Using classifiers for mail promotions. Part I. Building response predictor

Lab 2.1

Lab consists of two parts: classification and business analysis

- Part I. Data mining: build the classifier and use it for the prediction of potential responders
 - Part II. Business analytics: how to design the most profitable campaign

Plan

Part I. Data Mining. Classification with WEKA.

- 1. Prepare data
- 2. Build several classifiers. Choose the most accurate one.
- 3. Divide dataset into training and validation datasets
- 4. Predict class in the validation dataset
- 5. Prepare output for business analysis
- Part II. Business analysis
 - 1. Generate Lift chart(s)
 - 2. Cost-benefit analysis
 - 3. Recommendations

Dataset

- Load bank_data.csv into WEKA explorer
- Save file as bank1.arff

🖉 Weka Explorer	
Preprocess Classify Cluster Associate Select attributes Visualize	
Open file Open URL Open DB Gener	rate Undo Edit Save
Filter	
Choose None	Apply
Current relation Relation: bank Instances: 300 Attributes: 9	Selected attribute Name: married Type: Nominal Missing: 0 (0%) Distinct: 2 Unique: 0 (0%)
Attributes	No. Label Count
All None Invert Pattern	1 YES 202
No Name	2 NU 90
1 age 2 sex 3 region 4 income 5 marginal	
6 dhildren 7 car 8 mortgage 9 pep	Class: pep (Nom) Visualize All
Remove	98
Status OK	Log 💉 x0



Dataset: explore available attributes in text editor

- @relation bank-data
- @attribute id

{ID12101,ID12102,ID12103,ID12104,ID12105,ID12106,ID12107,ID12108,ID12109,ID12110,ID 12111,ID12112,ID12113,ID12114,ID12115,ID12116,ID12117,ID12118,ID12119,ID12120,ID12 121,ID12122,ID12123,ID12124,ID12125,ID12126,ID12127,ID12128,ID12129,ID12130,ID1213 1,ID12132,ID12133,ID12134,ID12135,ID12136,ID12137,ID12138,ID12139,ID12140,ID12141,I D12142,ID12143,ID12144,ID12145,ID12146,ID12147,ID12148,ID12149,ID12150,ID12151,ID1 2152,ID12153,ID12154,ID12155,ID12156,ID12157,ID12158,ID12159,ID12160,ID12161,ID121 ...

- @attribute age numeric
- @attribute sex {FEMALE,MALE}
- @attribute region {INNER_CITY,TOWN,RURAL,SUBURBAN}
- @attribute income numeric
- @attribute married {NO,YES}
- @attribute children numeric
- @attribute car {NO,YES}
- @attribute save_act {NO,YES}
- @attribute current_act {NO,YES}
- @attribute mortgage {NO,YES}
- @attribute pep {YES,NO} -

Class attribute: bought Personal Equity Plan after the last mailing



- training and validation datasets
- 4. Predict class in the validation dataset
- 5. Prepare output for business analysis

Dataset: working with attributes

- @relation bank-data
- @attribute id {ID12101,ID12102,ID12103,ID12104,ID12105,ID12106,ID12107,ID 12108,ID12109,ID12,10,ID12111,ID12112,ID12113,ID12114,ID121 15,ID12116,ID12117,ID12118,ID12119,ID12120,ID12121,ID12122,I D12123,ID12124,ID12125,12126,ID12127,ID12128,ID12129,ID1 2130,ID12131,ID12132,ID1213,ID12134,ID12135,ID12136,ID1213 7,ID12138,ID12139,ID12140,ID12,1,ID12142,ID12143,ID12144,ID 12145,ID12146,ID12147,ID12148,ID1,19,ID12150,ID12151,ID121 52,ID12153,ID12154,ID12155,ID12156,157,ID12158,ID12159,I D12160,ID12161,ID12162,ID12163,ID1216,ID12151,ID121 2167,ID12168,ID12169,ID12170,ID12171,ID12,12173,ID1217 4,ID12175...
- @attribute age numeric
- @attribute sex {FEMALE,MALE}
- @attribute region {INNER_CITY,TOWN,RURAL,SUBURBAN}
- @attribute income numeric
- @attribute married {NO,YES}
- @attribute children numeric
- @attribute car {NO,YES}
- @attribute save_act {NO,YES}
- @attribute current_act {NO,YES}
- @attribute mortgage {NO,YES}
- @attribute pep {YES,NO}

Non-predictive attribute: remove it and save file

- Part I. Data Mining.
 1. Prepare data
 2. Build several classifiers. Choose the most accurate one.
 3. Divide dataset into training and validation datasets
 4. Predict class in the validation dataset
 - 5. Prepare output for business analysis

Dataset: working with attributes

- @relation bank-data
- @attribute age numeric
- @attribute sex {FEMALE,MALE}
- @attribute region {INNER_CITYTOWN, RURAL, SUBURBAN}
- @attribute income numeric
- @attribute married {NO,YES}
- @attribute children numeric
- @attribute car {NO,YES}
- @attribute save_act {NO,YES}
- @attribute current_act {NO,YES}
- *@attribute mortgage {NO,YES}*
- @attribute pep {YES,NO}

Numeric attributes Part I. Data age and income: Mining. discretize into groups ▶ 1. Prepare data 2. Build several classifiers. Choose the most accurate one. 3. Divide dataset into training and validation datasets 4. Predict class in the validation dataset 5. Prepare output for business analysis

- Simple discretization techniques: distribute numeric values into a predefined number of bins
- Equal intervals: the bins are defined as equal-size numeric intervals
 - Equal frequency: the bins are defined such as to contain equal number of instances in each interval
- In WEKA: Filter: Choose -> Filters-> Unsupervised -> Attribute-> Discretize.
- Left-click to open parameters window



• Age	😧 weka.gui.GenericObjectEditor	Explains parameters
	weka.filters.unsupervised.attribute.Discretize About An instance filter that discretizes a range of numeric attributes in the dataset into nominal attributes. Capabilities	
	attributeIndices 1	Index of the attribute
Finds optimal	bins 8	to apply filter off. I
number of bins	desiredWeightOfInstancesPerInterval -1.0	Number of bins:
by data mining	findNumBins False	based on min-max
techniques	ignoreClass False	values and common
	invertSelection False	sense
If true - equal	makeBinary False	
frequency binning,	useEqualFrequency False	
if false – equal interval binning	Open Save OK Cancel	

The number of bins is found experimentally, by observing the distribution of the class label in different bins. To play with different settings, use the Undo button

• Age after discretization

Current relation Relation: bank-weka.filters.unsupervised.attribute.Discretize-B6-M-1 Instances: 300 Attributes: 9	Selected Name: Missing:	attribute age 0 (0%)	Distinct: 8	Type: Unique:	Nominal 0 (0%)
Attributes	No.	Label		Count	
All None Invert Dattern	1	1 '(-inf-24.125]'		41	
		2 (24.125-30.25]'	36	
		3 (30.25-36.375]'	36	
No. Name		4 '(36.375-42.5]'		36	
1 🦳 age		5 (42.5-48.625)		45	
2 sex		6 (48.625-54.75]'	33	
3 region		7 (54.75-60.875]'	26	
4 income		0 1/00 075 :-01		47	
5 married	Class: pep	(Nom)			 Visualize All
6 children					
7 car	-				
8 mortgage	-		45		47
9 pep	41				
		36 36	36	33	
					26
Remove					
tabus.					

• Income

😋 weka.gui.GenericObjectEditor	×
weka.filters.unsupervised.attribute.Disc About	retize
An instance filter that discretizes attributes in the dataset into nom	a range of numeric More inal attributes. Capabilities
attributeIndices	4
bins	6
desiredWeightOfInstancesPerInterval	-1.0
findNumBins	False 💌
ignoreClass	False 💌
invertSelection	False 💌
makeBinary	False 💌
useEqualFrequency	False 👻
Open Save	OK Cancel

• Income after discretization

Relation: bank-weka.niters.unsupervised.attribute.Discretize-b10-M Instances: 300 Attributes: 9	Name: Missing:	: Income : 0 (0%)	Distinct: 6	Unique: 0	ominai (0%)
Attributes	No.	Label		Count	
All None Invert Pattern		1 '(-inf-14700.	.191667]' 667-24386.1733	44	
No. Name		3 '(24386.173	333-34072.155]	69	
1 age		5 '(43758,136	-43758.130007j 667-53444.1183	45 26	
2 sex	1	6 '(53444.118	333-inf)'	15	
3 region				1	
4 income		A. A.		10	
6 children	Class: pep	o (Nom)		•	Visualize All
7 car					
8 mortgage]	101			
9pep					
			69		
	44		40		
Remove				26	15
	-				

Optional. [Binaryze multi-valued attributes]

• Undo

reprocess Classify Cluster Associate Select attributes Visualize		
Open file Open URL Open DB Gene	rate Undo	Edit Save
Filter		
Choose NominalToBinary -R 3		Apply
Current relation Relation: bank-weka.filters.unsupervised.attribute.Discretize-B6-M-1 Instances: 300 Attributes: 12	Selected attribute Name: region=RURAL Missing: 0 (0%) Distin	Type: Numeric ct: 2 Unique: 0 (0%)
Attributes	Statistic	Value
	Minimum	0
All None Invert Pattern	Maximum	1
No. Norma	Mean	0.17
No. Name	StdDev	0.376
2 sex 3 region=INNER_CITY 4 region=RURAL		
5 region=TOWN	Class: pep (Nom)	✓ Visualize /
6 region=SUBURBAN		
/ income	240	
9 children	248	
10 car		
11 mortgage		
12 pep		
		51
Remove	0 0	0
	0	0.5
tatus XK		Log

Dataset: working with attributes - children

- @relation bank-data
- *@attribute age numeric*
- @attribute sex {FEMALE, MALE}
- @attribute region {INNER_CITY,TOWN,RURAL,SUBURBAN}
- @attribute income numeric
- @attribute married {NO,YES}
- @attribute children numeric -
- @attribute car {NO,YES}
- @attribute save_act {NO,YES}
- @attribute current_act {NO,YES}
- @attribute mortgage {NO,YES}
- @attribute pep {YES,NO}

Not multi-valued: convert to nominal

Convert numeric to nominal

• Filters->Unsupervised->attribute -> NumericToNominal

🕢 weka.gui.Ger	ericObjectEditor	×	Index of the attribut	te
weka.filters.unsup	ervised.attribute.NumericToNominal		to apply filter on: 6	5
About A filter for turn	ing numeric attributes into nominal ones.	More Capabilities		
attributeIndices	6			
debug	False	-		
invertSelection	False	v		
Open	Save OK	Cancel		

Convert numeric to nominal

• Children after nominalizations: 4 groups

🕢 Weka Explorer							
Preprocess Classify Cluster Associate	Select attributes 🛛 Visua	lize					
Open file Open URL	Open DB	Genera	ate	Undo	Edit		Save
Filter							
Choose NumericToNominal -R 6							Apply
Current relation			Selected attr	ribute			
Relation: bank-data-weka.filters.unsupe Instances: 600 At	ervised.attribute.Remove ttributes: 11	e-R1	Name: ch Missing: 0	ildren (0%) Dis	tinct: 4	Type: N Upique: 0	lominal (0%)
Attributes			No.	Label		Co	unt
		. 1	1	0		263	
AllNone	Invert Pat	tern	2	1		135	
No	Name		3	2		134	
	Name		4	3		00	
2 sex							
3 region							
4 income							
5 🗖 married			Class: nen (N			-	Visualize All
6 children							
7 car							
8 save_act			263				
10 mortgage							
				135	134		
Remove						C C C C C C C C C C C C C C C C C C C	8
OK OK						Log	×0

Save the resulting dataset as 'bank2.arff'

Part I. Data

▶ 1. Prepare data

2. Build several classifiers.

Mining.

Choose the most accurate one. 3. Divide dataset into training and validation datasets

 This is the input to our classifiers

😋 Weka Explorer		-		 4. Predict class in the validation dataset 5. Prepare output for
Preprocess Classify Cluster Associate Select attributes Visualize				business analysis
Open file Open URL Open DB Gener	ate	Undo Edit	Save	
Choose NominalToBinary -R 3			Apply	
Current relation Relation: bank-weka.filters.unsupervised.attribute.Discretize-B6-M-1 Instances: 300 Attributes: 9	Selected attr Name: ind Missing: 0	ribute come (0%) Distinct: 6	Type: Nominal Unique: 0 (0%)	
Attributes	No.	Label	Count	
	1	'(-inf-14700.191667]'	44	
	2	'(14700.191667-24386.1733	101	
	3	'(24386.173333-34072.155]'	69	

Classification

• Our goal: the most accurate classifier

Algorithm	Dataset	Accuracy

Part I. Data Mining. 1. Prepare data ▶ 2. Build several classifiers. Choose the most accurate one. 3. Divide dataset into training and validation datasets 4. Predict class in the validation dataset 5. Prepare output for business analysis

Classification. Trees: J48

🕢 Weka Explorer		1	
Preprocess Classify Clus er Associate S Classifier	elect attributes Visualize		Part I. Data
Choose J48 -C 0.25 -M 2		Accuracy:	Mining.
Test options	Classifier output	89.5%	1 Droparo data
O Use training set			I. Prepare uata
C Supplied test set Set	Time taken to build model: 0.08 seconds		2. Build several
Cross-validation Folds 10	=== Stratified cross-validation ===	_	classifiers
C Percentage split % 66	=== Sunnary ===		Chaosa tha mast
More options	Correctly Classified Instances 537 89.5 %		accurate one.
(Nom) pep 🔹	Kappa statistic 0.7866		3 Divide dataset into
	Mean absolute error 0.1767		
StartStop	Root mean squared error 0.3063		training and
Result list (right-click for options)	Relative absolute error 35.6033 %		validation
09:59:30 - trees.J48	Root relative squared error b1.497 %		datacata
			udidsels
	=== Detailed Accuracy By Class ===		4. Predict class in
	TP Rate FP Rate Precision Recall F-Measure ROC Area		the validation
	0.832 0.052 0.931 0.832 0.879 0.902		tasset
	0.948 0.168 0.87 0.948 0.907 0.902		udidact
	Weighted Avg. 0.895 0.115 0.898 0.895 0.894 0.902		5. Prepare output for
	=== Confusion Matrix ===		husiness analysis
			business analysis
Status		1	
OK	Log X0		
L			

Report

Algorithm	Dataset	Accuracy
J48	bank2.arff	89.5

Attribute selection. Decision tree: J48

- The most important attributes (used in the tree for splitting nodes): children, married, mortgage, save_act, income
- Let's remove the rest of the attributes (but leave the class attribute!), save file as 'bank3.arff' and try J48 again

```
children = 0
  married = NO
     mortgage = NO: YES (48.0/3.0)
     mortgage = YES
        save act = NO: YES(12.0)
        save act = YES: NO(23.0)
  married = YES
     save act = NO
        mortgage = NO: NO (36.0/5.0)
        mortgage = YES: YES
(25.0/3.0)
  save act = YES: NO (119.0/12.0)
children = 1
  income = '(-inf-14700.191667]': NO
(21.0/3.0)
  income = '(14700.191667-
24386.173333]': YES (45.0/3.0)
  income = '(24386.173333-
34072.155]': YES (33.0/2.0)
  income = '(34072.155-
```

Type I classifiers.

Decision tree: J48 on reduced dataset

• Even better accuracy. Record

闭 Weka Explorer				_ 🗆 🗵
Preprocess Classify Cluster Associate	Select attributes Visualize			
Classifier				
Choose 348 -C 0.25 -M 2				
Test options	Classifier output			
C Use training set	income = '(53444.118333-inf)': YF	5555] . 125 (7.67 ES (1.0)		
C Supplied test set Set				
Cross-validation Folds 10	Number of Leaves : 24			
C Percentage split % 66	Size of the tree : 33			
More options				
(Nom) pep	Time taken to build model: 0.02 secon	nds		
Start Stop	=== Stratified cross-validation === === Summary ===			
Result list (right-click for options)				
09:59:30 - trees.J48	Correctly Classified Instances	538	89.6667 %	
10:04:11 - trees.J48	Norrectly classified instances	0 2001	10.3333 %	
	Meen ebsolute error	0.7901		
	Root mean squared error	0.3056		
	Relative absolute error	35 5796 %		
	Root relative squared error	61.3513 %		
	Total Number of Instances	600		
	Detailed Assurant By Class			
	Decalled Accuracy by class ===			-
	•			
Status				
ОК			Log	🐨 ×0

Report

Algorithm	Dataset	Accuracy
J48	bank2.arff	89.5
J48	bank3.arff	89.7

- Part I. Data Mining.
 1. Prepare data
 2. Build several classifiers. Choose the most
 - accurate one. 3. Divide dataset into training and validation
 - datasets
 - 4. Predict class in the validation dataset
 - 5. Prepare output for business analysis

Decision trees: Id3 and Simple cart

Algorithm	Dataset	Accuracy, %
J48	bank2.arff	89.5
J48	bank3.arff	89.7
ld3	bank2.arff	77.0
ld3	bank3.arff	86.0
SimpleCart	bank2.arff	86.8
SimpleCart	bank3.arff	89.5

The best accuracy for decision trees: J48 and on bank3.arff

Report so far

Algorithm	Dataset	Accuracy, %
J48	bank3.arff	89.7



Type 2 classifiers - Rules: DecisionTable on the full dataset bank2.arff



Attribute selection: DecisionTable on the full dataset

- The most important attributes:
 - 4- income
 - 5- married
 - 6- children
 - 8- save_act
 - 10 mortgage
- Let's remove the rest
- Save file as bank4.arff
- Re-build decision tree J48: accuracy 89.7 – very high!
- We will use bank4.arff as our input for the rest of the lab

ØWeka Explorer			
Preprocess Classify Cluster Associate S	ielect attributes Visualize		
Classifier			
Choose DecisionTable -X 1 -5 "wek	a.attributeSelection.BestFirst -D 1 -N 5"		
Test options	Classifier output		
C Use training set	Merit of best subset found	araacca. 55 : 82.5	
C Supplied test set Set	Evaluation (for feature selection)	. V (leave	one
Cross-validation Folds 10	Feature set: 4,5,6,8,10,11	J	
C Percentage split % 66	Time taken to build model: 0.14 se	conds	
More options	=== Stratified cross-validation == === Sunmary ===	-	
	Correctly Classified Instances	489	
Start Stop	Incorrectly Classified Instances	111	
Result list (right-click for options)	Kappa statistic	0.623	38
09/59/30 - trees 148	Mean absolute error	0.314	48
10:04:11 - trees. 348	Root mean squared error	0.386	52
10:09:53 - trees.Id3	Relative absolute error	63.423	35 %
10:10:18 - trees.SimpleCart	Root relative squared error	77.53	16 %
10:11:06 - trees.SimpleCart	Total Number of Instances	600	
10:11:56 - trees.Id3 10:15:14 - rules.DecisionTable	=== Detailed Accuracy By Class ===		
	TP Rate FP Rate	Precision	Re
	0.741 0.123	0.835	0
	0.877 0.259	0.801	0
· · · · · · · · · · · · · · · · · · ·	tr.		

The rest of the Rule learners on bank4.arff

	Algorithm	Dataset	Accuracy, %
	J48	bank3.arff	89.7
J48		bank4.arff	89.7
	JRip	bank4.arff	87.8
Dulas	Part	bank4.arff	88.3
Rules	Prism	bank4.arff	67.3
	Ridor	bank4.arff	84.7

The best result for rule learners



- 4. Predict class in the validation dataset
- 5. Prepare output for business analysis

Report so far

Algorithm	Dataset	Accuracy, %
J48	bank4.arff	89.7
Part	bank4.arff	88.3

Type III classifiers: Naïve Bayes

- For 'bank2.arff' (full dataset): 70.5% accurate
- For 'bank3.arff' (J48 reduction): 72.5% accurate
- For 'bank4.arff' (DecisionTable reduction): 72.5% accurate



Report

Algorithm	Dataset	Accuracy, %
J48	bank4.arff	89.7
Part	bank4.arff	88.3
NaiveBayes	bank4.arff	72.5

Part I. Data Mining. 1. Prepare data 2. Build several classifiers. Choose the most accurate one. 3. Divide dataset into training and validation datasets 4. Predict class in the validation dataset 5. Prepare output for business analysis

Generating validation dataset

- We will use 70% of the data for training the classifier, and 30% for the validation
- The validation dataset contains actual responses, but we will try to predict them with our best classifier, to see how good is the prediction



Generating output for business analysis

- Re-open bank4.arff
- Choose one of our best classifiers: J48
- Test options: Percentage split
- Press More Options button

🕣 Weka Explorer				
Preprocess Classify Cluster Associate	Se	elect attribute	es Visualize	
Classifier				
Choose J48 -C 0.25 -M 2				
Test options		Classifier out	:put	
C. Use training set				
C use training set		Time tak	en to buil	ld model:
C Supplied test set Set				
C Cross-validation Folds 10		=== Pred	ictions or	ntest spli
Percentage split % 70		inst#,	actual,	predicted
More options		1	2:NO	2:N
More options		2	1:YES	1:YE:
	1	3	2:NO	2:N
(Nom) pep		4	2:NO	2:N
Charak Chara	e ll	5	1:YES	1:YE:
Start		6	2:NO	2:N
Result list (right-click for options)	٦	7	2:NO	2:N
		8	1:YES	2:N
10:15:14 - rules.Decision l'able		9	1:YES	1:YE:
10:22:56 - trees.J46		10	2:NO	2:N
10:20:32 - rules.JRIP		11	2:NO	2:N
10:27:00 - rules.PART		12	1:YES	1:YE:
10:22:29 - rules Pider		13	2:NO	2:N
10:20:33 - haves NaiveBaves		14	1:YES	1:YE:
10:32:59 - bayes NaiveBayes		15	2:NO	2:N
10:32:33 - bayes NaiveBayes		16	2:NO	2:N
10:40:39 - trees, 148		17	1:YES	1:YE:
10:41:47 - trees.J48		18	1:YES	1:YE:
10:42:06 - trees.J48		•		

Generating output for business analysis

- Check: Output predictions
- Run J48 Decison tree classifier

Classifier evaluation options	
V Dutput model	
✓ Output per-class stats	
Output entropy evaluation measures	
✓ Output confusion matrix	
Store predictions for visualization	
Output predictions	
Output additional attributes	
Cost-sensitive evaluation Set	
Random seed for XVal / % Split 1	
✓ Preserve order for % Split	
Cutput source code WekaClassifier	
ОК	

Predict class in the validation dataset

 Run J48 using training and validation datasets. Note that the accuracy has decreased.



Transfer prediction into a text file

- Copy predictions and paste into a text file
- Save file as bank_predicted .txt
- Do find *,+ and replace them with a space

<u>F</u> ind Next
<u>R</u> eplace
Replace <u>A</u> ll
Cancel

reprocess Classify Cluster Associate Select attributes Visualize Classifier Choose 148 - C 0.25 - M 2 Image: Classifier output Image: Classifier output	🕽 Weka Explorer						
Classifier Choose 148 -C 0.25 -M 2 Test options Use training set Supplied test set Cross-validation Folds 10 Percentage split % 70 More options Nom) pep Start Stop Start Stop 166 2: N0 2: N0 0.0111 *0.889 167 1: YES 1: YES *0.923 0.077 166 2: N0 2: N0 0.111 *0.889 167 1: YES 1: YES *0.923 0.077 168 1: YES 1: YES *0.923 0.077 169 2: N0 0.0111 *0.889 170 2: N0 0.011 *0.786 171 2: N0 2: N0 0.011 *0.786 172 1: YES 2: N0 + 0.263 *0.737 173 2: N0 2: N0 0.011 *0.99 174 1: YES 1: YES *1 0 175 2: N0 2: N0 0.076 *0.924 176 2: N0 1: YES + *1 0 175 2: N0 2: N0 0.076 *0.924 178 1: YES 2: N0 + 0.076 *0.924 178 1: YES 2: N0 + 0.076 *0.924 178 1: YES 2: N0 + 0.076 *0.924 178 1: YES 1: YES *1 0 10:27:37 -uies.PART 10:27:39 - rules.PART 10:27:49 - rules.NaiveBayes 23 - bayes.NaiveBayes 39 - trees.J48 To crrectly Classified Instances 158 Incorrectly Classified Instances 22 4 4 4 4 4 4 4 4 4 4 4 4 4	Preprocess Classify Cluster Associate	Select attributes	Visualize				
Choose J48 - C 0.25 - M 2 Test options Classifier output C Use training set 163 2:N0 0.077 *0.923 Supplied test set Set 163 2:N0 0.263 *0.737 C Cross-validation Folds 10 165 1:YES 1:YES *0.923 0.077 C Cross-validation Folds 10 165 1:YES 1:YES *0.923 0.077 C Percentage split % 70 166 2:N0 2:N0 0.111 *0.889 167 1:YES 1:YES *0.923 0.077 168 1:YES *0.923 0.077 168 1:YES 1:YES *0.923 0.077 168 1:YES *0.923 0.077 169 2:N0 2:N0 0.0111 *0.889 167 1:YES *0.924 171 2:N0 0.0111 *0.89 171 2:N0 0.011 *0.9 175 2:N0 0.011 *0.9 175 2:N0 0.076 *0.924 177 2:N0 2:N0 0.076 *0.924 177 173 175 2:N0 0.076 *0.924 177	Classifier		· ·				
CHOOSE 140 C 0.23 H 12 Test options C Use training set Supplied test set Set C Cross-validation Folds 10 Percentage split % 70 More options More options Nom) pep Start Stop Start Stop Start Stop Start Stop 164 2: N0 2: N0 0.263 *0.737 165 1: YES 1: YES *0.923 0.077 166 2: N0 2: N0 0.111 *0.889 167 1: YES 1: YES *0.927 0.083 169 2: N0 2: N0 0.076 *0.924 170 2: N0 2: N0 0.0111 *0.889 171 2: N0 2: N0 0.0111 *0.889 172 1: YES 2: N0 + 0.263 *0.737 173 2: N0 2: N0 0.0111 *0.889 172 1: YES 2: N0 + 0.263 *0.737 173 2: N0 2: N0 0.0111 *0.899 172 1: YES 2: N0 + 0.263 *0.737 173 2: N0 2: N0 0.011 *0.924 174 1: YES 1: YES *1 0 175 2: N0 2: N0 0.076 *0.924 176 2: N0 1: YES + *1 0 177 2: N0 2: N0 0.076 *0.924 178 1: YES 2: N0 + 0.076 *0.924 179 2: N0 2: N0 0.076 *0.924 180 1: YES 1: YES *1 0 === Evaluation on test split === === Summary === Correctly Classified Instances 158 Incorrectly Classified Instances 22 * *	Choose 149 C 0 25 M 2						
Test options Classifier output Use training set 163 2:N0 2:N0 0.077 *0.923 164 2:N0 2:N0 0.263 *0.737 Supplied test set Set 165 1:YES 1:YES *0.923 0.077 Cross-validation Folds 166 2:N0 0.0111 *0.889 Cross-validation Folds 166 2:N0 0.111 Percentage split % 70 168 1:YES % 0.923 0.077 More options 169 2:N0 0.076 0.924 To 2:N0 2:N0 0.076 0.924 To Start Stop 173 2:N0 0.076 0.924 172 172 175 175 1	Choose 348-C 0.23-M 2						
C Use training set 163 2:N0 2:N0 0.077 *0.923 Supplied test set Set 164 2:N0 0.263 *0.737 C Cross-validation Folds 10 165 1:YES *0.923 0.077 C Cross-validation Folds 10 166 2:N0 2:N0 0.111 *0.889 I 167 1:YES 1:YES *0.923 0.077 166 2:N0 2:N0 0.111 *0.889 I 167 1:YES *0.923 0.077 I 166 2:N0 2:N0 0.0111 *0.889 I 170 2:N0 2:N0 0.076 *0.924 I 171 2:N0 2:N0 0.076 *0.924 I 173 2:N0 2:N0 0.076 *0.924 I 174 1:YES 1:YES *1 0 I0:25:32 rules.DecisionTable 177 2:N0 2:N0 0.076 *0.924 I0:22:58 trees.J48	Test options	Classifier output	ut				
Supplied test set Set Supplied test set Set Cross-validation Folds Percentage split % 70 More options 166 Nom) pep 167 1:YES 1:YES *0.923 0.077 166 2:N0 0.111 *0.889 167 1:YES 168 1:YES 169 2:N0 170 2:N0 2:N0 2:N0 0.214 *0.923 168 1:YES 172 1:YES 171 2:N0 2:N0 2:N0 0.214 *0.923 171 2:N0 2:N0 2:N0 0:171:S0 2:N0 10:22:58 - trees.J48 177 10:22:58 - trees.J48 1:YES 10:22:59 - rules.PART 1:YES 10:22:59 - rules.PART 1:YES 10:22:59 - bayes.NaiveBayes 1:YES 39 - bayes.NaiveBayes 1:YES 39 - trees.J48 . <tr< td=""><td>O Use training set</td><td>163</td><td>2:NO</td><td>2:N0</td><td></td><td>0.077</td><td>*0.923</td></tr<>	O Use training set	163	2:NO	2:N0		0.077	*0.923
C Supplied test set Set C Cross-validation Folds IO Cross-validation Folds Percentage split % 70 Ic6 1: YES 1: YES More options 166 1: YES More options 167 1: YES More options 169 2: N0 0.077 168 1: YES *0.923 0.077 168 1: YES 1: YES *0.923 0.077 168 1: YES 1: YES *0.923 0.077 168 1: YES 1: YES *0.917 0.083 170 2: N0 2: N0 0.011 *0.824 171 2: N0 2: N0 0.076 *0.924 172 1: YES 1: YES *1 0 10:22:58 - trees.J48 10:27:09 1: YES 1: N0 0.076 *0.924 </td <td></td> <td>164</td> <td>2:NO</td> <td>2:NO</td> <td></td> <td>0.263</td> <td>*0.737</td>		164	2:NO	2:NO		0.263	*0.737
C Cross-validation Folds 10 ● Percentage split % 70 ● Percentage split % 70 ● More options 166 1: YES 1: YES ● More options 169 2: N0 0.076 *0.923 0.077 168 1: YES 1: YES *0.917 0.083 ● More options 169 2: N0 0.076 *0.924 170 2: N0 2: N0 0.0111 *0.889 169 2: N0 2: N0 0.0111 *0.889 171 2: N0 2: N0 0.111 *0.889 172 1: YES 2: N0 0.0111 *0.889 172 1: YES 2: N0 0.0111 *0.889 172 1: YES 2: N0 2: N0 0.0111 *0.89 10:15:14 rules.DecisionTable 175 2: N0 2: N0 0.076 *0.924 10:27:03 rules.Prism 1: YES 1: YES 1: 0 0.076 *0.924 10:27:29 rules.Prism 1: YES 1: YES	C Supplied test set Set	165	1:YES	1:YES		*0.923	0.077
 Percentage split % 70 167 1:YES 1:YES *0.923 0.077 168 1:YES 1:YES *0.917 0.083 169 2:N0 2:N0 0.076 *0.924 170 2:N0 2:N0 0.214 *0.786 171 2:N0 2:N0 0.111 *0.889 172 1:YES 2:N0 + 0.263 *0.737 173 2:N0 2:N0 0.076 *0.924 174 1:YES 1:YES *1 0 175 2:N0 2:N0 0.1 *0.924 176 2:N0 1:YES *1 0 177 2:N0 2:N0 0.1 *0.9 176 2:N0 1:YES + *1 0 177 2:N0 2:N0 0.076 *0.924 178 1:YES 2:N0 + 0.076 *0.924 178 1:YES 2:N0 + 0.076 *0.924 178 1:YES 2:N0 + 0.076 *0.924 179 2:N0 2:N0 0.076 *0.924 179 2:N0 2:N0 0.076 *0.924 179 2:N0 2:N0 0.076 *0.924 180 1:YES 1:YES *1 0 180 1:YES 1:YES *1 0 180 1:YES 1:YES *1 0 190 2:N0 2:N0 0.076 *0.924 191 2:N0 2:N0 0.076 *0.924 192 2:N0 2:N0 0.076 *0.924 192 2:N0 2:N0 0.076 *0.924 192 2:N0 2:N0 0.076 *0.924 193 2:N0 2:N0 0.076 *0.924 194 1:YES 1:YES *1 0 195 2:N0 2:N0 0.076 *0.924 196 1:YES 1:YES *1 0 	C Cross-validation Folds 10	166	2:N0	2:NO		0.111	*0.889
168 1:YES 1:YES *0.917 0.083 More options 169 2:N0 2:N0 0.076 *0.924 170 2:N0 2:N0 0.214 *0.786 Nom) pep Image: start indice start indindice start indindindice start indice start	C Devreehage celik 9/ 70	167	1:YES	1:YES		*0.923	0.077
More options 169 2:N0 2:N0 0.076 *0.924 Nom) pep I70 2:N0 0.214 *0.786 171 2:N0 2:N0 0.111 *0.889 172 1:YES 2:N0 + 0.263 *0.737 Start Stop 173 2:N0 2:N0 0.076 *0.924 171 2:N0 2:N0 0.0111 *0.889 172 Start Stop 173 2:N0 0.076 *0.924 173 2:N0 2:N0 0.076 *0.924 174 1:YES 1:YES *1 0 10:15:14 - rules.DecisionTable 175 2:N0 0.1 *0.9 176 2:N0 1:YES + *1 0 10:25:258 - trees.J48 1:YES 2:N0 0.076 *0.924 10:27:29 - rules.PRT 1:YES 2:N0 0.076 *0.924 10:32:33 - bayes.NaiveBayes *1 0 10:32:33 - bayes.NaiveBayes *1 0 10:32:33 - bayes.NaiveBayes *1 0 10:32:33 - bayes.NaiveBayes 1:YES 1:YES 10	Percencage split % 70	168	1:YES	1:YES		*0.917	0.083
Nom) pep 170 2:N0 2:N0 0.214 *0.786 Nom) pep 171 2:N0 0.111 *0.889 Start Stop 172 1:YES 2:N0 +0.263 *0.737 Start Stop 173 2:N0 2:N0 0.0111 *0.889 172 1:YES 2:N0 +0.263 *0.737 173 2:N0 0.0076 *0.924 174 1:YES 1:YES *1 0 175 2:N0 0.11 *0.9 10:15:14 - rules.DecisionTable 175 2:N0 1:YES + *1 0 10:26:32 - rules.JRip 10:27:03 - rules.PART 10.076 *0.924 178 1:YES 2:N0 0.076 *0.924 10:28:49 - rules.Ridor 1:YES 1:YES *1 0 1:YES *1 0 10:32:33 - bayes.NaiveBayes 59 - bayes.NaiveBayes *1 0 1:YES *1 0 10:32:33 - bayes.NaiveBayes 59 - bayes.NaiveBayes 1:YES *1 0 10:32:33 - bayes.NaiveBayes 1:YES 1:YES 1:1 0 10:32:33 - bayes.NaiveBayes 59 - brees.J48 1:O	More options	169	2:N0	2:NO		0.076	*0.924
Nom) pep 171 2:N0 2:N0 0.111 *0.889 Start Stop 172 1:YES 2:N0 + 0.263 *0.737 Start Stop 173 2:N0 2:N0 0.076 *0.924 Result list (right-click for options) 174 1:YES 1:YES *1 0 10:15:14 - rules.DecisionTable 175 2:N0 2:N0 0.11 *0.9 10:22:58 - trees.J48 10:26:32 - rules.JRip 177 2:N0 2:N0 0.076 *0.924 10:27:03 - rules.PART 1:YES 2:N0 0.076 *0.924 10:22:33 - bayes.NaiveBayes 1:YES 1:YES *1 0 10:22:33 - bayes.NaiveBayes 1:YES 1:YES *1 0 10:22:33 - bayes.NaiveBayes 1:YES 1:YES *1 0 10:22:33 - bayes.NaiveBayes - 1:YES *1 0 10:22:33 - bayes.NaiveBayes - 1:YES *1 0 10:22:33 - bayes.NaiveBayes - - 1:YES *1 0 10:22:33 - bayes.NaiveB		170	2:N0	2:NO		0.214	*0.786
172 1:YES 2:N0 + 0.263 *0.737 Start Stop 173 2:N0 2:N0 0.076 *0.924 173 2:N0 2:N0 0.076 *0.924 174 1:YES 1:YES *1 0 10:15:14 - rules.DecisionTable 175 2:N0 2:N0 0.1 *0.9 10:22:58 - trees.J48 10:26:32 - rules.JRip 177 2:N0 2:N0 0.076 *0.924 10:27:03 - rules.PART 10:27:29 - rules.Ridor 179 2:N0 2:N0 0.076 *0.924 10:28:49 - rules.Ridor 1:YES 1:YES *1 0 10:22:33 - bayes.NaiveBayes 12 1:YES 1:YES *1 0 10:22:33 - bayes.NaiveBayes 19 1:YES 1:YES *1 0 10:32:33 - bayes.NaiveBayes 19 1:YES 1:YES *1 0 10:32:33 - bayes.NaiveBayes 19 1:YES 1:YES 158 10:00 - trees.J48 10 10:00 - trees.J48 158 10 - trees.J48 10 1:Settif	(Nom) pep	171	2:N0	2:NO		0.111	*0.889
Start Stop Result list (right-click for options) 173 2:N0 2:N0 0.076 *0.924 Result list (right-click for options) 174 1:YES 1:YES *1 0 No.11.50 dccs.nds 175 2:N0 2:N0 0.1 *0.924 10:15:14 - rules.DecisionTable 175 2:N0 2:N0 0.1 *0.9 10:22:58 - trees.J48 10:26:32 - rules.JRip 176 2:N0 2:N0 0.076 *0.924 10:22:59 - rules.PART 178 1:YES 2:N0 + 0.076 *0.924 10:22:703 - rules.PART 10:22:33 - bayes.NaiveBayes 1:YES 1:YES 1:N0 0.076 *0.924 10:22:33 - bayes.NaiveBayes 1:YES 1:YES 1:YES *1 0 10:23:33 - bayes.NaiveBayes 1:YES 1:YES *1 0 10:28:49 - rules.Ridor 1:YES 1:YES *1 0 10:28:49 - rules.NaiveBayes 23 - bayes.NaiveBayes 1:YES 1:S 23 - bayes.NaiveBayes 39 - trees.J48 Incorrectly Classified Instances 158 10 - trees.J48 -		172	1:YES	2:NO	+	0.263	*0.737
174 1:YES *1 0 Result list (right-click for options) 175 2:N0 2:N0 0.1 *0.9 10:15:14 - rules.DecisionTable 176 2:N0 1:YES + *1 0 10:22:58 - trees.J48 10:26:32 - rules.JRip 177 2:N0 2:N0 0.076 *0.924 10:27:03 - rules.PART 10:27:09 - rules.Prism 10:28:49 - rules.Ridor 0.076 *0.924 179 2:N0 2:N0 0.076 *0.924 10:32:33 - bayes.NaiveBayes 10:32:33 - bayes.NaiveBayes 1:YES 1:YES *1 0 10:32:33 - bayes.NaiveBayes 23 - bayes.NaiveBayes 23 - bayes.NaiveBayes 1:YES 1:YES 10 10:32:34 - bayes.NaiveBayes 10:32:33 - bayes.NaiveBayes 1:YES 1:YES 1:YES 1:YES 10:32:33 - bayes.NaiveBayes 10:Correctly Classified Instances 158 10:correctly Classified Instances 158 10:0 - trees.J48 1 1 1 1 1 1 1 10 - trees.J48 1 1 1 1 1 1 1 1 10 - trees.J48 1	Start Stop	173	2:NO	2:NO		0.076	*0.924
Result list (right-click for options) 175 2:N0 2:N0 0.1 *0.9 10:15:14 - rules.DecisionTable 176 2:N0 1:YES + *1 0 10:22:58 - trees.J48 10:26:32 - rules.JRip 177 2:N0 2:N0 0.076 *0.924 10:27:03 - rules.PART 1:YES 2:N0 + 0.076 *0.924 10:27:29 - rules.Pism 1:YES 2:N0 - 0.076 *0.924 10:28:49 - rules.Ridor 1:YES 1:YES *1 10:32:33 - bayes.NaiveBayes 1:YES 1:YES *1 23 - bayes.NaiveBayes 23 - bayes.NaiveBayes 23 - bayes.NaiveBayes 158 1ncorrectly Classified Instances 158 1ncorrectly Classified Instances 22		174	1:YES	1:YES		*1	0
176 2:N0 1:YES + *1 0 10:15:14 - rules.DecisionTable 177 2:N0 2:N0 0.076 *0.924 10:22:58 - trees.J48 178 1:YES 2:N0 + 0.076 *0.924 10:26:32 - rules.JRip 179 2:N0 2:N0 0.076 *0.924 10:27:03 - rules.PART 10:27:29 - rules.Prism 180 1:YES 1:YES *1 0 10:28:49 - rules.Ridor 10:32:33 - bayes.NaiveBayes *1 0 === Evaluation on test split === === === Summary == Summar	Result list (right-click for options)	175	2:N0	2:NO		0.1	*0.9
10:13:14 - Holes, Decision Hable 177 2:N0 2:N0 0.076 *0.924 10:22:58 - trees.J48 1:YES 2:N0 + 0.076 *0.924 10:22:58 - trues.JRip 1:YES 2:N0 + 0.076 *0.924 10:22:03 - rules.PART 1:YES 2:N0 0.076 *0.924 10:27:29 - rules.Prism 10:28:49 - rules.Ridor 1:YES 1:YES *1 0 10:28:33 - bayes.NaiveBayes 59 - bayes.NaiveBayes === Evaluation on test split === === Summary === 23 - bayes.NaiveBayes 39 - trees.J48 Correctly Classified Instances 158 1ncorrectly Classified Instances 22 1 Incorrectly Classified Instances 22	10.1E.14 vulos DesirionTable	176	2:N0	1:YES	+	*1	0
10:22:30 = Grees.340 178 1:YES 2:N0 + 0.076 *0.924 10:26:32 - rules.JRip 179 2:N0 2:N0 0.076 *0.924 10:27:03 - rules.PART 180 1:YES 1:YES *1 0 10:27:29 - rules.Ridor 10:28:49 - rules.Ridor 180 1:YES 1:YES *1 0 10:32:33 - bayes.NaiveBayes 59 - bayes.NaiveBayes 59 - bayes.NaiveBayes *1 0 23 - bayes.NaiveBayes 39 - trees.J48 Correctly Classified Instances 158 47 - trees.J48 * * * * *	10:22:59 - trees 149	177	2:N0	2:NO		0.076	*0.924
10:20:32 Fields JACP 10:27:03 - rules.PART 179 2:N0 2:N0 0.076 *0.924 10:27:03 - rules.PART 180 1:YES 1:YES *1 0 10:27:29 - rules.Ridor 180 1:YES 1:YES *1 0 10:28:49 - rules.Ridor 10:32:33 - bayes.NaiveBayes === Evaluation on test split === === 10:32:33 - bayes.NaiveBayes 23 - bayes.NaiveBayes 23 - bayes.NaiveBayes === Summary === 23 - bayes.NaiveBayes 23 - bayes.NaiveBayes Correctly Classified Instances 158 47 - trees.J48 T Incorrectly Classified Instances 22 4 Incorrectly Classified Instances 22	10:22:30 - 0 665:370	178	1:YES	2:NO	+	0.076	*0.924
10:27:29 - rules.Prism 10:28:49 - rules.Ridor 10:32:33 - bayes.NaiveBayes 10:32:33 - bayes.NaiveBayes 23 - bayes.NaiveBayes 39 - trees.J48 47 - trees.J48 06 - trees.J48 10:27:29 - rules.Prism 180 1:YES 1:YES *1 0	10:27:03 - rules.PART	179	2:N0	2:NO		0.076	*0.924
10:28:49 - rules.Ridor 10:32:33 - bayes.NaiveBayes 23 - bayes.NaiveBayes 39 - trees.J48 47 - trees.J48 66 - trees.J48 47 - trees.J48	10:27:29 - rules.Prism	180	1:YES	1:YES		*1	0
<pre>10:32:33 - bayes.NaiveBayes 59 - bayes.NaiveBayes 23 - bayes.NaiveBayes 47 - trees.J48 66 - trees.J48 70 -</pre>	10:28:49 - rules.Ridor						
59 - bayes.NaiveBayes === Summary === 23 - bayes.NaiveBayes Correctly Classified Instances 39 - trees.J48 Incorrectly Classified Instances 6 - trees.J48 Incorrectly Classified Instances	10:32:33 - bayes.NaiveBayes	=== Evalua	ation on te	st split ===			
23 - bayes.NaiveBayes 39 - trees.J48 47 - trees.J48 06 - trees.J48 Correctly Classified Instances 158 Incorrectly Classified Instances 22	? × 59 - bayes.NaiveBayes	=== Summan	ry ===				
39 - trees.J48 Correctly Classified Instances 158 47 - trees.J48 Incorrectly Classified Instances 22 06 - trees.J48 Image: Correctly Classified Instances 22	23 - bayes.NaiveBayes						
47 - trees.J48 Incorrectly Classified Instances 22	xt 39 - trees. J48	Correctly	Classified	Instances		158	
□ 06 - trees. 348	47 - trees. J48	47 - trees.J48 Incorrectly Classified Instances 22					
	06 - trees. J48 🔽						
	All						

Import predictions into Electronic tables program: example - Excel

Import data from bank_predicted.txt

ſ		1 2 Home	j 📑 Inse	F ert Pag	le Layou	ut For		ata Revie	w Vie			
FI	rom	From Web	From Text Get Ext	From Other Sources *	Ex	isting nections	Refresh All + Conn	Connections Properties Edit Links ections	Â↓ Â Z↓ So			
		A1		- (•	f	x.						
Book1 [Compatibility Mode]												
	đ Bo	ok1 [Co	mpatibil	ity Mode]								
	환 Bo	ok1 [Co	mpatibil E	ity Mode] B	С	D	E	F	G			
	원 Bo 1	ok1 [Co A	ompatibil E	ity Mode] B	С	D	E	F	G			
	Bo 1 2	ok1 [Co	empatibil E	ity Mode] 3	С	D	E	F	G			
	Bo 1 2 3	A	empatibil E	ity Mode] 3	С	D	E	F	G			
	Bo 1 2 3 4	A	empatibil E	ity Mode] 3	С	D	E	F	G			



Import predictions into Electronic tables program: example - Excel

• Import data from bank_predicted.txt



Part I. Data Mining. 1. Prepare data 2. Build several classifiers. Choose the most accurate one. 3. Divide dataset into training and validation datasets 4. Predict class in the validation dataset 5. Prepare output for business analysis

Import predictions into Electronic tables program: example - Excel

- Import data from bank_predicted.txt
- Save file as bank_results.xls (sample file is attached)

ext Impo	rt Wizard ·	Step 2 of 3					
'his screen ielow.	lets you set	the delimiters y	our data c	ontains. You ca	n see how your I	text is affected in I	the previe
Delimiters	olon a	Treat conserved Text gualifier:	ecutive del	limiters as one			
I <u>O</u> ther		-	,				
Data <u>P</u> revi	iew						
inst#,	actual,	predicted,	error,	probability	distributi	on	
· · ·	þ	2:NO	2:NO	0.076	0.924		
	2	1:YES	1:YES	0.938	0.063		
	3	2:N0	2:NO	0.214	0.786		
	4	2:NO	p:No	p.076	p.924		
•							Þ
			Γ	Cancel	< <u>B</u> ack	<u>N</u> ext >	<u>F</u> inist
			_				

Close WEKA

• The data mining part is complete

