Introduction to Machine Learning part 2

Martha Lewis

Department of Engineering Mathematics
University of Bristol
martha.lewis@bristol.ac.uk

Outline

- Performance metrics
- Generalisation and overfitting
- Train, validate, test

Performance Metrics: Classification

- True Positives (TP): The class is + and the prediction is +
- True Negatives (TN): The class is - and the prediction is -
- False Positive (FP): The class is - and the prediction is +
- False Negative (FN): The class is + and the prediction is -

Predicted

Actual		**Dog**	**Cat**
	Dog	TP	FN
	Cat	FP	TN

Accuracy

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

$$acc = \frac{\text{n° correctly classified}}{n}$$

Predicted

Actual

	Dog	Cat
Dog	TP	FN
Cat	FP	TN

Precision

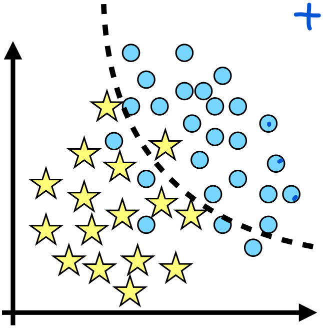
- $Precision = \frac{TP}{TP+FP}$

$$\frac{21}{21+1} = \frac{21}{22}$$

Predicted

Actual

	Dog	Cat
Dog	TP	FN
Cat	FP	TN



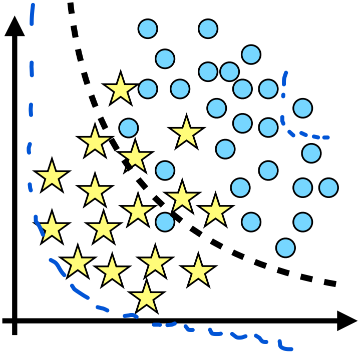
Recall

- $Recall = \frac{TP}{TP+FN}$

$\frac{25}{26}$

Predicted

Actual		Dog	Cat
	Dog	TP	FN
	Cat	FP	TN



F_1-score

The F_1 score is the harmonic mean of precision and recall

$$\begin{aligned} F_1 &= \frac{1}{\mathit{precision}^{-1} + \mathit{recall}^{-1}} \\ &= 2\frac{\mathit{precision} \cdot \mathit{recall}}{\mathit{precision} + \mathit{recall}} \end{aligned}$$

Metrics for more than two classes

$$Accuracy = \frac{\sum_{i=1}^{n} 1(y_i = f(\vec{x}_i))}{n} = \frac{\text{Correct class}^{n}}{n}$$

- This is the same as the binary case, even though the equation looks different

		Predicted		
		Dog	**Cat**	**Bird**
Actual	**Dog**	10	2	0
	Cat	1	5	1
	Bird	2	0	12

$10 + 5 + 12 = 27$

$acc = \frac{27}{33}$

Metrics for more than two classes

- Macro-averaging:
  $Precision = \frac{1}{k}\sum_{j=1}^{k} Precision_j \quad Recall = \frac{1}{k}\sum_{j=1}^{k} Recall_j$

Predicted (columns), Actual (rows)

	Dog	**Cat**	**Bird**
Dog	$\boxed{10}$	2	0
Cat	1	5	1
Bird	2	0	12

$precision_{dog} = \frac{10}{13}$

Predicted (columns), Actual (rows)

	Dog	**Cat**	**Bird**
Dog	$\boxed{10}$	2	0
Cat	1	5	1
Bird	2	0	12

$Recall_{dog} = \frac{10}{12}$

Metrics for more than two classes

- Micro-averaging

Predicted

TP

Actual	Dog	Cat	Bird
Dog	10	2	0
Cat	1	5	1
Bird	2	0	12

FN

FP

TN

Predicted

Actual	D	non D
D	10	2
non D	3	18

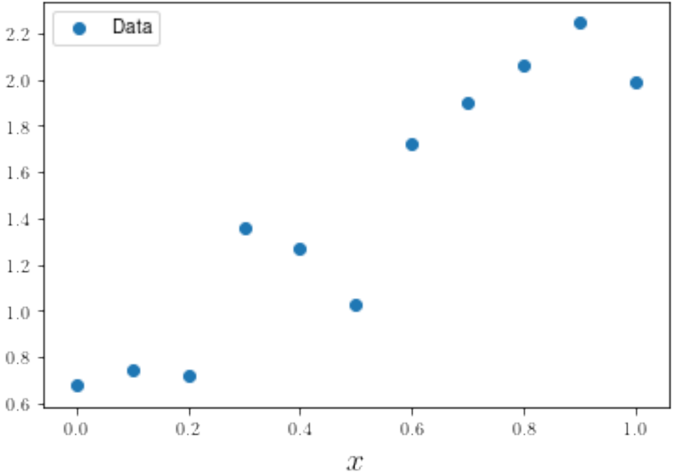
Predicted

	C	non C
C	5	2
non C	2	24

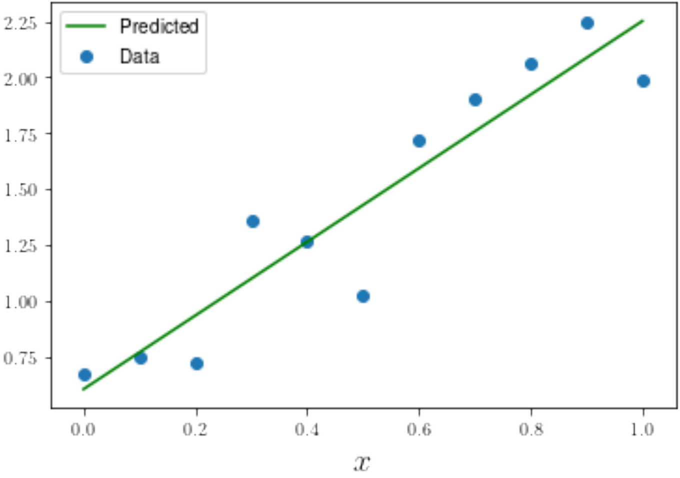
Actual	B	non B
B	12	2
non B	1	18

	+	-
+	27	6
-	6	60

Performance Metrics: Regression



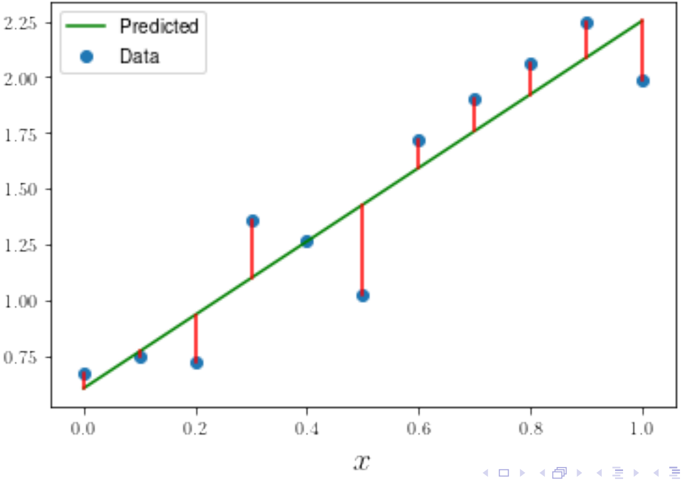
Performance Metrics: Regression



Performance Metrics: Regression

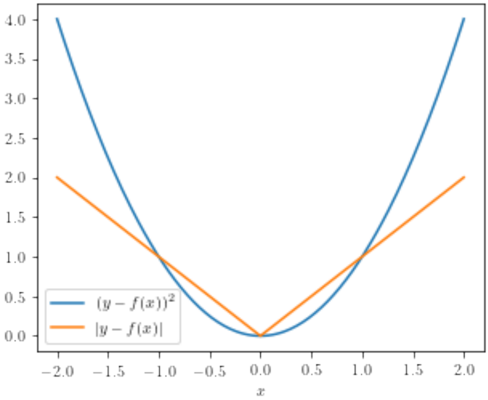
- Mean Squared Error (MSE):

$$E = \frac{1}{N} \sum_{(\vec{x},y)} (y - f(\vec{x}))^2$$



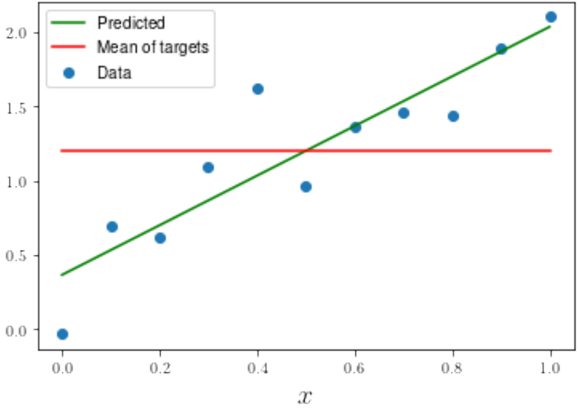
Performance Metrics: Regression

- Root Mean Squared Error (RMSE):
  $E = \sqrt{\frac{1}{N} \sum_{(\vec{x},y)} (y - f(\vec{x}))^2}$
- Mean Absolute Error (MAE): $E = \frac{1}{N} \sum_{(\vec{x},y)} \underline{|y - f(\vec{x})|}$



Performance Metrics: Regression

- $R^2 = 1 - \frac{\sum_{(\vec{x},y)}(y - f(\vec{x}))^2}{\sum_{(\vec{x},y)}(y - \bar{y})^2}$
  where $\bar{y}$ is the mean of the target values y

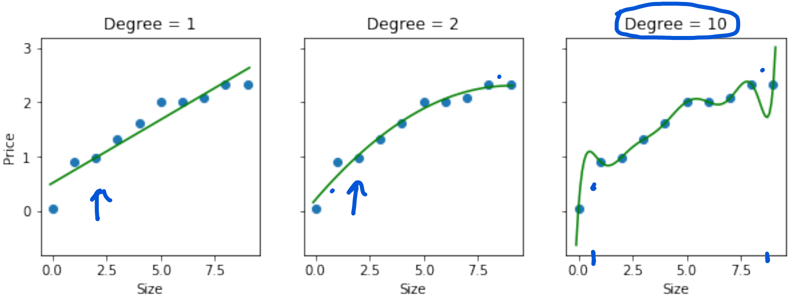


Summary: Performance Metrics

- For classification problems, we can use metrics such as accuracy, precision, recall, or F_1 score.
- These can be used for multi-class problems as well as binary problems
- Mean squared error or absolute error can be used for regression problems
- R^2 is often used for linear regression

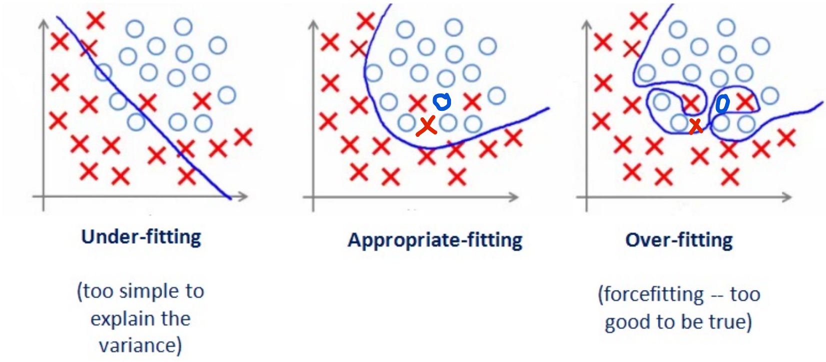
Generalisation

- By fitting an approximation function with a high number of parameters it is possible to obtain very high accuracy for the data on which you train.
- However, this learnt approximation may not then perform well on different, previously unseen, data.
- This is called over-fitting



Over-Fitting in Classification

- For classification problems we must be careful how we model the boundary between different classes in the feature space.

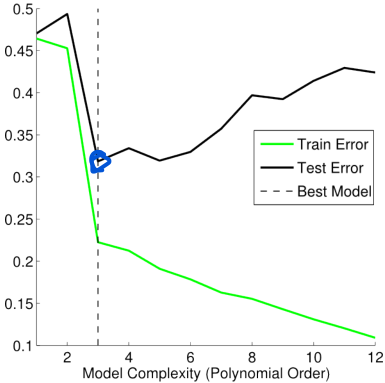


Summary: Generalisation and overfitting

- As well as attaining good scores onour training data, we want our algorithms to generalise well to unseen data
- Both regression and classification problems can overfit to data

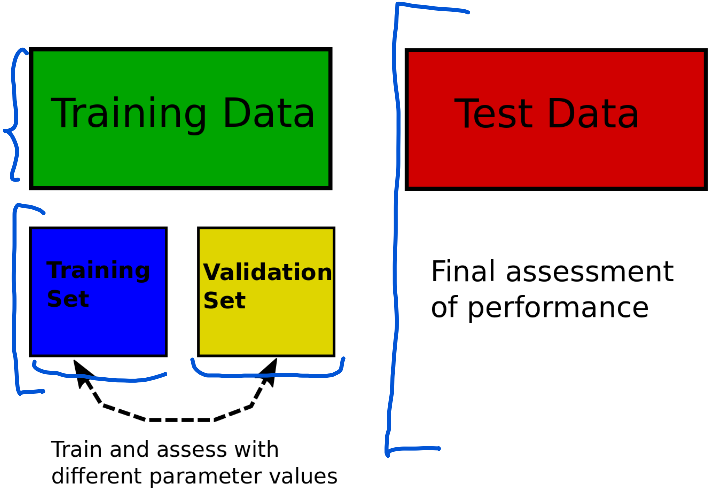
Train and Test Data

- To evaluate the generalisation of a model available data should be divided into training and test sets.



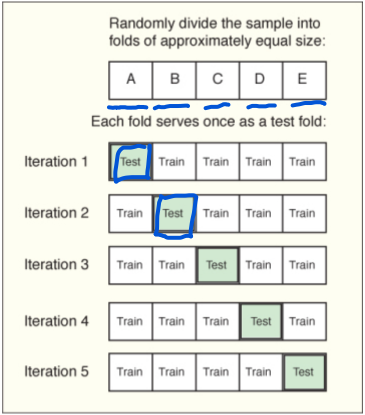
Train, Validation and Test Data

- To fit model parameters the training data is sometimes further divided into train and validation sets.



n-Folds

- How do you know that you have not just been lucky in your choice of training and test data?
- Answer: Repeat many times with different divisions into training and test.



Cross-validation can be used with the validation set.

- Splitting data into training and test sets helps ensure that our model can generalise.
- We can further split our training set into training and validation sets, to test the performance of our model on unseen data
- Cross-validation can give a more rounded view of performance

Overall summary

We have looked at:

- Performance metrics
- Generalization and overfitting
- Training and test splits

You can practice working with these concepts in the...

Worksheet

- Available as pdf and also as jupyter notebook
- Covers evaluation metrics, generalisation, overfitting
- If you don't already know Python, please do work through the Introduction to Python available on BlackBoard
- You can ask questions in the lecture or in problem classes

Next time

Simple supervised learning algorithms

- k-Nearest neighbours
- Linear regression
- Naive Bayes classifier