Click on "RemoveWithValues" to set the parameters:
  - attributeIndex: 1
  - invertSelection: False
  - matchMissingValues: False
  - modifyHeader: True
  - nominalIndices: 5
  - splitPoint: 0.0
  - Click "Apply"!
  - Check to see that there are only 5 distinct values in Cyl

MiPerGal: Missing Value?

- There are 126 distinct values, one of which is missing
- Delete the missing value, as it is the class variable:
  - Another way to delete (for a small number of deletions): Click on Edit in the Preprocess tab
  - Highlight the row
  - Right-click and choose “Delete selected instance”
  - Click on OK

MiPerGal: Nominal Value?

- Miles per gallon should really not be a nominal value
- There are now 125 distinct values
- Let's change is to numeric:
  - Normaly this is done by discretizing
  - Nevertheless, we want to keep all the values, just call them numeric:
    - Save the current file as .arff
    - Open it in WordPad or similar application
    - Change the line: @attribute MiPerGal {18.0,15.0,16.0,17.0,… to @attribute MiPerGal numeric

More Numeric Values

- Also, change these to numeric:
  - Cyl
  - CuPerIn
  - Hpwr
  - Wt_lbs
  - Acc_0-60
  - Year
- Also, change all the '?' into ?

Reopen the File

- Save and go back to weka
- Open cars1.arff
- Check to see that MiPerGal is a numeric variable now by clicking on it in the Preprocess tab and checking its statistics

Let's Run and Evaluate Again

- Run RepTree again
- Evaluation:

Correlation coefficient	0.4772
Mean absolute error	5.2765
Root mean squared error	6.8733
Relative absolute error	80.2084 %
Root relative squared error	88.0794 %
Total Number of Instances	397

Examine the Model

Model = chevrolet_chevelle malibu : 17.64
Model = buick_skylark 320 : 15
Model = plymouth_satellite : 18
Model = amc_rebel sst : 16
Model = ford_torino : 17
Model = ford_galaxie 500 : 14.33
Model = chevrolet_impala : 12.78
Model = plymouth_fury iii : 14.33
Model = pontiac_catalina : 14.5
Model = amc_ambassador dpl : 15

Model Examination Feedback

- Model not what we need as a predictor
- Car characteristics is what is preferred
- Delete the Model attribute from the training set:
  - Click to select to the left of Model
  - Click on Remove

Let's Run and Evaluate Again

- Run RepTree again
- Evaluation:

Correlation coefficient	0.8716
Mean absolute error	2.7382
Root mean squared error	3.8324
Relative absolute error	41.6237 %
Root relative squared error	49.1117 %
Total Number of Instances	397

Examine this Model

```
CuPerIn < 153
|   Hpwr < 70.5
|   |   Year < 1978.5
|   |   |   Brand = chevrolet_chevy : 28
|   |   |   Brand = buick : 29.63
|   |   |   Brand = plymouth : 26
|   |   |   Brand = amc : 29.63
|   |   |   Brand = ford : 29.63
|   |   |   Brand = pontiac : 29.63
|   |   |   Brand = dodge : 29.63
|   |   |   Brand = toyota : 31.33
|   |   |   Brand = datsun : 32.72
|   |   |   Brand = vw : 25.99
|   |   |   Brand = peugeot : 30
|   |   |   Brand = audi : 29.63
```

Model Examination Feedback

- Brand not what we need as a predictor
- Brand is most likely not very expressive
- Car characteristics is what is preferred
- Delete the Brand attribute from the training set:
  - Click to select to the left of Brand
  - Click on Remove

Let's Run and Evaluate Again

- Run RepTree again
- Evaluation:

Correlation coefficient	0.9003
Mean absolute error	2.4097
Root mean squared error	3.3916
Relative absolute error	36.6291 %
Root relative squared error	43.4628 %
Total Number of Instances	397

Examine this Model

```
CuPerIn < 153
|   Hpwr < 70.5
|   |   Year < 1978.5
|   |   |   Wt_Lbs < 1829.5 : 33.17
|   |   |   Wt_Lbs >= 1829.5
|   |   |   |   Year < 1973.5 : 26.2
|   |   |   |   Year >= 1973.5
|   |   |   |   |   Origin = US_USA : 28
|   |   |   |   |   Origin = Japan : 31.82
|   |   |   |   |   Origin = Europe : 28.24
|   |   Year >= 1978.5 : 36.05
|   Hpwr >= 70.5
|   |   Year < 1979.5
|   |   |   Wt_Lbs < 2212.5 : 28.26
|   |   |   Wt_Lbs >= 2212.5
```

Model Examination Feedback

- Similar values in the leaves:

```
| | | | | Origin = US_USA : 28
| | | | | Origin = Japan : 31.82
| | | | | Origin = Europe : 28.24
```

- Need to discretize
- Check with the business problem definition/ statement

Discretize

- Weka bug – version 3.5.7 still has the same bug - cannot discretize the very last variable
- To get around: set it as the second to last(by Edit/R-click on Origin/Set as class attribute)
- Then, discretize by: Filters/unsupervised/Discretize

Number of Bins?

- Bins = 4: 78.8413 %
- Bins = 6: 66.2469 %
- Bins = 10: 46.0957 %

- Bins = 3: 81.6121 %
- Bins = 3; Then use C4.5 (J48): 82.6196 %

Feedback

- Abort the discretization route
- Go back to numeric prediction
- Try another method
- Other numeric methods:
  - Model Trees
  - Regression Trees
  - ANN
  - Etc.

Model Tree

Correlation coefficient	0.9386
Mean absolute error	1.8921
Root mean squared error	2.6858
Relative absolute error	28.7617 %
Root relative squared error	34.4184 %
Total Number of Instances	397

Regression Tree

Correlation coefficient	0.9057
Mean absolute error	2.4781
Root mean squared error	3.4288
Relative absolute error	37.6695 %
Root relative squared error	43.9393 %
Total Number of Instances	397

Multilayer Perceptron

Correlation coefficient	0.9315
Mean absolute error	2.1877
Root mean squared error	2.8719
Relative absolute error	33.2558 %
Root relative squared error	36.8024 %
Total Number of Instances	397

Some More Investigation

- Other methods (SVM, SMO, Decision Stump etc.)
- Parameter tuning for well performing methods (such as useSmoothed and minNumberInstances in Model Tree)
- We pretty much converged to our “best” solution

Examine the Model

```
CuPerIn <= 190.5 :
|   Wt_Lbs <= 2217 : LM1
|   Wt_Lbs >  2217 :
|   |   Year <= 1979.5 : LM2
|   |   Year >  1979.5 : LM3
CuPerIn >  190.5 :
|   Hpwr <= 141 :
|   |   CuPerIn <= 241 : LM4
|   |   CuPerIn >  241 :
|   |   |   Year <= 1978.5 : LM5
|   |   |   Year >  1978.5 : LM6
```

Model, continued

```
|   Hpwr >  141 :
|   |   Wt_Lbs <= 4361.5 :
|   |   |   Year <= 1977.5 :
|   |   |   |   Wt_Lbs <= 3682.5 : LM7
|   |   |   |   Wt_Lbs >  3682.5 : LM8
|   |   |   Year >  1977.5 :
|   |   |   |   Wt_Lbs <= 3997 : LM9
|   |   |   |   Wt_Lbs >  3997 : LM10
|   |   Wt_Lbs >  4361.5 :
|   |   |   Year <= 1974.5 : LM11
|   |   |   Year >  1974.5 : LM12
```

The Leaves

LM num: 1

MiPerGal =

-0.0307 * Cyl

- 0.0863 * CuPerIn

- 0.0119 * Hpwr

- 0.0013 * Wt_Lbs

+ 0.2022 * Acc_0-60

+ 0.9936 * Year

+ 0.4035 * Origin=Europe,Japan

- 1925.1059 ...

Just One More Iteration

- Origin is most likely not that helpful
- Let's run it without Origin and evaluate:

Correlation coefficient	0.9331
Mean absolute error	2.0088
Root mean squared error	2.801
Relative absolute error	30.5358 %
Root relative squared error	35.8942 %
Total Number of Instances	397

Feedback

- Scores not that much different
- Model is simpler – only 10 leaves
- Model is more presentable
- Keep it!

Final Model

```
CuPerIn <= 190.5 : LM1
CuPerIn >  190.5 :
|   Hpwr <= 141 :
|   |   CuPerIn <= 241 : LM2
|   |   CuPerIn >  241 :
|   |   |   Year <= 1978.5 : LM3
|   |   |   Year >  1978.5 : LM4
|   Hpwr >  141 :
|   |   Wt_Lbs <= 4361.5 :
|   |   |   Year <= 1977.5 :
|   |   |   |   Wt_Lbs <= 3682.5 : LM5
|   |   |   |   Wt_Lbs >  3682.5 : LM6
|   |   |   Year >  1977.5 :
|   |   |   |   Wt_Lbs <= 3997 : LM7
|   |   |   |   Wt_Lbs >  3997 : LM8
|   |   Wt_Lbs >  4361.5 :
|   |   |   Year <= 1974.5 : LM9
|   |   |   Year >  1974.5 : LM10
```

Model Details

LM num: 1
MiPerGal =
-0.0219 * Cyl
- 0.0438 * CuPerIn
- 0.0582 * Hpwr
- 0.0053 * Wt_Lbs
+ 0.911 * Year
- 1750.8314

LM num: 2
MiPerGal =
-0.0261 * Cyl
- 0.0163 * CuPerIn
- 0.0019 * Hpwr
- 0.0031 * Wt_Lbs
+ 0.3404 * Year
- 638.9453

LM num: 3
MiPerGal = -0.0261 * Cyl
- 0.0242 * CuPerIn- 0.01 * Hpwr
- 0.0023 * Wt_Lbs - 0.2648 * Acc_0-60
+ 0.2856 * Year - 526.4988

LM num: 4
MiPerGal = -0.0261 * Cyl
+ 0.011 * CuPerIn - 0.0169 * Hpwr
- 0.0052 * Wt_Lbs+ 1.4864 * Year
- 2905.0233

LM num: 5
MiPerGal = -0.0261 * Cyl
- 0.0017 * CuPerIn - 0.0143 * Hpwr
- 0.0014 * Wt_Lbs - 0.0578 * Acc_0-60
+ 0.0007 * Year + 22.552

Details, continued

LM num: 6

MiPerGal = -0.0261 * Cyl

- 0.0009 * CuPerIn

- 0.0056 * Hpwr

- 0.0014 * Wt_Lbs

- 0.0578 * Acc_0-60

+ 0.1733 * Year

- 319.7441

LM num: 7

MiPerGal = -0.0261 * Cyl

- 0.0056 * Hpwr

- 0.0023 * Wt_Lbs

- 0.0578 * Acc_0-60

+ 0.2464 * Year

- 460.0164

LM num: 8

MiPerGal = -0.0261 * Cyl

- 0.0056 * Hpwr - 0.002 * Wt_Lbs

- 0.0578 * Acc_0-60+ 0.2464 * Year

- 461.4961

LM num: 9

MiPerGal = -0.0261 * Cyl

- 0.0204 * Hpwr- 0.0024 * Wt_Lbs

- 0.4546 * Acc_0-60 + 0.3107 * Year

- 579.9211

LM num: 10

MiPerGal = -0.0261 * Cyl

- 0.0041 * Hpwr

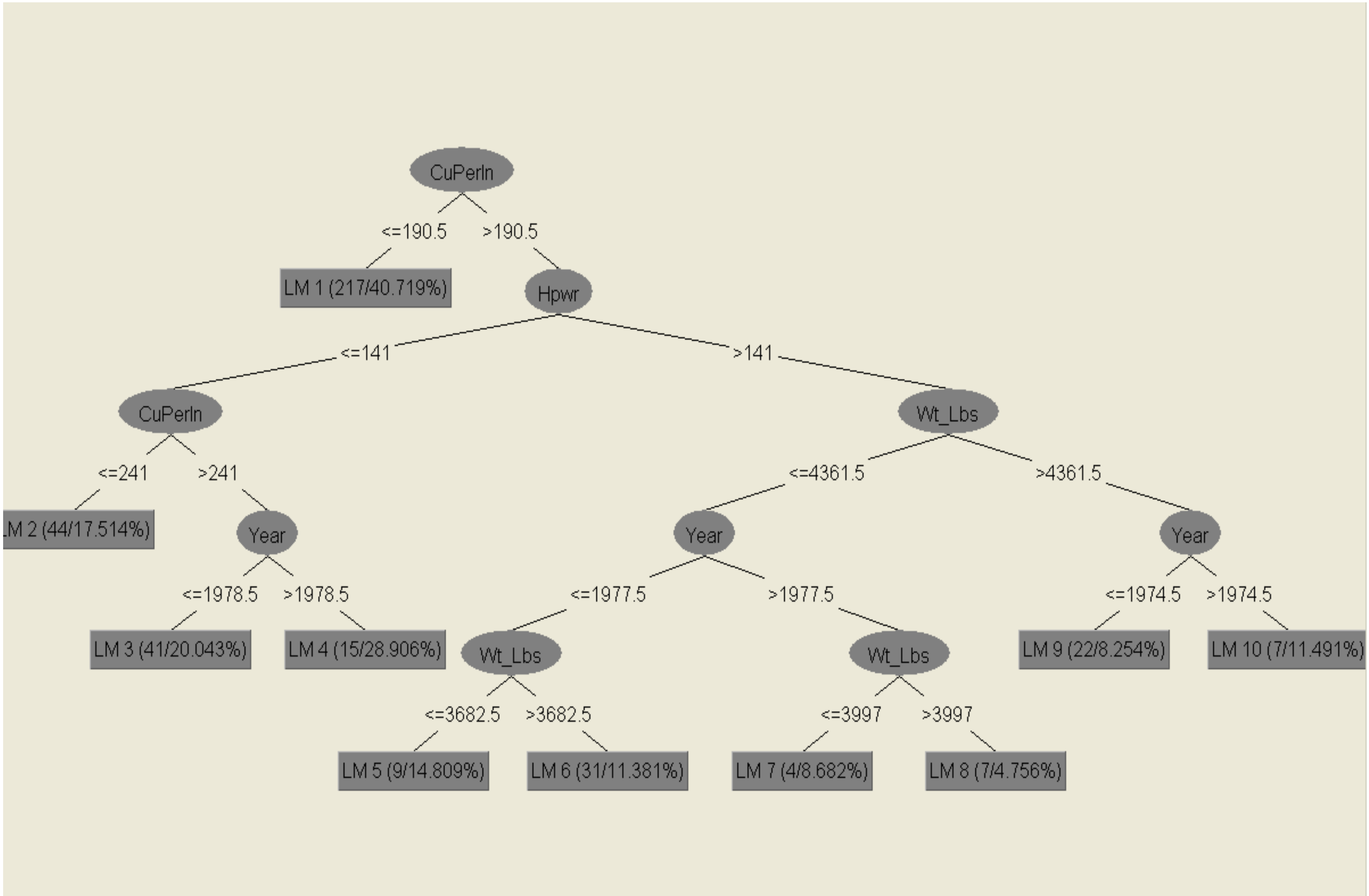
- 0.0022 * Wt_Lbs

- 0.2926 * Acc_0-60

+ 0.3885 * Year

- 738.4909

Model Presentation



Summary

- Start-to-end easy data mining project
- The importance of Problem Definition/Business Statement/Expert Input
- The importance of Data Understanding
- The importance of Data Preparation
- Data mining expertise and understanding of the methods and the evaluation
- Solution presentation

Conclusion

- Solid data mining practice:
  - 8.6634% to ~93% evaluation score improvement!!!
- Incorrect approach:
  - False 99.5% accuracy

Next

- Assignment I
- Lesson 3: Moving on with Our Practical Data Mining