Lecture 13: Evaluation

Sarah Erfani and Karin Verspoor, CIS

Semest

Tensions in Classification

-

- f
?

- **Consistency:** is the classifier able t
all training instances?

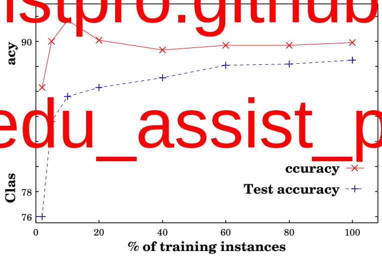
Generalisation Problem in Classification

- **Under-fitting:** model not expressive enough to capture patterns in the data.
- **Over-fitting:** model too complicated; capture noise in the data.
- **Appropriate-fitting** model captures essential patterns in the data.

Evaluating Classification

- Usually, the given data set is partitioned into two disjoint sets. The *training set* is used to build the model; the *test set* is used to validate it.
-
- Learning curves represent the performance of a fixed learning strategy over different sizes of training data, relative to a fixed evaluation metric.

Inductive Learning Hypothesis: Any hypothesis found to approximate the target function well over (a sufficiently large) training data set will also approximate the ld-out



How to evaluate a classifier?

For a two class problem:

There are Positive and Negative cases

A classifier may classify

- a Negative instance as Positive (False Positive, FP)
- a Negative instance as Negative (Tru

		Y	N
Actual	**Y**	*true positive (TP)*	*false negative (FN)*
	N	*false positive (FP)*	*true negative (TN)*

Clustering accuracy

Outlook	Temperature	Humidity	Windy	Cluster	Play
sunny	hot	high	FALSE	0	no
sunny	hot	high	TRUE	0	no
overcast	hot	high	FALSE	0	yes
		hig	FAL	1	yes
		n			yes
		n			no
					yes
		hi	FA	0	no
sunny	cool	normal	FALSE	1	
rainy	mild	normal	FALSE	1	
sunny	mild	normal	TRUE	1	
overcast	mild	high	TRUE	1	
overcast	hot	normal	FALSE	0	yes
rainy	mild	high	TRUE	0	no

Cluster 0 = “no”, Cluster 1 = “yes”

Clustering accuracy

Outlook	Temperature	Humidity	Windy	Cluster	Play
sunny	hot	high	FALSE	0	no
sunny	hot	high	TRUE	0	no
overcast	hot	high	FALSE	0	yes
rainy	mild	high	FALSE	1	yes
rainy	cool	normal	FALSE	1	yes
rainy	cool	normal	TRUE	1	no
		hi	FA	0	
		n			
overcast	mild	high	TRUE	1	
overcast	hot	normal	FALSE	0	
rainy	mild	high	TRUE	0	

Cluster 0 = “no”, Cluster 1 = “yes”

		Predicted *Y*	*N*
Actual	***Y***	*TP (7)*	*FN (2)*
	N	*FP (1)*	*TN (4)*

Accuracy

- we

$$\mathrm{ACC} = \frac{}{TP +}$$

Lecture 13: Evaluation

COMP90049 Knowledge Technologies

Evaluation

Measures

Model Validation

Results comparison

Random Baseline

Zero-R

One-R

- Alternatively, we sometimes talk about the *error rate*:

- paring the error rate ER for a given method wit method ER_0:

$$\mathrm{ERR} = \frac{\quad}{\quad 0}$$

Precision and Recall

- If we wish to know what we have positively identified **not** what we have correctly ignored (or equivalently performance relative to a single class of interest), we use *precision* and *recall*

$$\frac{TP}{\quad FP}$$

$$\frac{P}{\quad FN}$$

- Precision: Proportion of positive predictions that are correct
- Recall: Accuracy with respect to positive c
  also called true positive rate
- *Specificity* is the accuracy with resp

$$\text{Specificity} = \frac{TN}{TN + FP}$$

(sensitivity/specificity is often used in scientific applications)

Precision and Recall over Multiple Categories

- To compute an overall P/R value over multiple categories:
  1. *micro-averaging*

$$\text{Precision}_\mu = \frac{\sum_{i=1}^{c} TP_i}{\quad}$$

  2. *macro-averaging*

$$\text{Precision}_M = \frac{\quad}{\quad}$$

$$\text{Recall}_M = \frac{\quad}{c}$$

- In what situations are they the same/different?

Metrics, compared

https://en.wikipedia.org/wiki/Sensitivity_and_specificity

F-score

- In applications where we make individual decisions for each data point rather than generating a monolithic ranking , *F-score* gives us an overall picture of system performance:

c mean]

- Set β depending on how much we ca
  false positives

- Conventionally $\beta = 1$, called the

$$\text{F1-score} = 2\frac{PR}{P+R}$$

ROC and AUC

You may see people refer to AUC and ROC.

The **ROC** = Receiver Operating Characteristic

- A plot illustrating the performance of a classifier as

  (Recall/Sensitivity) vs. False Positive Rate (1−Specificity)
- The best possible prediction method would yield a point in the upper left corner or coordinate (0,1) of the ROC space, representing 100% sensitivity (no false negatives) and 100% specificity (no false positives).

AUC = Area Under the Curve

- sometimes called **AUROC**
- equal to the probability that a classifier will rank a randomly

  tance higher

  sen

- If we use all of our data to train a model, how haven't *overfit* our model to our data

Bias and Variance in Evaluation

- The (training) *bias* of a classifier is the average distance between the expected value and the estimated value
  - Bias is large if the learning method produces classifiers that are consistently wrong.
  - Bias is small if (i) the classifiers are consistently right or (ii) different

- een
  - Variance is large if different training s classifiers.
  - It is small if the training set has a minor eff decisions made, be they correct or in
  - Variance measures how inconsis they are correct or incorrect.

- The *noise* in a dataset is the inherent variability of the training data
- In evaluation, we aim to minimise classifier bias and variance (but there's not a lot we can do about noise!)

Holdout

- Train a classifier over a fixed training dataset, and evaluate it over a

- Disadvantages:
  - trade-off between more training an
  - representativeness of the training

- Perform holdout over multiple iterations, randomly selecting the

-
-
    - reduction in variance and bias over “
- Disadvantages:
    - reproducibility

Evaluation strategies: Leave-One-Out

Let us assume we have N data points for which we know the labels.

We choose each data point as test case and the rest as training data.

This means we have to train the system N times and the average performance is computed across the N predictions.

- will be unique and repeatable.
- The method also generally gives high
  all (N -1) points are used in training
  (It is typically the case that having more
  accurate classifier can be built.)

Bad point:

- It is infeasible if we have large data set and the training is itself very expensive.

Evaluation strategies: M-fold Cross-Validation

Let us assume we have N data points for which we know the labels.

This means we have to train the system M time
performance is computed across the M run

Typical values for M: 5 or 10 (i.e. 5-fold cross-
cross-validation)

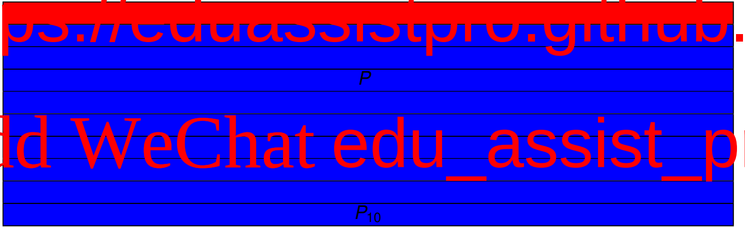
Cross Validation: Partitioning

- Split up into N equal-sized partitions P_i:

3
9
P_{10}

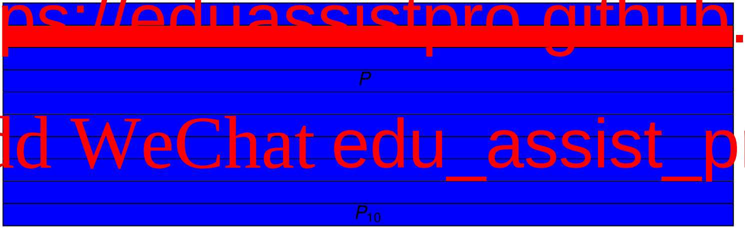
Cross Validation: Fold 1

- For each $i = 1 \ldots N$, take P_i as the test data and $\{P_j : j \neq i\}$ as the training data



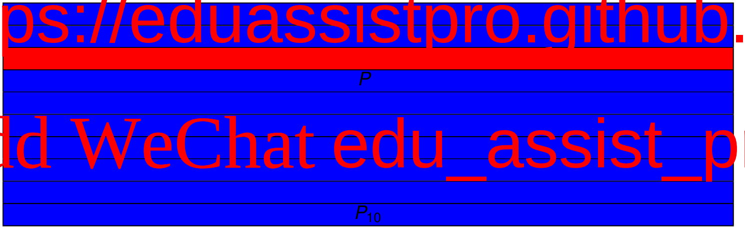
Cross Validation: Fold 2

- For each $i = 1...N$, take P_i as the test data and $\{P_j : j \neq i\}$ as the training data



Cross Validation: Fold 3

- For each $i = 1 \ldots N$, take P_i as the test data and $\{P_j : j \neq i\}$ as the training data



Cross Validation: Fold *i*

M-fold Cross-Validation Pros/Cons

Good points:

- We need to train the system only M times unlike Leave-One-Out which requires training N times.
- We can measure the stability of the system across different

- There can be a bias in evaluating the system due to sampling, how data is distributed among the M partitio
- The results will not be unique unless we identically. One solution is repeat the randomly shuffling the data M/2 times
- The results will give slightly lower accuracy values as only $\frac{M-1}{M}$ of the data is used for training.
- For small data sets it is not always possible to partition the data properly such that each partition represents the data IID (Identically Independently Distributed).

- *Baseline* = naive method which we would expect any reasonably

walk 42km

- ainst

e.g. for a marathon runner, t
run/the world record time/

- "Baseline" often used as umbrella ter

The Importance of Baselines

- Baselines are important in establishing whether any proposed method is doing better than "dumb and simple"

- In formulating a baseline, we need to be
  to the importance of positives and neg

  limited utility of a baseline of
  classification task aimed
  new diamond mines (as nearly all sites are unsuitable)

Random Baseline

Method 1: randomly assign a class to each test instance

- Often the only option in unsupervised/semi-supervised contexts

e

- Assumes we know the prior probabilit
- Alleviate effects of variance by:
  - running method N times and cal

    OR
  - arriving at a deterministic estimate of the accuracy of random assignment = $\sum_i P(C_i)^2$

Zero-R

- **Method:** classify all instances according to the most common class

-
- Inappropriate if the majority class is is to identify needles in the haystack
- For `weather.nominal`, zero-R ca

One-R (One Rule)

Creates one rule for each attribute in the training data, then selects the rule with the smallest error rate as its one rule

- **Method:** create a "decision stump" for each attribute, with branches

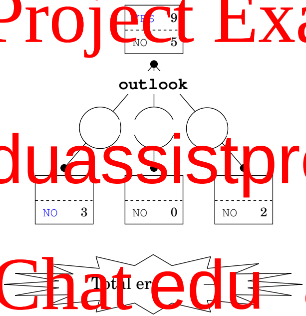
-

```
For each attribute,
  For each value of the attribute, make a rule
    1 count how often each class appears
    2 find the most frequent class
    3 make the rule assign that class to this attribute-value
  Calculate the error rate of the rules
Choose the rules with the smallest error rate
```

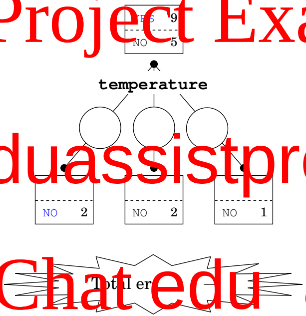
Clustering accuracy

Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
[illegible]	[illegible]	hig	FAL	[illegible]
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

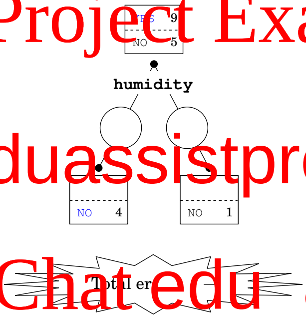
Decision Stump (`outlook`)



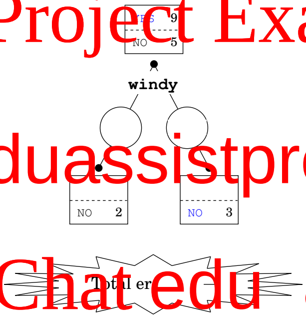
Decision Stump (`temperature`)



Decision Stump (humidity)



Decision Stump (`windy`)



One-R: Reflections

Lecture 13: Evaluation

COMP90049 Knowledge Technologies

Evaluation

Measures

Model Validation

Results comparison

Random Baseline

Zero-R

One-R

-
  -
  -
  -
-
  - unable to capture attribute interacti
  - bias towards high-arity attributes (

Summary

-

-

- What is a baseline? What are some examples of reasonable baselines to compare with?

Further Reading

- Evaluation in IR (unranked retrieval): Manning, Raghavan and Schütze Introduction to Information Retrieval, Cambridge University Press. 2008. **Section 8.** http://nlp.stanford.edu/IR-book/html/htmledition/

- to **4.6.** http://nlp.stanford.edu/IR-book/html/htmledition/the-bias-variance-tradeoff-1.html

- ROC: Tom Fawcett, “An introduction to ROC a Letters 27 (2006) https://ccrma.stanford.edu/workshops/mir2009/references/ROCintro.pdf