COMP47460 Tutorial

Evaluation

Aonghus Lawlor
Deepak Anjwani

School of Computer Science
Autumn 2019

Tutorial Q1

- The contingency table below shows the evaluation results for a binary classifier applied to a set of 768 test examples, which are annotated with the class labels (A, B). From this table calculate:
  a) The precision score for both of the classes.
  b) The recall score for both of the classes.
  c) The F1-measure score for both of the classes.
  d) The overall classification accuracy for all the data.

Real Class \ Predicted Class	A	B
A	407	93
B	108	160

Tutorial Q1(a,b)

$$\text{Precision} = \frac{TP}{TP + FP}$$

	Predicted Pos	Predicted Neg	Real
P	TP	FN	Pos
N	FP	TN	Neg

$$\text{Recall} = \frac{TP}{TP + FN} = \text{Sensitivity}$$

	Predicted Pos	Predicted Neg	Real
P	TP	FN	Pos
N	FP	TN	Neg

Note: These measures are always relative to one class!

	Predicted Class A	Predicted Class B	Real Class
	407	93	A
	108	160	B

Class	Precision	Recall
A	407/(407+108) = 0.79	407/(407+93) = 0.814
B	160/(93+160) = 0.632	160/(108+160) = 0.597

Tutorial Q1(c)

- F1-Measure: harmonic mean of precision and recall

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Also relative to one class!

Class	Precision	Recall	F1
A	407/(407+108) = 0.79	407/(407+93) = 0.814	(2*0.79*0.814)/(0.79+0.814) = 0.802
B	160/(93+160) = 0.632	160/(108+160) = 0.597	(2*0.632*0.597)/(0.632+0.597) = 0.614

Tutorial Q1(d)

- Accuracy: Number of predictions correct / all predictions

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}$$

Predicted Class

A	B	
407	93	A
108	160	B

Real Class

Accuracy score is relative to the overall dataset, often reported as a percentage.

OVERALL ACCURACY:

(407+160)/(407+93+108+160)

= 73.8281%

Tutorial Q2

- The table below shows the true classes for 12 example emails, which are labelled as “spam” or “non-spam”. The table also reports the labels predicted by three different binary classifiers

Example	True Class Label	KNN Prediction	J48 Prediction	SVM Prediction
1	spam	spam	spam	spam
2	non-spam	non-spam	spam	non-spam
3	spam	non-spam	non-spam	spam
4	non-spam	non-spam	non-spam	non-spam
5	spam	spam	spam	spam
6	non-spam	non-spam	non-spam	non-spam
7	non-spam	spam	spam	non-spam
8	non-spam	non-spam	spam	spam
9	spam	spam	non-spam	spam
10	spam	spam	non-spam	non-spam
11	spam	non-spam	non-spam	spam
12	spam	spam	spam	spam

Tutorial Q2 (a,b)

Example	True Class Label	KNN Prediction	J48 Prediction	SVM Prediction
1	spam	spam	spam	spam
2	non-spam	non-spam	spam	non-spam
3	spam	non-spam	non-spam	spam
4	non-spam	non-spam	non-spam	non-spam
5	spam	spam	spam	spam
6	non-spam	non-spam	non-spam	non-spam
7	non-spam	spam	spam	non-spam
8	non-spam	non-spam	spam	spam
9	spam	spam	non-spam	spam
10	spam	spam	non-spam	non-spam
11	spam	non-spam	non-spam	spam
12	spam	spam	spam	spam

#Correct	9/12	5/12	10/12
Accuracy	75%	41.7%	83.3%

"Spam"	**TP**	5	3	6
	FP	1	3	1
	Precision	5/6 = 0.833	3/6 = 0.5	6/7 = 0.857

Overall Accuracy:
Number of predictions correct / all predictions

Precision for spam:
Number of correct spam predictions / all predictions of spam

SVM classifier is most accurate

SVM classifier has highest precision for spam

Tutorial Q3

- The table below shows the number of correct and incorrect predictions made by an image classifier during a 10-fold cross validation experiment, where the goal was to classify 500 images into one of three categories: {cats, dogs, people}.

Fold	Class: Cats		Class: Dogs		Class: People	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
1	82	68	82	68	164	36
2	81	69	102	48	176	24
3	99	51	97	53	160	40
4	81	69	102	48	148	52
5	94	56	99	51	148	52
6	97	53	91	59	162	38
7	81	69	94	56	148	52
8	76	74	79	71	181	19
9	76	74	97	53	160	40
10	96	54	79	71	179	21

Tutorial Q3(a)

a) What is the overall accuracy of the classifier based on the cross-validation results?

Fold	Class: Cats		Class: Dogs		Class: People		
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	**Accuracy**
1	82	68	82	68	164	36	65.6%
2	81	69	102	48	176	24	71.8%
3	99	51	97	53	160	40	71.2%
4	81	69	102	48	148	52	66.2%
5	94	56	99	51	148	52	68.2%
6	97	53	91	59	162	38	70.0%
7	81	69	94	56	148	52	64.6%
8	76	74	79	71	181	19	67.2%
9	76	74	97	53	160	40	66.6%
10	96	54	79	71	179	21	70.8%
Mean							**68.2%**

Fold 1: (82+82+164)/(82+68+82+68+164+36) = 65.6% accuracy for fold …

Overall: (65.6% + 71.8% + 71.2% + 66.2% + 68.2% + 70.0% + 64.6% + 67.2% + 66.6% + 70.8%)/10 = 68.2%

Tutorial Q3(b)

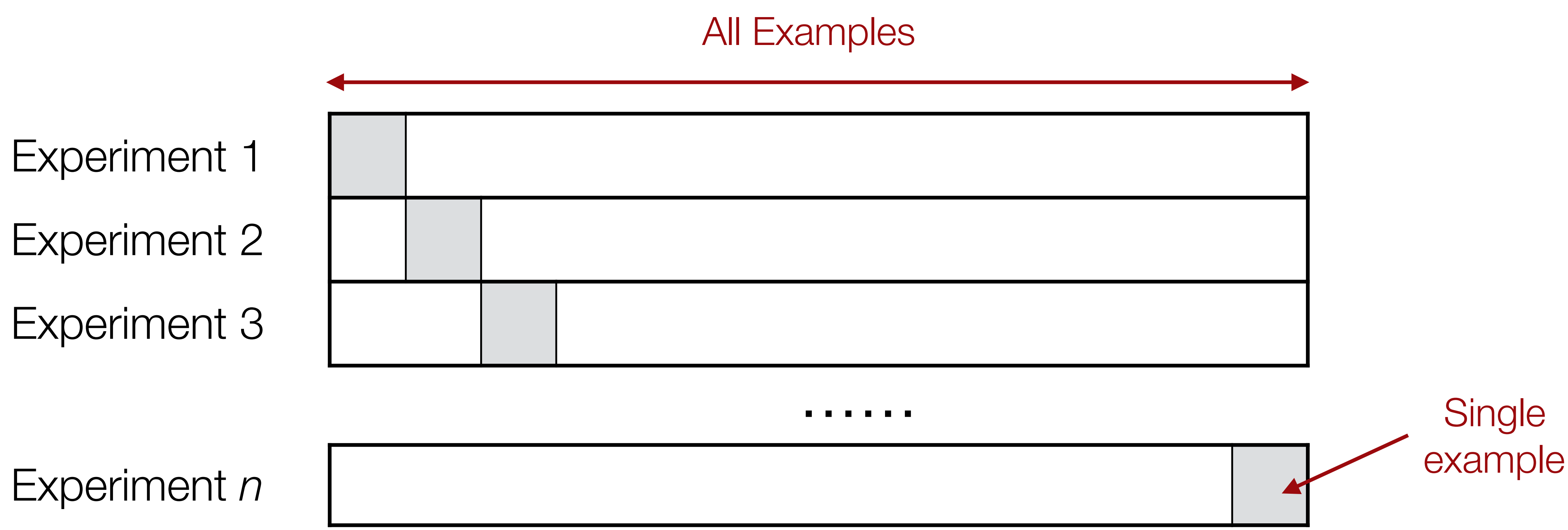
b) What conclusion might be draw about the different classes in the data, based on the results above?

Fold	Class: Cats		Class: Dogs		Class: People		
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	**Accuracy**
1	82	68	82	68	164	36	65.6%
2	81	69	102	48	176	24	71.8%
3	99	51	97	53	160	40	71.2%
4	81	69	102	48	148	52	66.2%
5	94	56	99	51	148	52	68.2%
6	97	53	91	59	162	38	70.0%
7	81	69	94	56	148	52	64.6%
8	76	74	79	71	181	19	67.2%
9	76	74	97	53	160	40	66.6%
10	96	54	79	71	179	21	70.8%
Mean	86.3	63.7	92.2	57.8	162.6	37.4	**68.2%**
Class Acc.	**57.5%**		**61.5%**		**81.3%**		

➡ High accuracy for class "People", low accuracy for "Cats" and "Dogs". Suggests system is poor at distinguishing between these classes.

Tutorial Q3(c)

c) Would "leave-one-out cross validation" be an appropriate evaluation strategy on this dataset? Justify your answer.



Dataset has n=500 examples. So leave-one-out would require running 500 experiments where 1 example is left out each time. May be computationally intractable to do this, so *k*-fold cross validation might be more suitable.