Data Mining and Machine Learning

Comp 3027J

Dr Catherine Mooney
Assistant Professor

catherine.mooney@ucd.ie

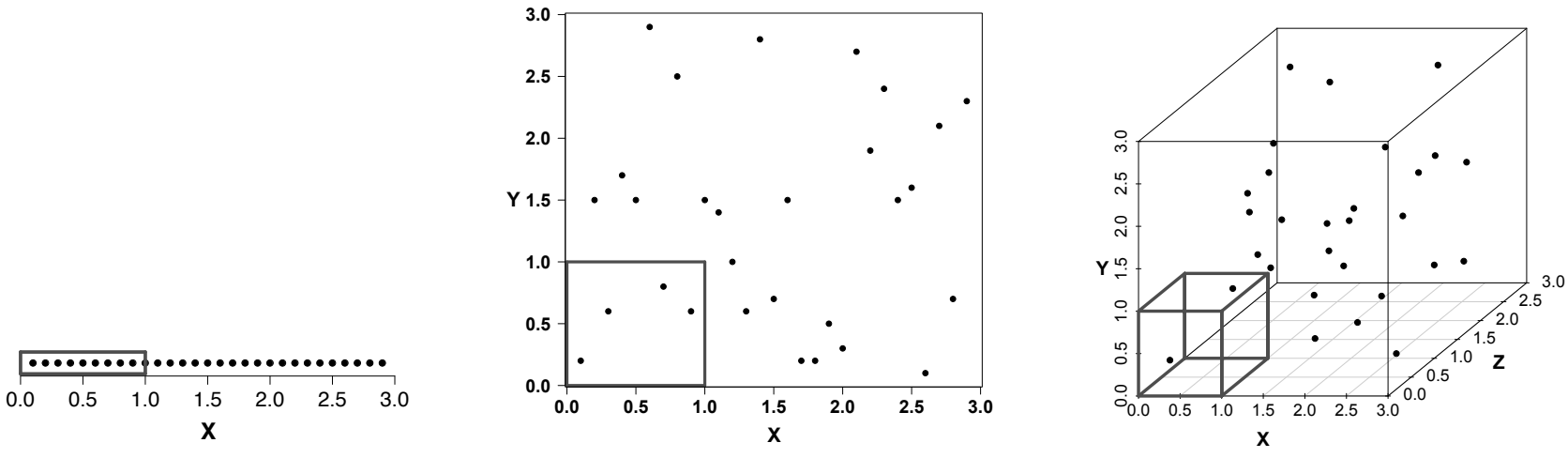
Lectures and Text

- **Core Text:**
  Fundamentals of Machine Learning for Predictive Data Analytics
  By John D. Kelleher, Brian Mac Namee and Aoife D'Arcy
- Has everyone got a copy of the book from the library?
- Last week we covered Chapter 5, sections 5.2, 5.3, 5.4.1 and 5.4.3.
- This week we will cover Chapter 5, sections 5.4.6 (Feature Selection) and Chapter 8 (Evaluation), sections 8.2, 8.3 and 8.4.1–8.4.2.3.
- Please read these sections of the book.

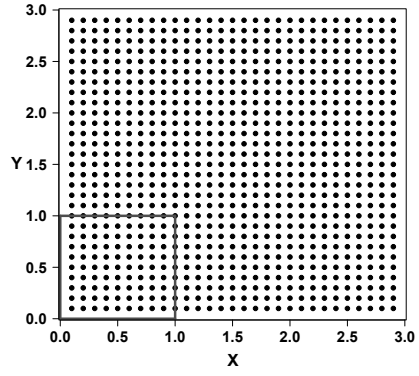
Feature Selection

- Intuitively, adding more descriptive features to a dataset provides more information about each instance and should result in more accurate predictive models.
- Surprisingly, however, the number of descriptive features in a dataset increases, there often comes a point at which continuing to add new features to the dataset results in a decrease in the predictive power of the induced models.

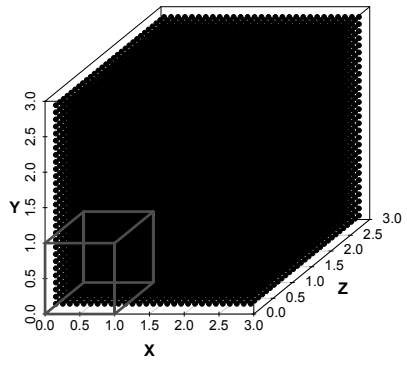
- The predictive power of an induced model depend on a reasonable sampling density
- If the sampling density is too low, then large regions of the feature space do not contain any training instances



A set of scatter plots of 29 instances illustrating **the curse of dimensionality**. Across figures (a), (b) and (c) the sampling density decreases from as the number of dimensions increases. Fig (a) sample density = 10; Fig (b) sample density = 4; Fig (c) sample density = 2.



(d)



(e)

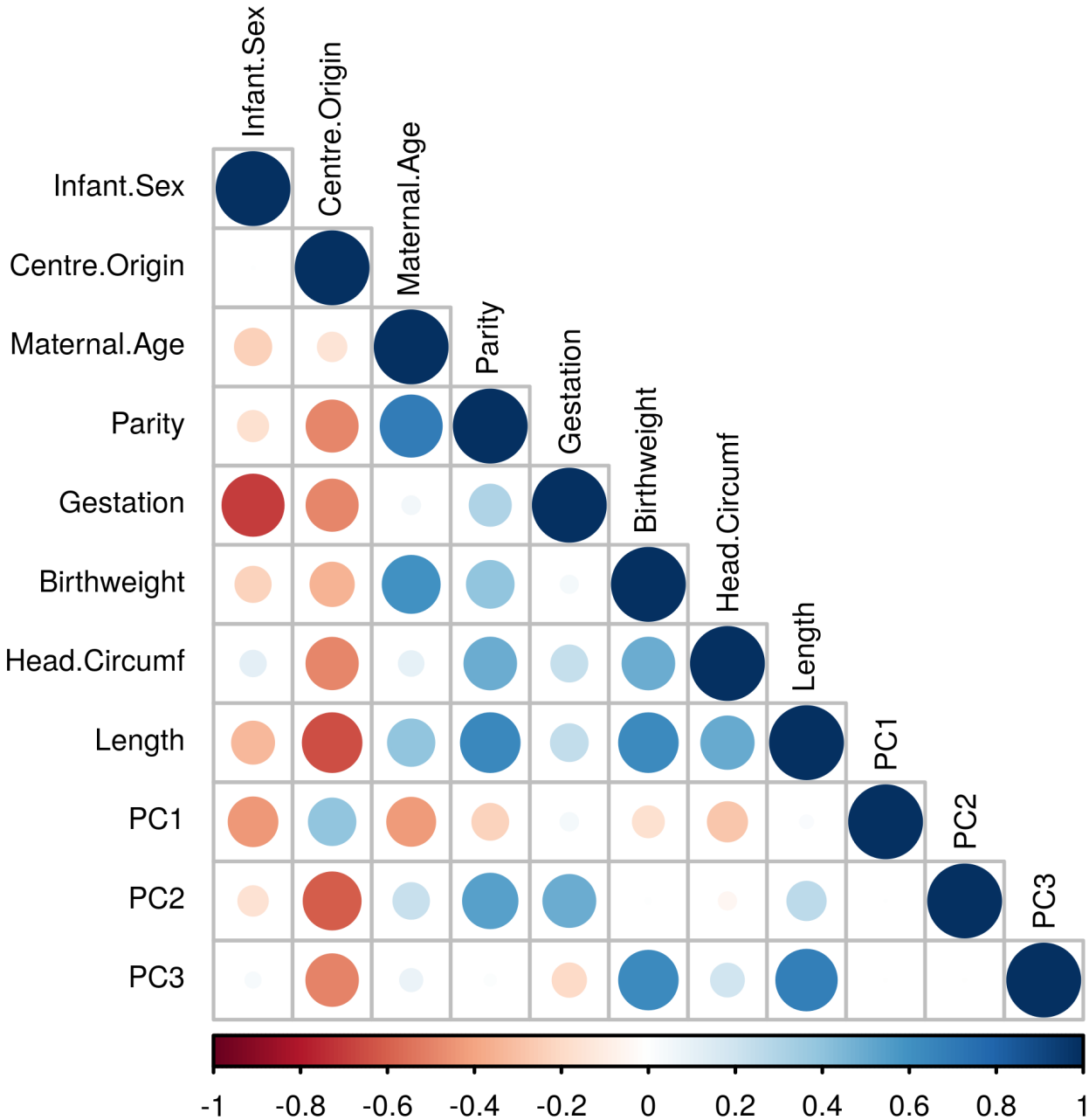
Figures (d) and (e) illustrate the cost we must incur if we wish to maintain the density of the instances in the feature space as the dimensionality of the feature space increases. To maintain the same sample density as (a) we need 841 instances in Fig (d) and 24,389 instances in Fig (e).

The trade-off between the number of descriptive features and the density of the instances in the feature space is known as **the curse of dimensionality**.

- We can use **feature selection** to help reduce the number of descriptive features in a dataset to just the subset that is most useful.
- The goal of any feature selection approach is to identify the smallest subset of descriptive features that maintains overall model performance.

- There are four classes of descriptive features:
  1. Predictive a predictive descriptive feature provides information that is useful in estimating the correct value of a target feature.
  2. Interacting by itself, an interacting descriptive feature is not informative about the value of the target feature. In conjunction with one or more other features, however, it becomes informative.
  3. Redundant a descriptive feature is redundant if it has a strong correlation with another descriptive feature.
  4. Irrelevant an irrelevant descriptive feature does not provide information that is useful in estimating the value of the target feature.

C. Correlation Matrix



Ideally, a feature selection approach will return the subset of features that includes the predictive and interacting features while excluding the irrelevant and redundant features.

- The most popular and straight forward approach to feature selection is to **rank and prune**.
  - Features are ranked using a measure of their predictiveness
  - Any feature outside the top X% of the features in the list is pruned
- Disadvantage – the predictiveness of each feature is evaluated in isolation from the other features in the dataset. This leads to the undesirable result that ranking and pruning can exclude interacting features and include redundant features
- See Lab 6 for an example of using a Random Forest for feature selection

Any questions so far?

Evaluation

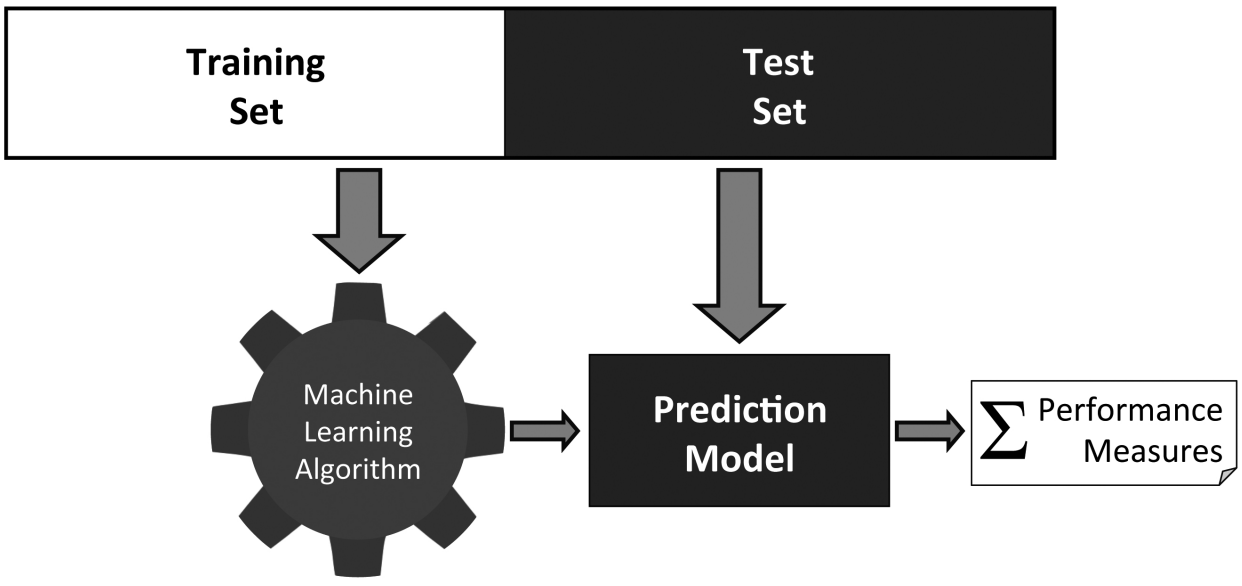
The purpose of evaluation

- To determine which model is the most suitable for a task
- To estimate how the model will perform

Designing Evaluation Experiments

Designing Evaluation Experiments

The most important part of the design of an evaluation experiment for a predictive model is ensuring that the data used to evaluate the model is not the same as the data used to train the model.



The process of building and evaluating a model using a **hold-out test set**.

k-Fold Cross Validation

k-fold cross validation

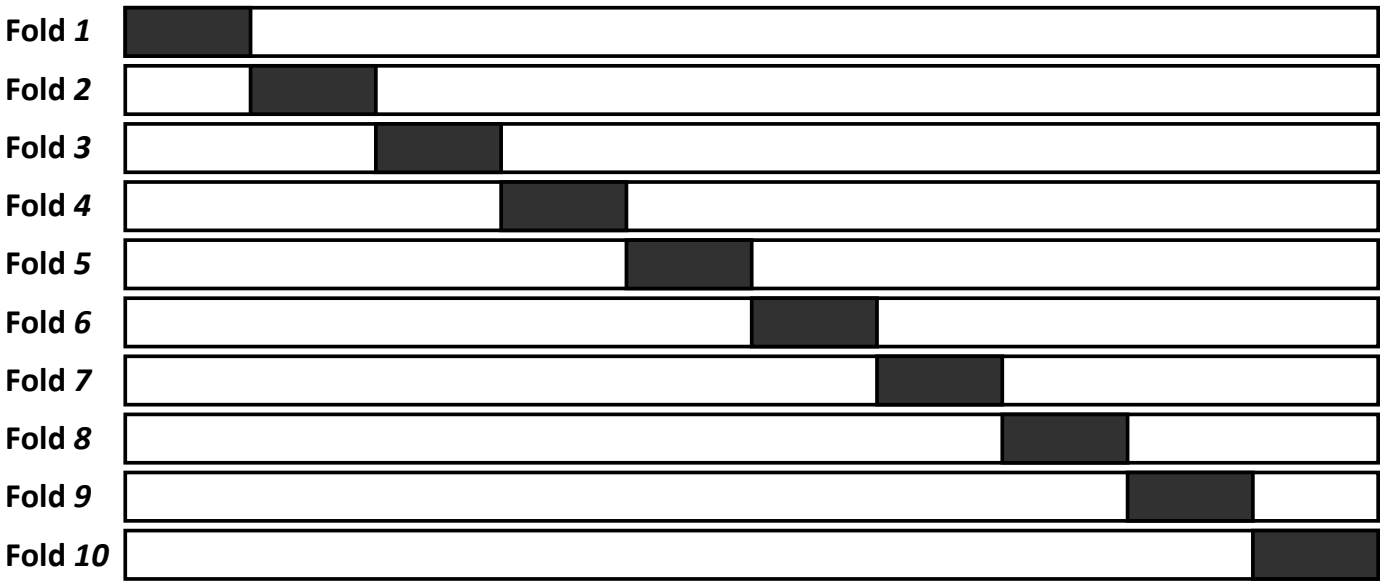
- The available data is divided into k equal-sized folds (or partitions)
- k separate evaluation experiments are performed

k-fold cross validation

- In the first evaluation experiment, the data in the 1st fold is used as the test set, and the data in the remaining k - 1 folds is used as the training set.
- A second evaluation experiment is then performed using the data in the 2nd fold as the test set and the data in the remaining k - 1 folds as the training set.
- Again the relevant performance measures are calculated on the test set and recorded.
- This process continues until k evaluation experiments have been conducted and k sets of performance measures have been recorded.
- Finally, the k sets of performance measures are aggregated to give one overall set of performance measures.

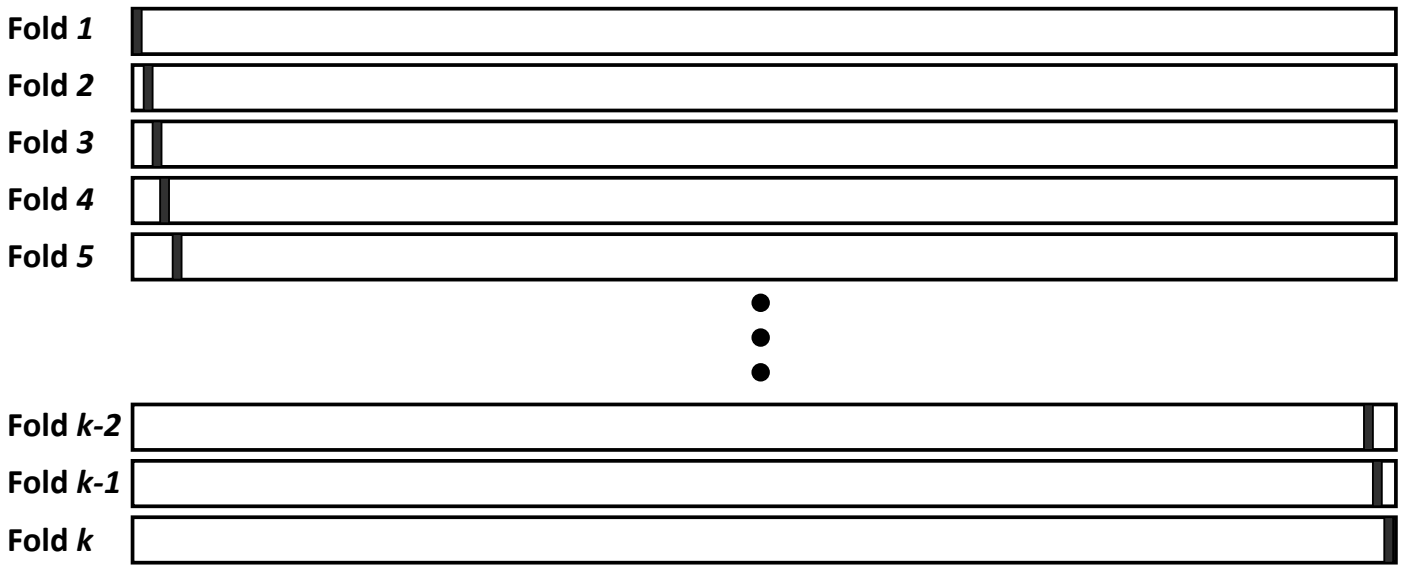
k-fold cross validation

Although k can be set to any value, 10-fold cross validation is probably the most common variant used in practice.



The division of data during the ***k*-fold cross validation** process. Black rectangles indicate test data, and white spaces indicate training data.

Fold	Confusion Matrix				Class Accuracy
1			Prediction		81%
			LATERAL	FRONTAL	
	Target	LATERAL	43	9	
		FRONTAL	10	38	
2			Prediction		88%
			LATERAL	FRONTAL	
	Target	LATERAL	46	9	
		FRONTAL	3	42	
3			Prediction		82%
			LATERAL	FRONTAL	
	Target	LATERAL	51	10	
		FRONTAL	8	31	
4			Prediction		85%
			LATERAL	FRONTAL	
	Target	LATERAL	51	8	
		FRONTAL	7	34	
5			Prediction		84%
			LATERAL	FRONTAL	
	Target	LATERAL	46	9	
		FRONTAL	7	38	
Overall			Prediction		84%
			LATERAL	FRONTAL	
	Target	LATERAL	237	45	
		FRONTAL	35	183	



The division of data during the **leave-one-out cross validation** process. Black rectangles indicate instances in the test set, and white spaces indicate training data.

Any questions so far?
5 min break...

Performance Measures

For binary prediction problems there are 4 possible outcomes

- True Positive (TP)
- True Negative (TN)
- False Positive (FP)
- False Negative (FN)

The structure of a confusion matrix.

		Prediction	
		POSITIVE	**NEGATIVE**
Target	**POSITIVE**	*TP*	*FN*
	NEGATIVE	*FP*	*TN*

ID	Target	Prediction	Outcome
1	spam	not-spam	FN
2	spam	not-spam	FN
3	not-spam	not-spam	TN
4	spam	spam	TP
5	not-spam	not-spam	TN
6	spam	spam	TP
7	not-spam	not-spam	TN
8	spam	spam	TP
9	spam	spam	TP
10	spam	spam	TP
11	not-spam	not-spam	TN
12	spam	not-spam	FN
13	not-spam	not-spam	TN
14	not-spam	not-spam	TN
15	not-spam	not-spam	TN
16	not-spam	not-spam	TN
17	not-spam	spam	FP
18	spam	spam	TP
19	not-spam	not-spam	TN
20	not-spam	spam	FP

A confusion matrix for the set of predictions shown in the previous slide

		Prediction	
		SPAM	**NOT-SPAM**
Target	**SPAM**	6	3
	NOT-SPAM	2	9

		Prediction	
		POSITIVE	**NEGATIVE**
Target	**POSITIVE**	*TP*	*FN*
	NEGATIVE	*FP*	*TN*

$$\text{misclassification rate} = \frac{\text{number incorrect predictions}}{\text{total predictions}}$$

$$\text{misclassification rate} = \frac{(FP + FN)}{(TP + TN + FP + FN)}$$

$$\text{misclassification rate} = \frac{(2 + 3)}{(6 + 9 + 2 + 3)} = 0.25$$

$$\text{classification accuracy} = \frac{\text{number correct predictions}}{\text{total predictions}}$$

$$\text{classification accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$\text{classification accuracy} = \frac{(6 + 9)}{(6 + 9 + 2 + 3)} = 0.75$$

$$\text{True Positive Rate (TPR)} = \frac{TP}{(TP + FN)}$$

$$\text{True Negative Rate (TNR)} = \frac{TN}{(TN + FP)}$$

$$\text{False Positive Rate (FPR)} = \frac{FP}{(TN + FP)}$$

$$\text{False Negative Rate (FNR)} = \frac{FN}{(TP + FN)}$$

$$\begin{aligned}
\text{TPR} &= \frac{6}{(6+3)} &= 0.667 \\
\text{TNR} &= \frac{9}{(9+2)} &= 0.818 \\
\text{FPR} &= \frac{2}{(9+2)} &= 0.182 \\
\text{FNR} &= \frac{3}{(6+3)} &= 0.333
\end{aligned}$$

$$\text{precision} = \frac{TP}{(TP + FP)}$$

$$\text{recall} = \frac{TP}{(TP + FN)}$$

		Prediction	
		POSITIVE	**NEGATIVE**
Target	**POSITIVE**	*TP*	*FN*
	NEGATIVE	*FP*	*TN*

$$\text{precision} = \frac{6}{(6+2)} = 0.75$$

$$\text{recall} = \frac{6}{(6+3)} = 0.667$$

$$F_1\text{-measure} = 2 \times \frac{(\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})}$$

$$F_1\text{-measure} = 2 \times \frac{\left(\frac{6}{(6+2)} \times \frac{6}{(6+3)}\right)}{\left(\frac{6}{(6+2)} + \frac{6}{(6+3)}\right)}$$

$$= 0.706$$

$$\text{average class accuracy} = \frac{1}{|levels(t)|} \sum_{l \in levels(t)} \text{recall}_l$$

$$\text{average class accuracy}_{\text{HM}} = \frac{1}{\frac{1}{|levels(t)|}\sum_{l \in levels(t)} \frac{1}{\text{recall}_l}}$$

Exercises

Explain the problem associated with measuring the performance of a predictive model using a single accuracy figure

A confusion matrix for a k-NN model trained on a churn prediction problem. What is the classification accuracy of this model?

		Prediction	
		NON-CHURN	**CHURN**
Target	**NON-CHURN**	90	0
	CHURN	9	1

$$\text{classification accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$\text{classification accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$\text{classification accuracy} = \frac{(90 + 1)}{(90 + 1 + 9 + 0)} = 0.91$$

A confusion matrix for a naive Bayes model trained on a churn prediction problem. What is the classification accuracy of this model?

		Prediction	
		NON-CHURN	**CHURN**
Target	**NON-CHURN**	70	20
	CHURN	2	8

$$\text{classification accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$\text{classification accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$\text{classification accuracy} = \frac{(70 + 8)}{(70 + 8 + 2 + 20)} = 0.78$$

Which model is best, the k-NN model or the naive Bayes model?

$$k\text{-NN model classification accuracy} = 0.91$$
$$\text{naive Bayes model classification accuracy} = 0.78$$

If we compare the performance of the two models only using classification accuracy then the k-NN model looks better. But is this correct?

- A single accuracy figure can hide the real performance of a model.
- This is particularly evident when we are dealing with imbalanced test datasets, as in this case.
- The *k*-NN model only predicts 1 out of the 10 CHURN instances correctly.
- The classification accuracy of 91%, is not at all an accurate reflection of the performance of the model.
- The confusion matrix for this scenario shows how poorly the model is really performing.

What happens if instead of classification accuracy we use average class accuracy?

$$\text{average class accuracy}_{\text{HM}} = \frac{1}{\frac{1}{|levels(t)|}\sum_{l \in levels(t)} \frac{1}{\text{recall}_l}}$$

average class accuracy$_{HM}$ for the k-NN model:

$$\frac{1}{\frac{1}{2}\left(\frac{1}{1.0}+\frac{1}{0.1}\right)} = \frac{1}{5.5} = 18.2\%$$

average class accuracy$_{HM}$ for the naive Bayes model:

$$\frac{1}{\frac{1}{2}\left(\frac{1}{0.778}+\frac{1}{0.800}\right)} = \frac{1}{1.268} = 78.873\%$$

- Measures such as average class accuracy or the F_1-measure attempt to address this issue.
- However, no single measure is perfect.
- Always a good idea to use multiple performance measures as part of an evaluation.

Based on the predictions made for this test set, create a confusion matrix.

ID	Target	Prediction
1	FALSE	FALSE
2	FALSE	FALSE
3	FALSE	FALSE
4	FALSE	FALSE
5	TRUE	TRUE
6	FALSE	FALSE
7	TRUE	TRUE
8	TRUE	TRUE
9	FALSE	FALSE
10	FALSE	FALSE
11	FALSE	FALSE
12	TRUE	TRUE
13	FALSE	FALSE
14	TRUE	TRUE
15	FALSE	FALSE
16	FALSE	FALSE
17	TRUE	FALSE
18	TRUE	TRUE
19	TRUE	TRUE
20	TRUE	TRUE

A confusion matrix for the set of predictions shown in the previous slide

		Prediction	
		POSITIVE	**NEGATIVE**
Target	**POSITIVE**	*TP*	*FN*
	NEGATIVE	*FP*	*TN*

		Prediction	
		SPAM	**NOT-SPAM**
Target	**SPAM**	8	1
	NOT-SPAM	0	11

Using your confusion matrix calculate the misclassification rate:

$$\text{misclassification rate} = \frac{\text{number incorrect predictions}}{\text{total predictions}}$$

$$\text{misclassification rate} = \frac{(FP + FN)}{(TP + TN + FP + FN)}$$

Then, calculate the precision, recall, and F_1-measure:

$$\text{precision} = \frac{TP}{(TP + FP)}$$

$$\text{recall} = \frac{TP}{(TP + FN)}$$

$$F_1\text{-measure} = 2 \times \frac{(\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})}$$

$$\text{misclassification rate} = \frac{(FP + FN)}{(TP + TN + FP + FN)}$$

$$\text{misclassification rate} = \frac{(0 + 1)}{(8 + 11 + 0 + 1)} = 0.05$$

$$\begin{aligned}
\text{precision} &= \frac{8}{(8+0)} && = 1.000 \\
\text{recall} &= \frac{8}{(8+1)} && = 0.889 \\
\text{F}_1\text{-measure} &= 2 \times \frac{(1.000 \times 0.889)}{(1.000 + 0.889)} \\
&= 0.941
\end{aligned}$$

Summary

Recommended Reading

- **Core Text:**
  Fundamentals of Machine Learning for Predictive Data Analytics
  By John D. Kelleher, Brian Mac Namee and Aoife D'Arcy
- This week we covered Chapter 5, sections 5.4.6 (Feature Selection) and Chapter 8 (Evaluation), sections 8.2, 8.3 and 8.4.1–8.4.2.3.
- I would suggest that you would read over these sections again.
- Email me if you have any questions and I will cover them at the beginning of class next week.
- Next week we will cover Chapter 7 – "Information-based Learning".

Review of Lab 3

Review Lab 3

- The biggest problem in Lab 3 seemed to be identifying and handling outliers.

Handling Outliers

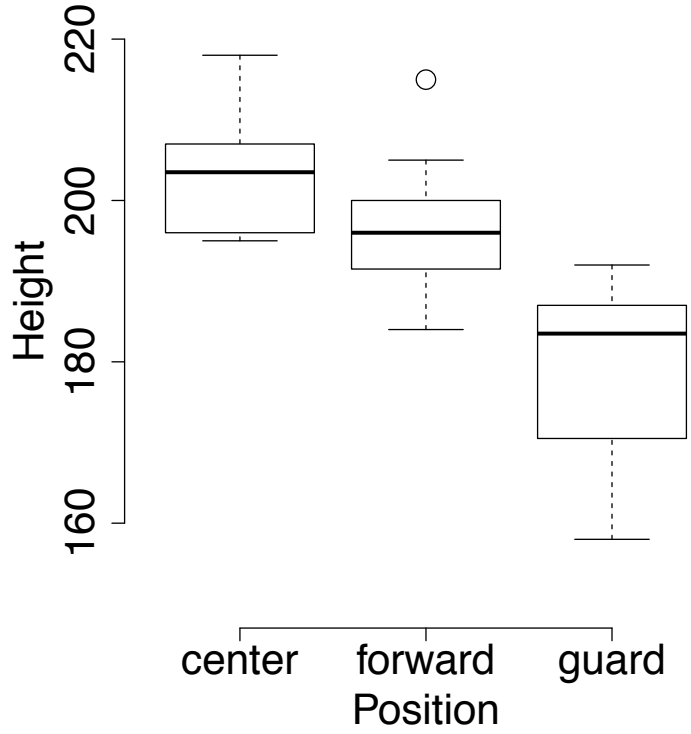
- Outliers are values that lie far away from the central tendency of a feature.
- There are two kinds of outliers that might occur in an ABT: invalid outliers and valid outliers.
- Invalid outliers are values that have been included in a sample through error and are often referred to as noise in the data.
- Valid outliers are correct values that are simply very different from the rest of the values for a feature, for example, a billionaire who has a massive salary compared to everyone else in a sample.

Identifying Outliers

- Examine the minimum and maximum values for each feature and use domain knowledge to determine whether these are plausible values.
- Invalid outliers and should immediately be either corrected, if data sources allow this, or removed and marked as missing values if correction is not possible.
- In some cases we might even remove a complete instance from a dataset based on the presence of an outlier.

Identifying Outliers

- Compare the gaps between the median, minimum, maximum, 1st quartile, and 3rd quartile values.
- Box plots can help to make this comparison.
- Exponential or skewed distributions in histograms are also good indicators of the presence of outliers.


Height
160
180
200
220
center
forward
guard
Position

Handling Outliers

- The easiest way to handle outliers is to use a **clamp transformation** that clamps all values above an upper threshold and below a lower threshold to these threshold values, thus removing the offending outliers

$$a_i = \begin{cases} lower & \text{if } a_i < lower \\ upper & \text{if } a_i > upper \\ a_i & otherwise \end{cases} \quad (1)$$

where a_i is a specific value of feature a, and *lower* and *upper* are the lower and upper thresholds.

Preview of Lab 4

Preview of Lab 4

- In Lab 4 we will be using R for Data Exploration and Similarity-based Learning (KNN)
- We will also use a number of different Performance Measures to evaluate your KNN
- You should download the following packages in advance:
  - install.packages("class")
  - install.packages("e1071")
  - install.packages("caret")

Data Mining and Machine Learning Assignment Part 1

Any questions on Assignment Part 1?

Emails

If you email me:

- Put COMP3027J in the subject line
- Use your UCD email

Otherwise I may not respond

Emails

- I will answer questions about the Assignment during lectures and labs in front of the whole class
- I will not give private responses to questions, this is unfair to the rest of the class

- **Weighting:** N/A
- **Due Date:** Friday, March 29, 4.30 pm
- **Method of Submission:** Printed report handed into BDIC academic affairs office and zip file of report (pdf) plus code (.r file) to Moodle

Pima Indians Diabetes Database

- **Number of Instances:** 768
- **Number of Attributes:** 8 plus class
- **For Each Attribute:** (all numeric-valued)
  - **Pregnancies:** Number of times pregnant
  - **Glucose:** Plasma glucose concentration
  - **BloodPressure:** Diastolic blood pressure
  - **SkinThickness:** Triceps skin fold thickness
  - **Insulin:** 2-Hour serum insulin
  - **BMI:** Body mass index
  - **DiabetesPedigreeFunction:** Diabetes pedigree function
  - **Age:** Age
  - **Outcome:** Class variable (0 or 1)
- **Missing Attribute Values:** Yes

Task – Using the Pima Indians Diabetes Database you will:

1. Design an **Analytics Base Table** (See lecture 2)
2. Create a **Data Quality Report**
   - Identify and handle any data quality issues e.g. missing values, irregular cardinality or outliers.
   - Note: some of the variables have 0 values but it is not possible for someone's BMI or blood-pressure be 0.What are you going to do about this? (See lecture 2)
3. **Feature Selection**
   - Visualize the relationships between the features (see lecture 3)
   - Are the features predictive, interacting, redundant or irrelevant? (see lecture 5)
4. **Data preparation**
   - Normalization, binning, sampling, etc. (see lecture 2, 3 and 4)
   - Note: There are nearly double the number of observations with class 0 than there are with class 1. Should you be concerned about this?

Deliverables

1. A printed report handed into the BDIC academic affairs office before the deadline. This should be a clearly written report detailing how you carried out each of the task above and showing the results that you got.
   - Font: Times New Roman
   - Font size: 12
   - 1.5 line spacing
   - Use clear headings for each task (1 – 4)
   - Include tables and figures as appropriate (Use captions i.e. Table 1 followed by a description of Table 1, and then you can refer to Table 1 in you text.)
   - There is no page limit, use as much space as you need to clearly report your findings but please keep the report short and to the point. There are no marks for waffle!
   - Your report should be presented as professionally as possible. Spelling, grammar and formatting all count.
   - Include a copy of your R code in an appendix at the end
   - Include a coversheet with your name, email, UCD and BJUT student numbers and the title “Comp3027J Data Mining and Machine Learning Assignment”

Deliverables

2. A zip file submitted to the moodle before the deadline. The zip file should be made up of:
    1. A commented R file named “ml-assignment.r”. This should have all the code you used to perform the tasks 1 – 4 above. Please keep the code clear and simple. Use the examples from the labs.
    2. A pdf of your report