Data Mining: Classifier Evaluation

CSCI-B490 Seminar in Computer Science
(Data Mining)

Predictor Evaluation

1. Question:
   - how good is our algorithm?
   - how will we estimate its performance?
2. Question:
   - what is the performance measure we should use?

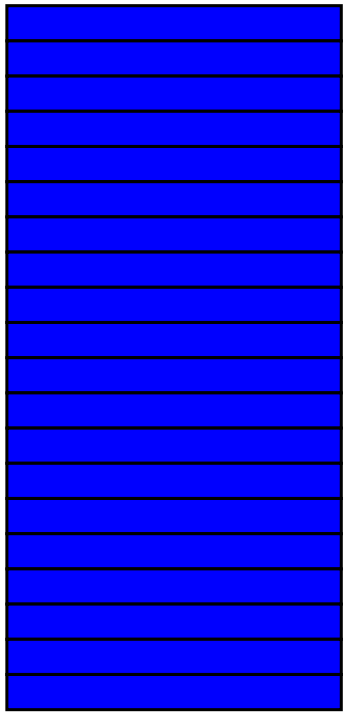
1. Classification:
   - we will use accuracy

2. Regression:
   - we will use mean square error and R^2 measure

4-Fold Cross-Validation

D

20 data points

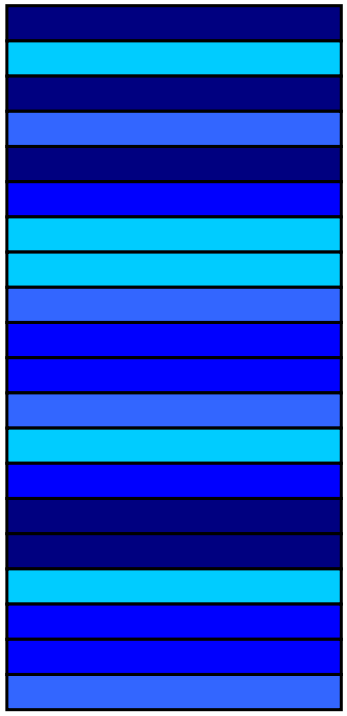


4-Fold Cross-Validation

Randomly and evenly split into 4 non-overlapping partitions

D

20 data points



How to make a random split?

Use random number generator!!!

D

5 data points

```
>> x = rand(1, 5);
>> [a b] = sort(x)
a =
    0.1576    0.4854    0.8003    0.9572    0.9706
b =
     1     4     5     3     2
```

```
>> p1 = b(1 : 2 : length(b))
p1 =
     1     5     2
```

partition 1

```
>> p2 = b(2 : 2 : length(b))
p2 =
     4     3
```

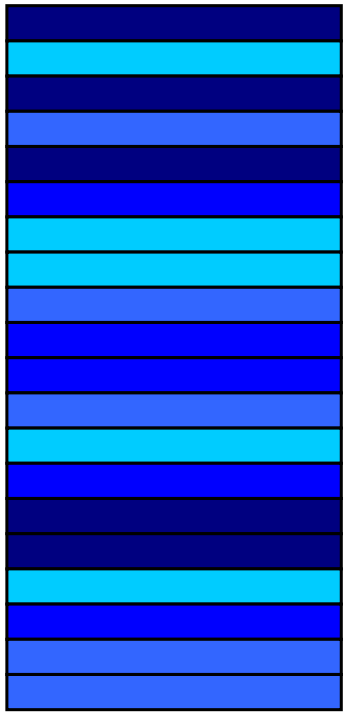
partition 2

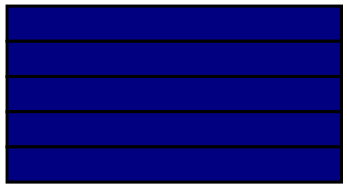
4-Fold Cross-Validation

Randomly and evenly split into 4 non-overlapping partitions

D

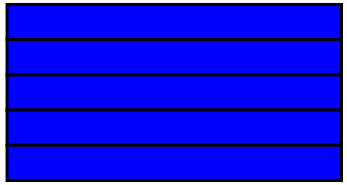
20 data points





Partition 1.

Data points: 1, 3, 5, 15, 16



Partition 2.

Data points: 6, 10, 11, 14, 17

Partition 3.

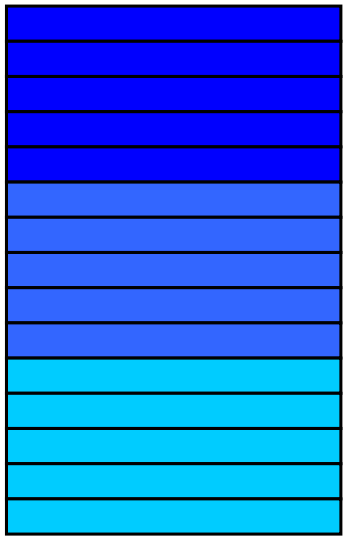
Data points: 4, 9, 12, 19, 20

Partition 4.

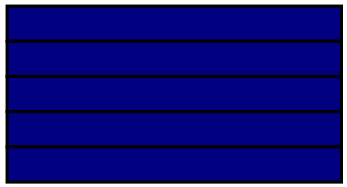
Data points: 2, 7, 8, 13, 17

4-Fold Cross-Validation

Step 1: Use partition 1 as test and partitions 2, 3 and 4 as training



Training

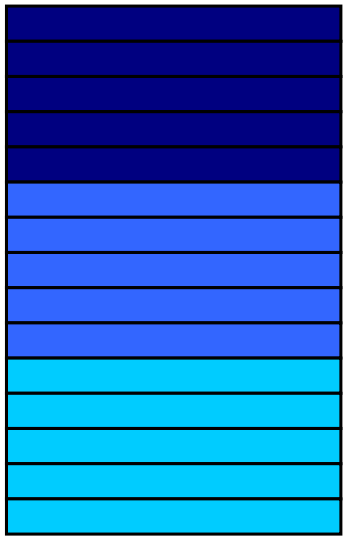


Test

- **Pretend class (target) is not known for the data points in Test.**
- **Use Training set to predict class (target) for the data points in Test**
- **Count number of misses (classification) or the error (regression)**

4-Fold Cross-Validation

Step 2: Use partition 2 as test and partitions 1, 3 and 4 as training



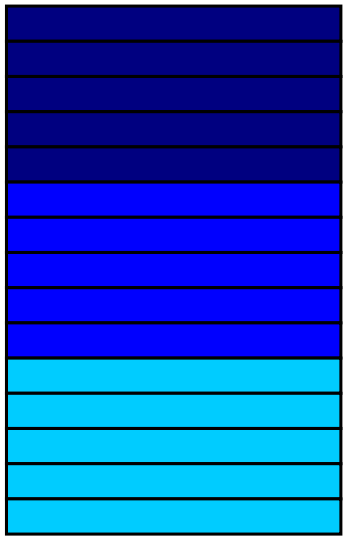
Training

Test

- **Pretend class (target) is not known for the data points in Test.**
- **Use Training set to predict class (target) for the data points in Test**
- **Count number of misses (classification) or the error (regression)**

4-Fold Cross-Validation

Step 3: Use partition 3 as test and partitions 1, 2 and 4 as training



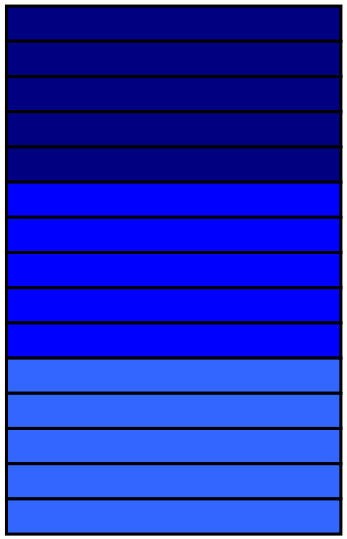
Training

Test

- **Pretend class (target) is not known for the data points in Test.**
- **Use Training set to predict class (target) for the data points in Test**
- **Count number of misses (classification) or the error (regression)**

4-Fold Cross-Validation

Step 4: Use partition 4 as test and partitions 1, 2 and 3 as training

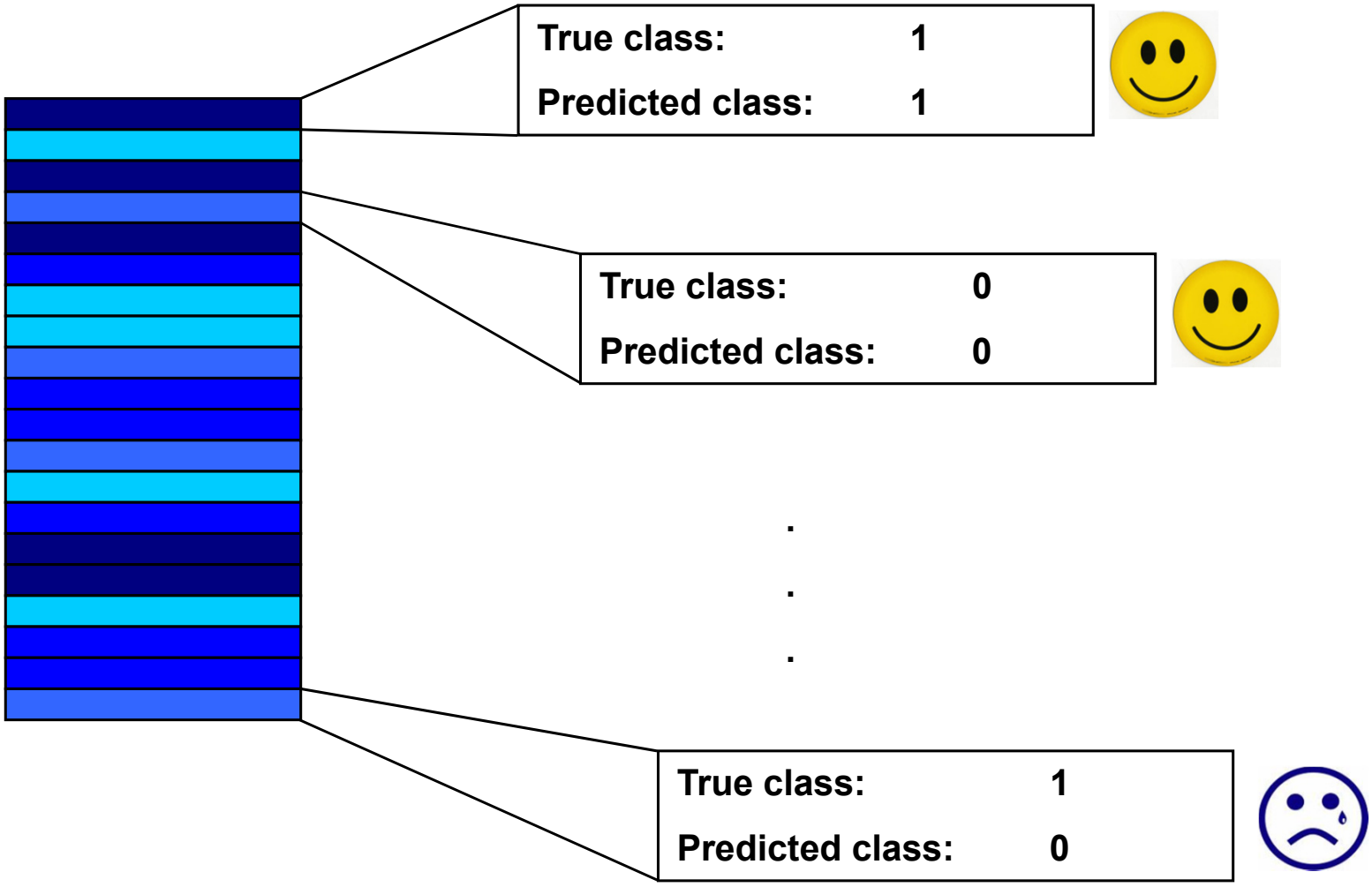


Training

Test

- **Pretend class (target) is not known for the data points in Test.**
- **Use Training set to predict class (target) for the data points in Test**
- **Count number of misses (classification) or the error (regression)**

4-Fold Cross-Validation



For each of 20 data points there is true and predicted class

Confusion Matrix (Binary Class)

Predicted class \ True class	0	1
0	N_{00}	N_{01}
1	N_{10}	N_{11}

Number of data points whose true class was 0 but predicted class was 1.

$$Accuracy = \frac{N_{00} + N_{11}}{N_{00} + N_{10} + N_{01} + N_{11}}$$

$$Error = 1 - Accuracy$$

Classification Accuracy and Error

$$Accuracy = \frac{N_{correct}}{N}$$

$N_{correct}$: number of correctly classified data points

N: total number of data points

$$Error = 1 - \frac{N_{correct}}{N}$$

Another naming convention:

N_{00} = number of true negatives

N_{01} = number of false negatives

N_{10} = number of false positives

N_{11} = number of true positives

More on Accuracy – Binary Class Case

$$sn = \frac{N_{11}}{N_{01} + N_{11}}$$ ← **Sensitivity or accuracy on data points whose class is 1. Also called true positive rate.**

$$sp = \frac{N_{00}}{N_{10} + N_{00}}$$ ← **Specificity or accuracy on data points whose class is 0. Also called true negative rate.**

$$1 - sn = 1 - \frac{N_{11}}{N_{01} + N_{11}}$$ ← **False negative rate.**

$$1 - sp = 1 - \frac{N_{00}}{N_{10} + N_{00}}$$ ← **False positive rate.**

$$Accuracy_B = \frac{sn + sp}{2}$$ ← **Balanced-sample accuracy**

More on Accuracy – Binary Class Case

$$rc = \frac{N_{11}}{N_{01} + N_{11}}$$

Recall is accuracy on data points whose class is 1. Same as sensitivity or true positive rate.

$$pr = \frac{N_{11}}{N_{10} + N_{11}}$$

Precision is accuracy on data points that were predicted as 1. Also called positive predictive value.

$$1 - rc = 1 - \frac{N_{11}}{N_{01} + N_{11}}$$

False negative rate.

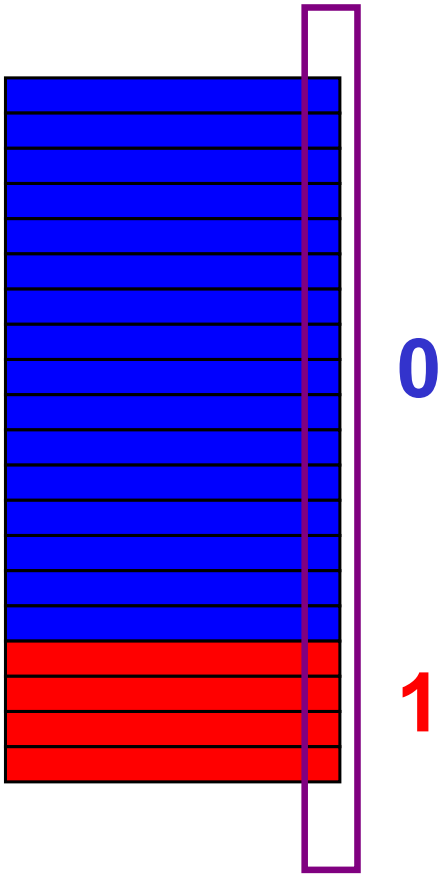
$$1 - pr = 1 - \frac{N_{11}}{N_{10} + N_{11}}$$

False discovery rate. Important: it is very different from the false positive rate!!!

$$F_\beta = (1 + \beta^2) \cdot \frac{pr \cdot rc}{\beta^2 \cdot pr + rc}$$

F-measure.

Trivial and Random Classifiers



16 data points have class 0 (majority class)

4 data points have class 1 (minority class)

Trivial classifier: always predict majority class

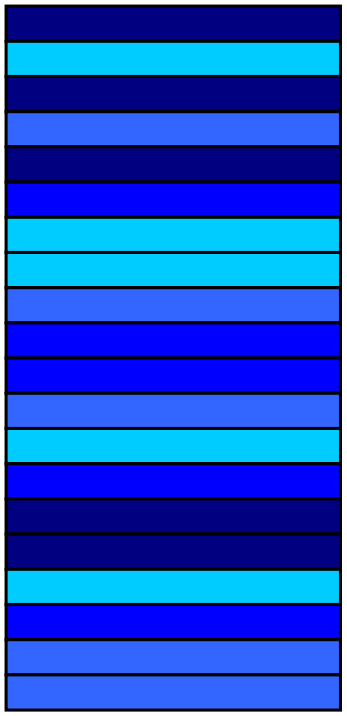
Accuracy of a trivial classifier is: 16/20 = 80%

Random classifier: predict class 0 with probability 0.8 and class 1 with probability 0.2

Accuracy of the random classifier: 68%

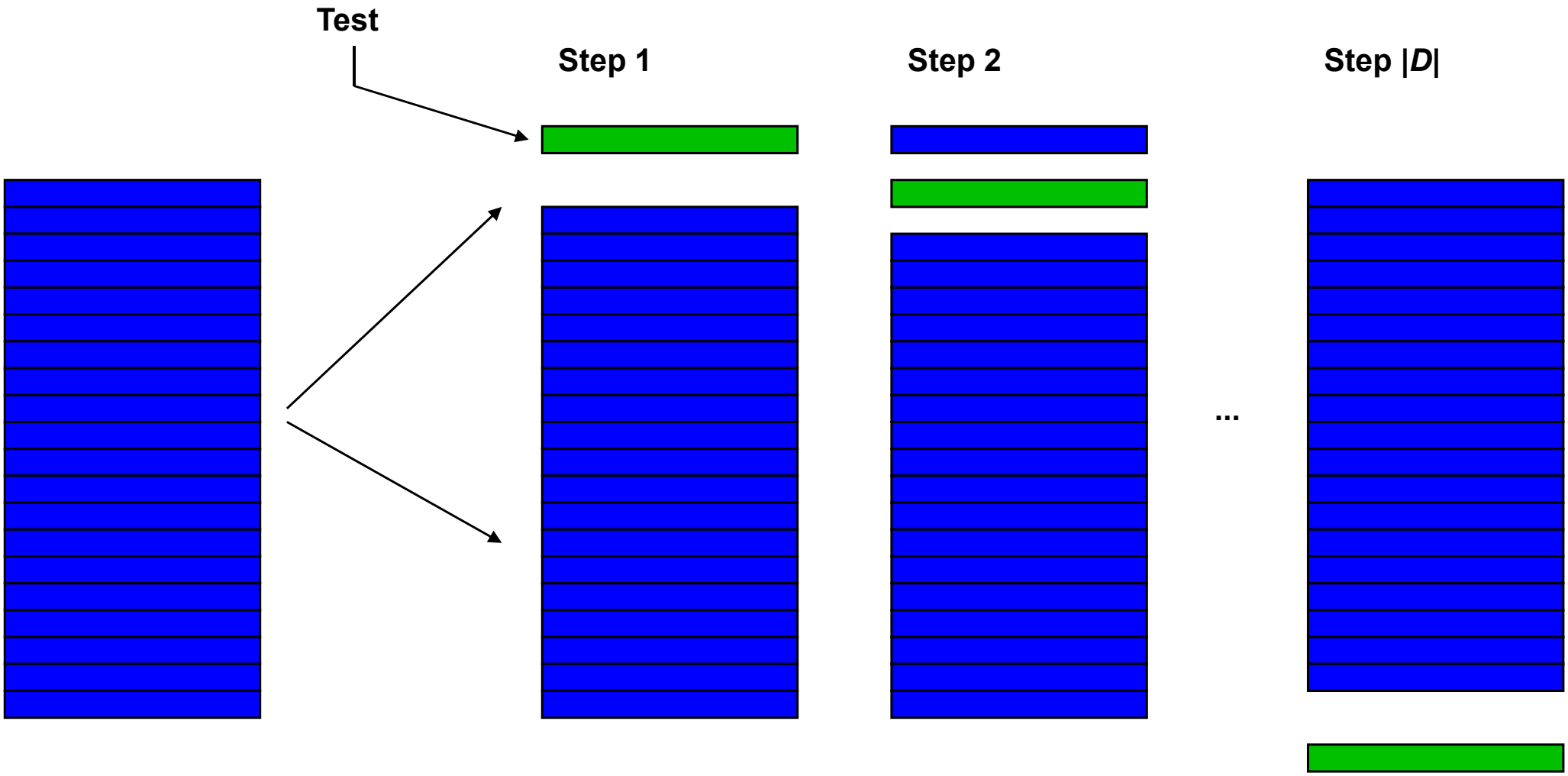
$(0.8^2 + 0.2^2 = 0.68)$

N-Fold Cross-Validation



- Is there a chance that our accuracy estimate depends on the random partition we did at the beginning?
- Yes!
- One way to overcome this is to generate several random splits into *N* folds, estimate accuracy for each fold and finally average this number.

Leave-One-Out Performance Estimation



Leave-one-out is an extreme case of cross-validation when *N* is equal to the number of data points in data set *D*.

In Short…

1. Whatever *N* you use for N-fold cross-validation, every training data point will be used exactly once as test.
2. Each data points will thus have one true value and one predicted value.
3. From these two values we compute accuracy.

4. Terminology: we always say that we are *estimating* classifier's performance. If the data set is large enough and representative this will be a good estimate. The truth is, we can only hope that our test cases will be representative and that our estimate is meaningful.

Error Bars on Accuracy Estimates

- For example, data set contains 20 data points and the confusion matrix looks like

		True class	
Predicted class		**0**	**1**
	0	**10**	**1**
	1	**4**	**5**

$$Accuracy = \frac{15}{20} = 0.75$$

$$\sigma = \sqrt{\frac{Accuracy \cdot (1 - Accuracy)}{n}} = \sqrt{\frac{0.75 \cdot 0.25}{20}} = 0.10$$

How to Report Accuracy Estimates

- For example, the estimated accuracy is 0.75
- Let's see what the estimate is for N = 10, 100, and 1000.

$$\sigma_{10} = \sqrt{\frac{0.75 \cdot 0.25}{10}} = 0.14$$

$$\sigma_{100} = \sqrt{\frac{0.75 \cdot 0.25}{100}} = 0.04$$

For N = 100 we will report accuracy as:

Accuracy = 0.75 ± 0.04

$$\sigma_{1000} = \sqrt{\frac{0.75 \cdot 0.25}{1000}} = 0.01$$