DALT7011_S1
Intro to Machine Learning

OXFORD BROOKES UNIVERSITY

Lecture 2: K-NN and Model Evaluation

Inna Skarga-Bandurova
iskarga-bandurova@brookes.ac.uk

Slides based on

- Dr. Matthias Rolf (Oxford Brookes University)

LECTURE PLAN

1. Foundations of ML
2. K-nearest neighbour (K-NN)
3. Linear regression & non-linear features
4. Neural Networks (NN)
5. Q&A
6. Preprocessing & dimensionality reduction
7. Case Study 1: Spam detection
8. Decision trees, forests (+ Clustering)
9. Deep learning
10. Q/A
11. Reinforcement learning
12. Case Study 2: AlphaGO

TUTORIALS

1. Linear regression
2. A self-learning tutorial on R
3. K-NN
4. NN
5. CW
6. PCA
7. CW
8. CW
9. CW
10. CW
11. CW
12. CW

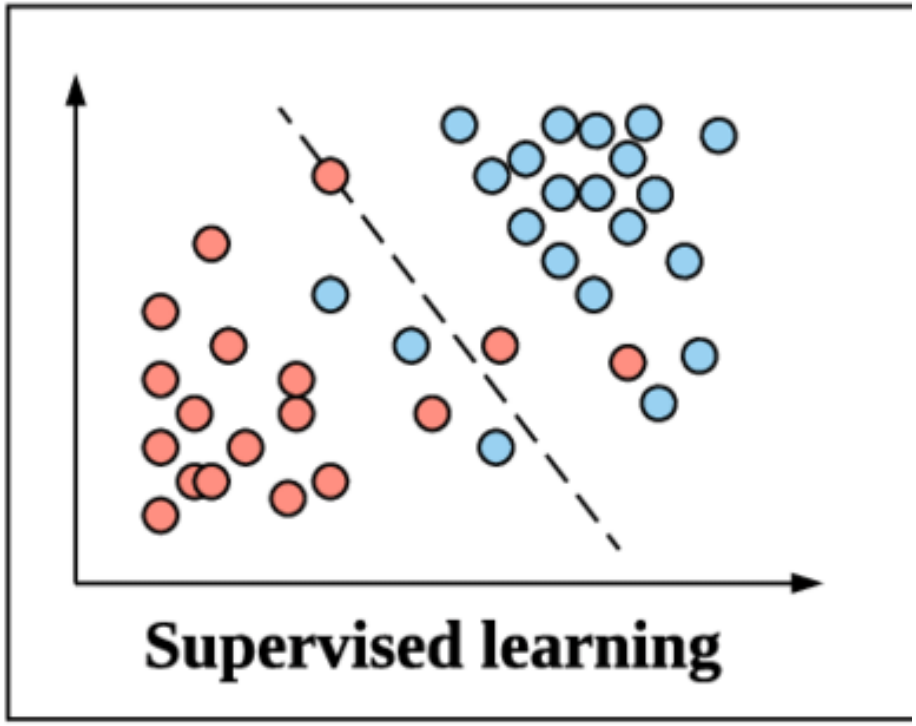
OXFORD BROOKES UNIVERSITY

Today's Lecture

- **k-Nearest Neighbours**

- **Model evaluation**
- Test Error
- Over- and Underfitting
- K-fold cross validation
- Validation and testing

SUPERVISED LEARNING



This is a very general learning paradigm applicable to a large class of decision making applications

Supervised learning is one very general group of machine learning problems

Given a vector of training data

$x = (x1 , . . . , x_N)$,

and a corresponding set of target output values

$t = (t1 , . . . , tN)$ (the “supervision”)

learn a function $y(x) = t$.

Given a new test input $\hat{x}$, the function $y(\cdot)$ predicts the output value $\hat{t}$.

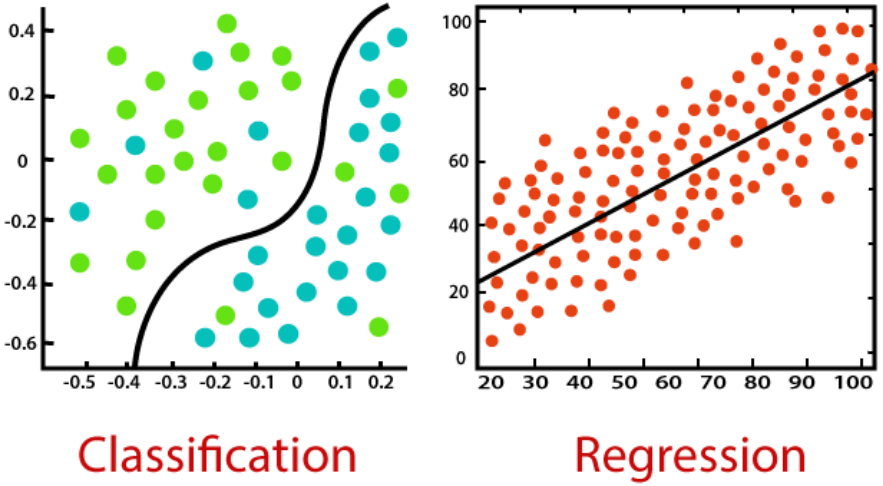
The inputs x are usually D dimensional vectors, its components are referred to as features or attributes

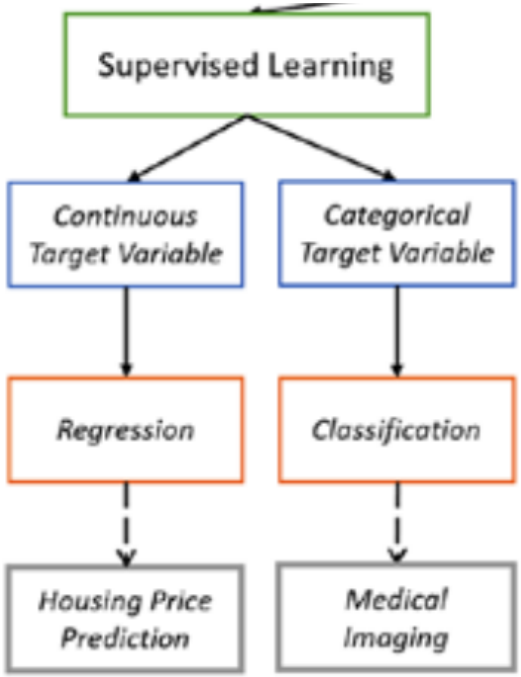
These features describe the properties of the input which are thought to correlate with the output, for example the pixels in an image of a face

OXFORD BROOKES UNIVERSITY

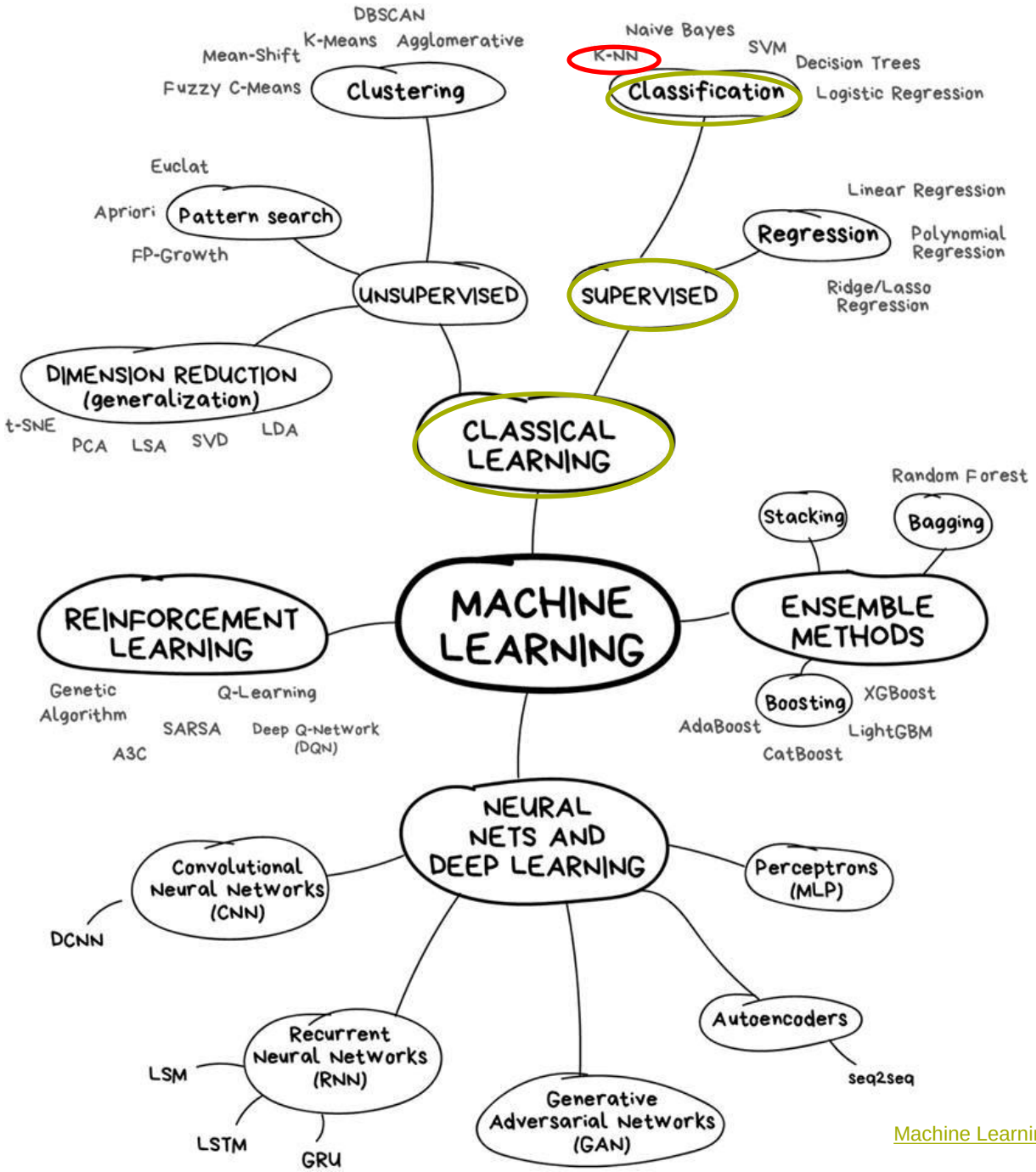
SUPERVISED LEARNING

- Regression
- Classification





Classification is about predicting a discrete class label and regression is about predicting a quantity

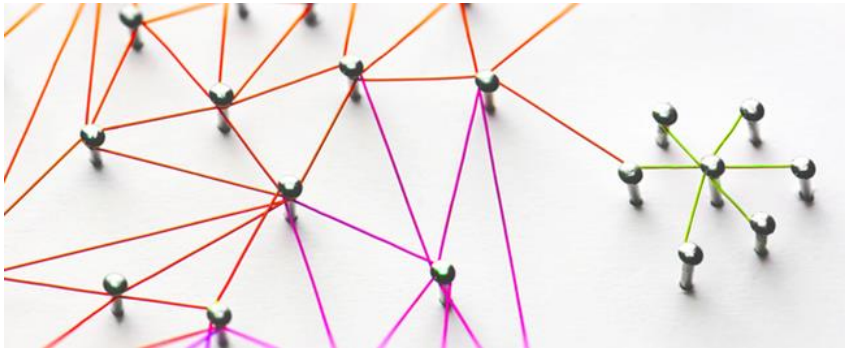


Machine Learning for Everyone In simple words.

OXFORD BROOKES UNIVERSITY

K-NEAREST NEIGHBOURS

- A powerful classification algorithm used in pattern recognition
- Stores all available cases and classifiers new cases based on similarity measure (e.g. **distance function**)
- A non-parametric lazy learning algorithm (an instance based learning method)
- Classify by majority vote of k-nearest neighbours



WHAT IS KNN?

scans	x						y
	2.3	5.9	...	11.0	-0.3	8.9	0
	-8.5	-1.7	...	-1.3	9.0	7.2	1
	-0.4	6.7	...	-2.4	4.7	-7.3	0
	1.6	-0.4	...	-4.6	6.4	1.9	1
	3.9	-4.1	...	6.7	-3.1	2.1	1
	5.1	3.7	...	1.8	-4.2	9.3	1

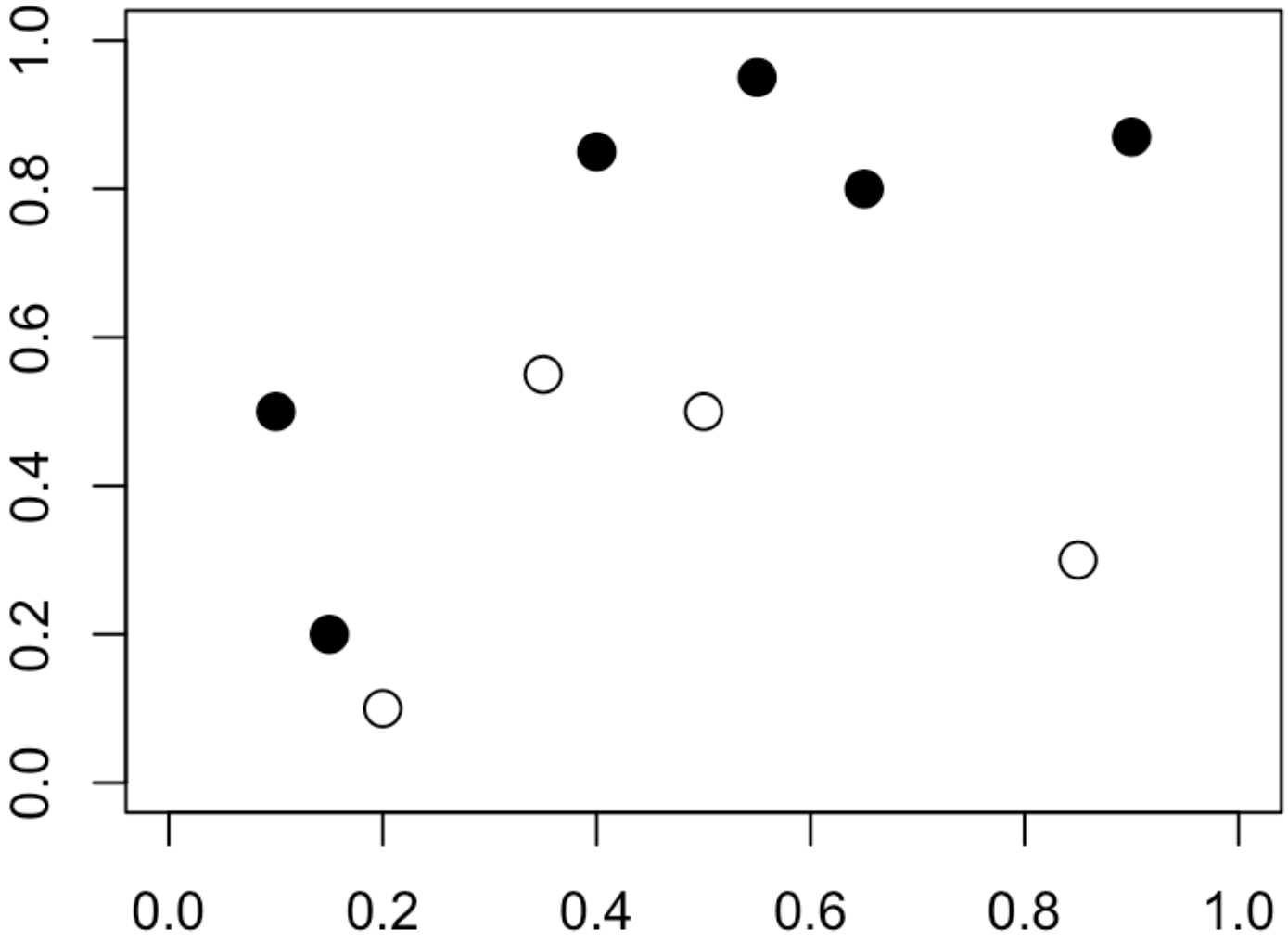
OXFORD BROOKES UNIVERSITY

- All the instances correspond to points in an n-dimensional feature space
- Each instance is represented with a set of numerical attributes
- Each of the training data consists of a set of vectors and a class label associated with each vector
- Classification is done by comparing feature vectors of different K nearest points

1) Calculate distance
2) Select the k-nearest examples to E in the training set
3) Assign E to the most common class among its K-nearest neigbours

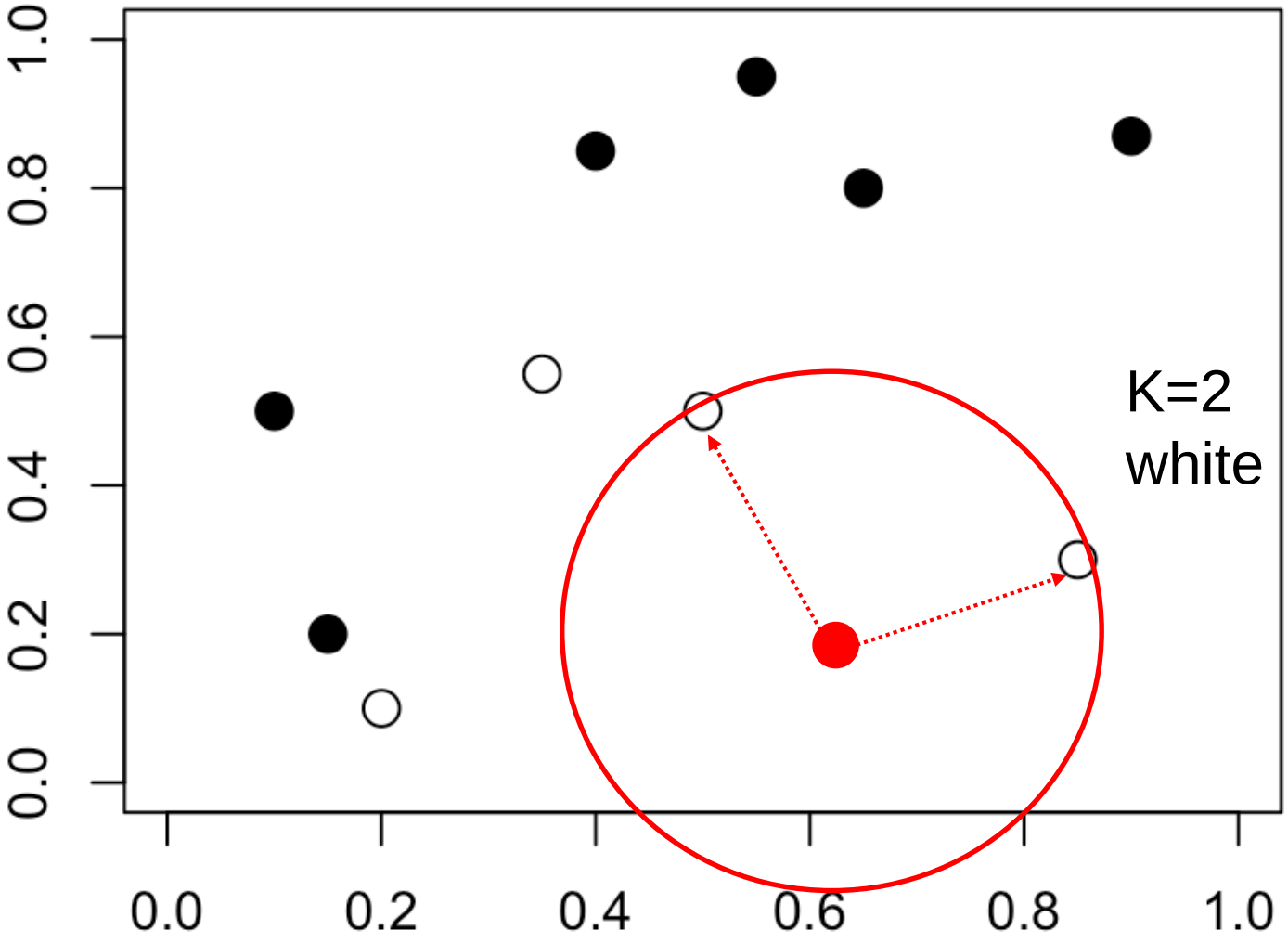
OXFORD BROOKES UNIVERSITY

K-NN



OXFORD BROOKES UNIVERSITY

K-NN



OXFORD BROOKES UNIVERSITY

K-NN

To make classifier:

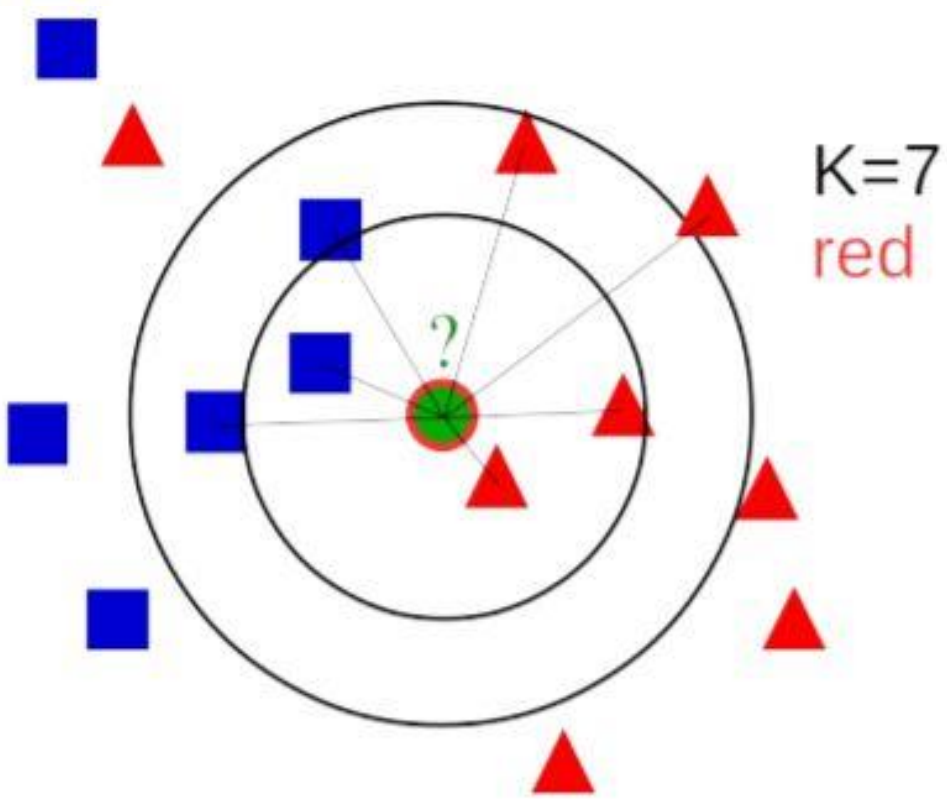
- Feature Space (Training Data)
- Distance metric to compute distance between records
- The value of k - the number of nearest neighbors to retrieve from which to get majority class

To classify an unknown record:

- Compute distance to other training records
- Identify k nearest neighbors
- Use class labels of nearest neighbors to determine the class label of unknown record

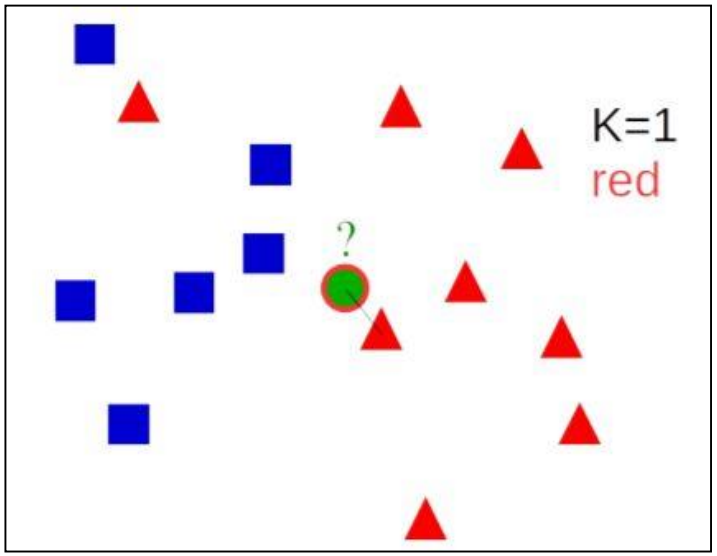
OXFORD BROOKES UNIVERSITY

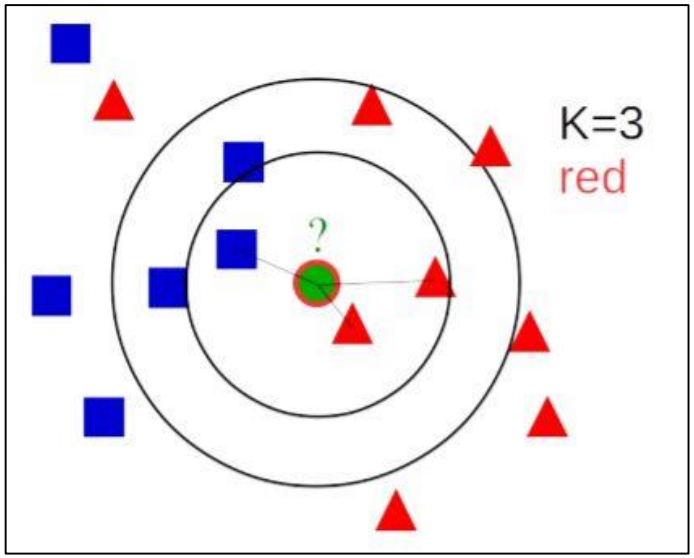
K-NN

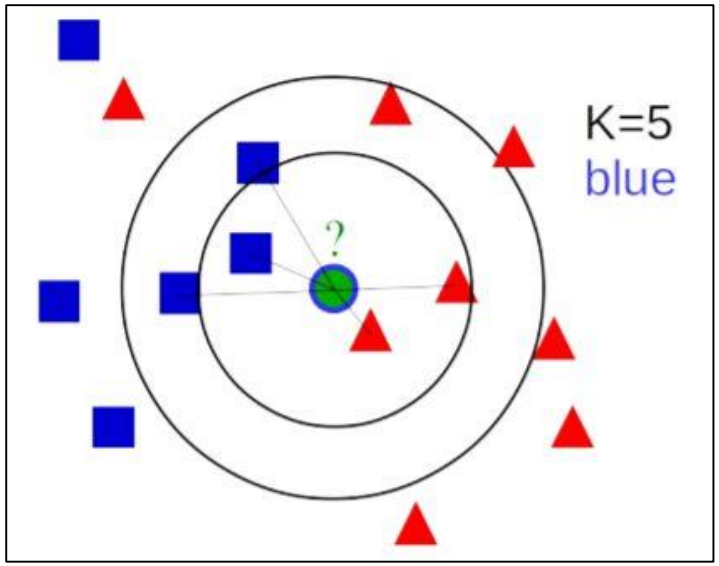


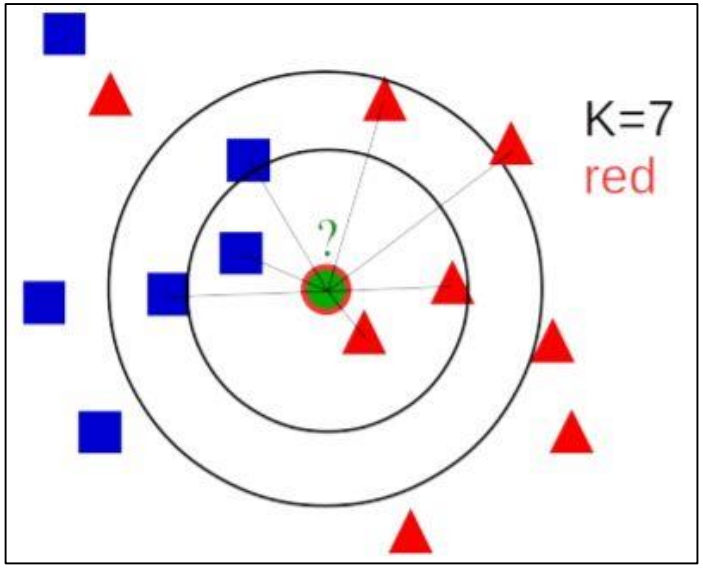
OXFORD BROOKES UNIVERSITY

K=?
DISTANCE?



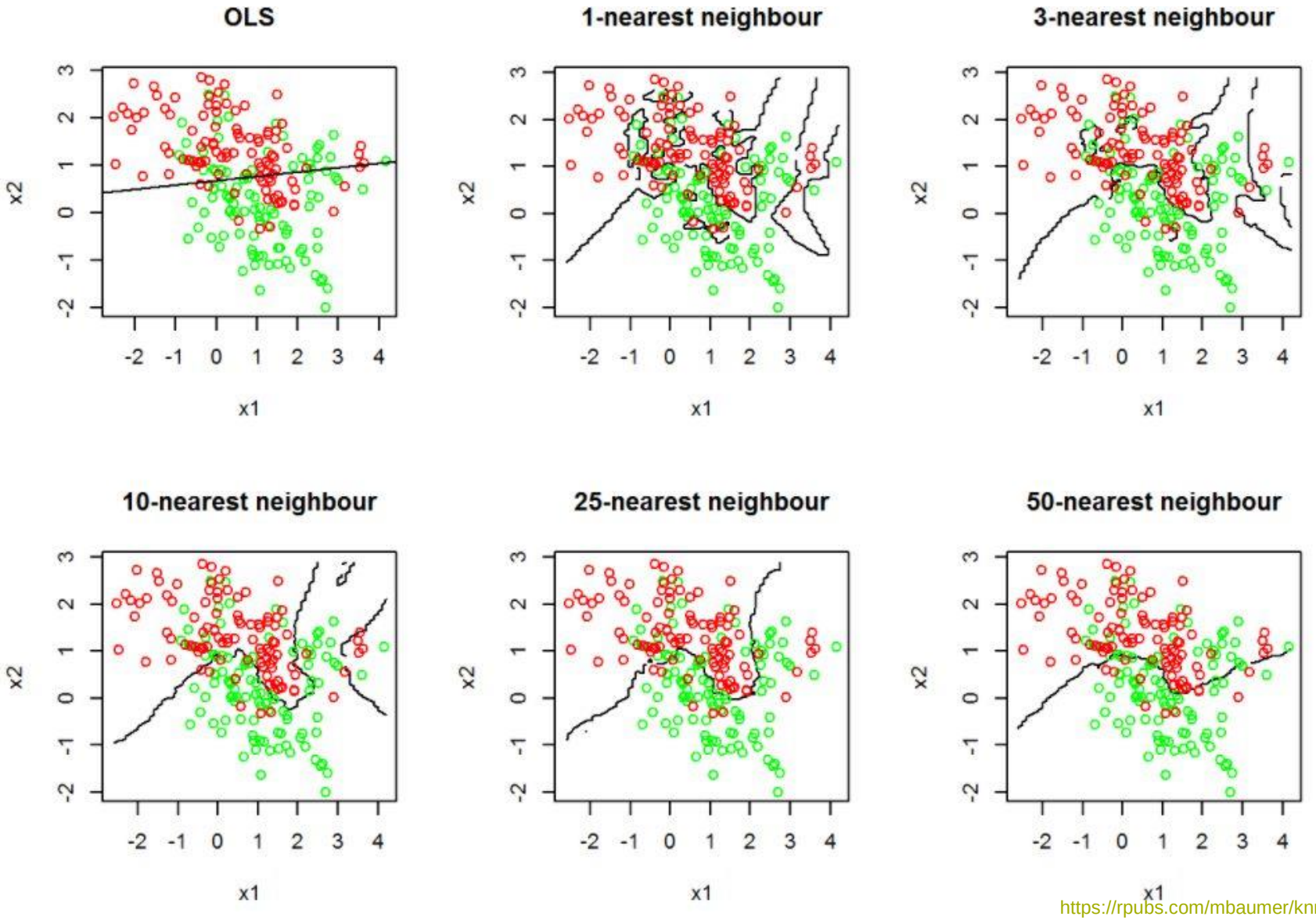






OXFORD BROOKES UNIVERSITY

K=?

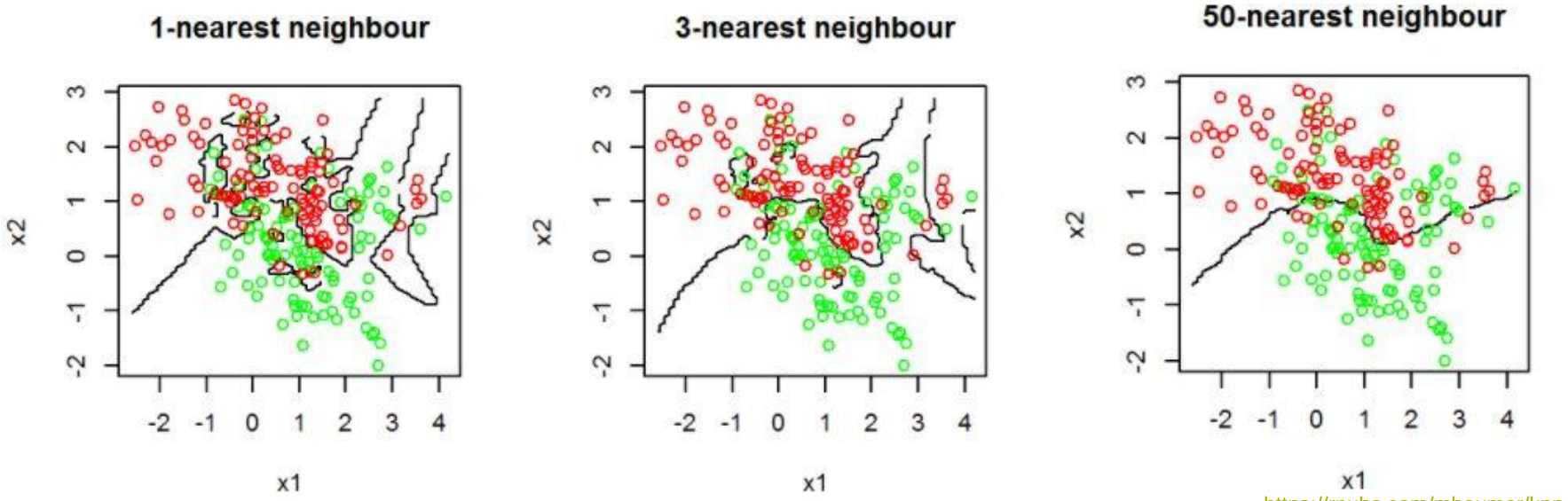


https://rpubs.com/mbaumer/knn

OXFORD BROOKES UNIVERSITY

HOW TO CHOOSE K?

- If K is too small it is sensitive to noise points
- Larger K works well but too large K may include majority points from other classes.



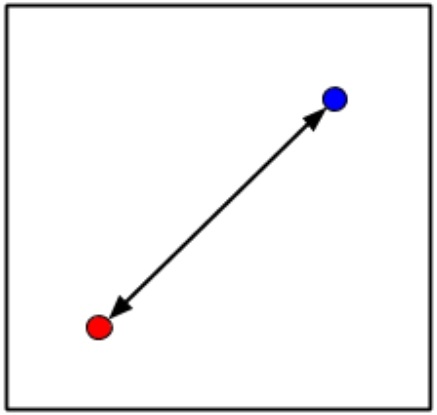
https://rpubs.com/mbaumer/knn

- Rule of thumb is K< sqrt(n), n is the number of examples
- Choose an **odd value for k**, to eliminate ties

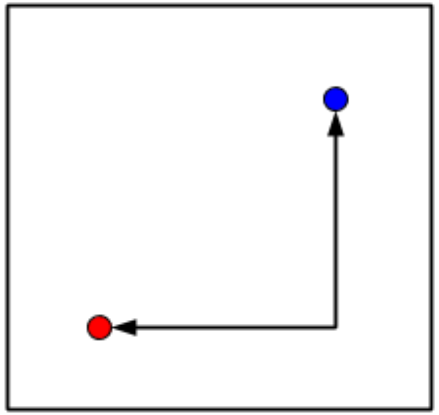
OXFORD BROOKES UNIVERSITY

THE MOST COMMON DISTANCE MEASURES

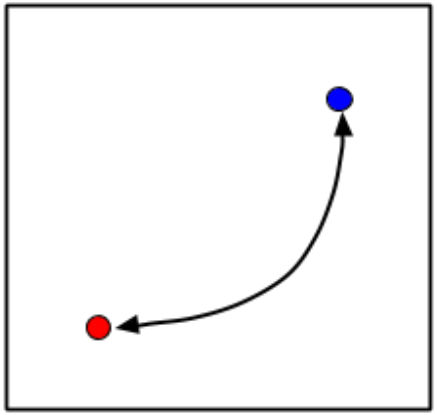
Euclidean



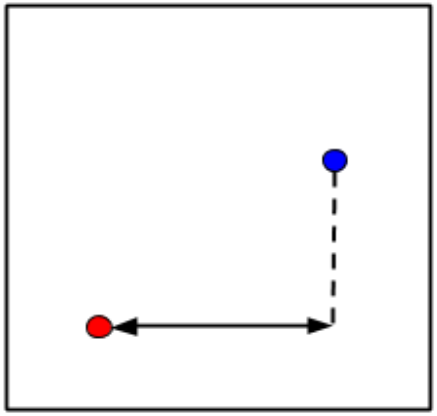
Manhattan



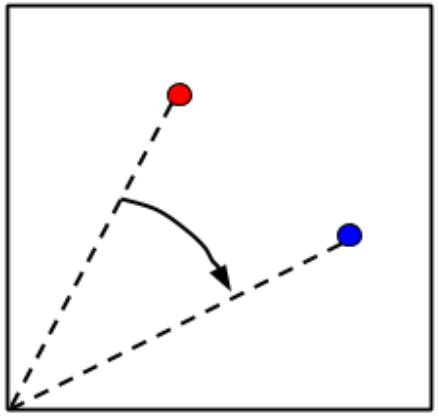
Minkowski



Chebychev



Cosine Similarity



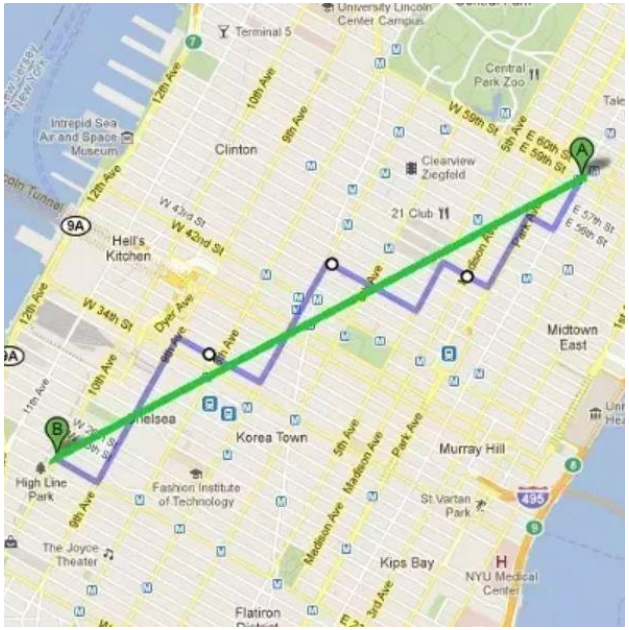
Hamming

K-NN: DISTANCE METRICS

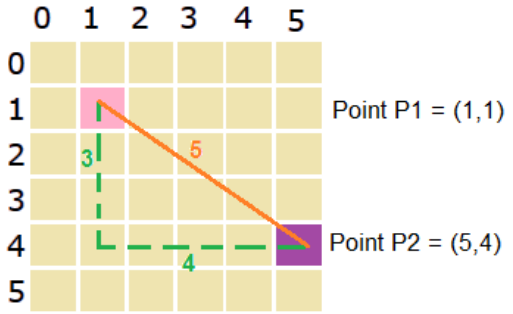
- Euclidian distance (straight line)

$$d(p,q) = \sqrt{\sum (p_i - q_i)^2}$$

- Manhattan distance (feature-wise distance)



$$d(p,q) = \sum |p_i - q_i|$$



Euclidean distance = $\sqrt{(5-1)^2 + (4-1)^2} = 5$

Manhattan distance = $|5-1| + |4-1| = 7$

NOMINAL/CATEGORICAL DATA

- Distance works naturally with numerical attributes
- Binary value categorical data attributes can be taken as o or 1.

$$D_H = \sum_{i=1}^{k} |x_i - y_i|$$

- $x = y \Rightarrow D = 0$
- $x \neq y \Rightarrow D = 1$

x	y	Distance
Male	Male	0
Male	Female	1

- Factors are used to represent categorical data. Factors can be ordered or unordered and are an important class for statistical analysis and for plotting.
- Factors are stored as integers, and have labels associated with these unique integers. While factors look (and often behave) like character vectors, they are actually integers under the hood, and you need to be careful when treating them like strings. Once created, factors can only contain a pre-defined set values, known as *levels*. By default, R always sorts *levels* in alphabetical order.

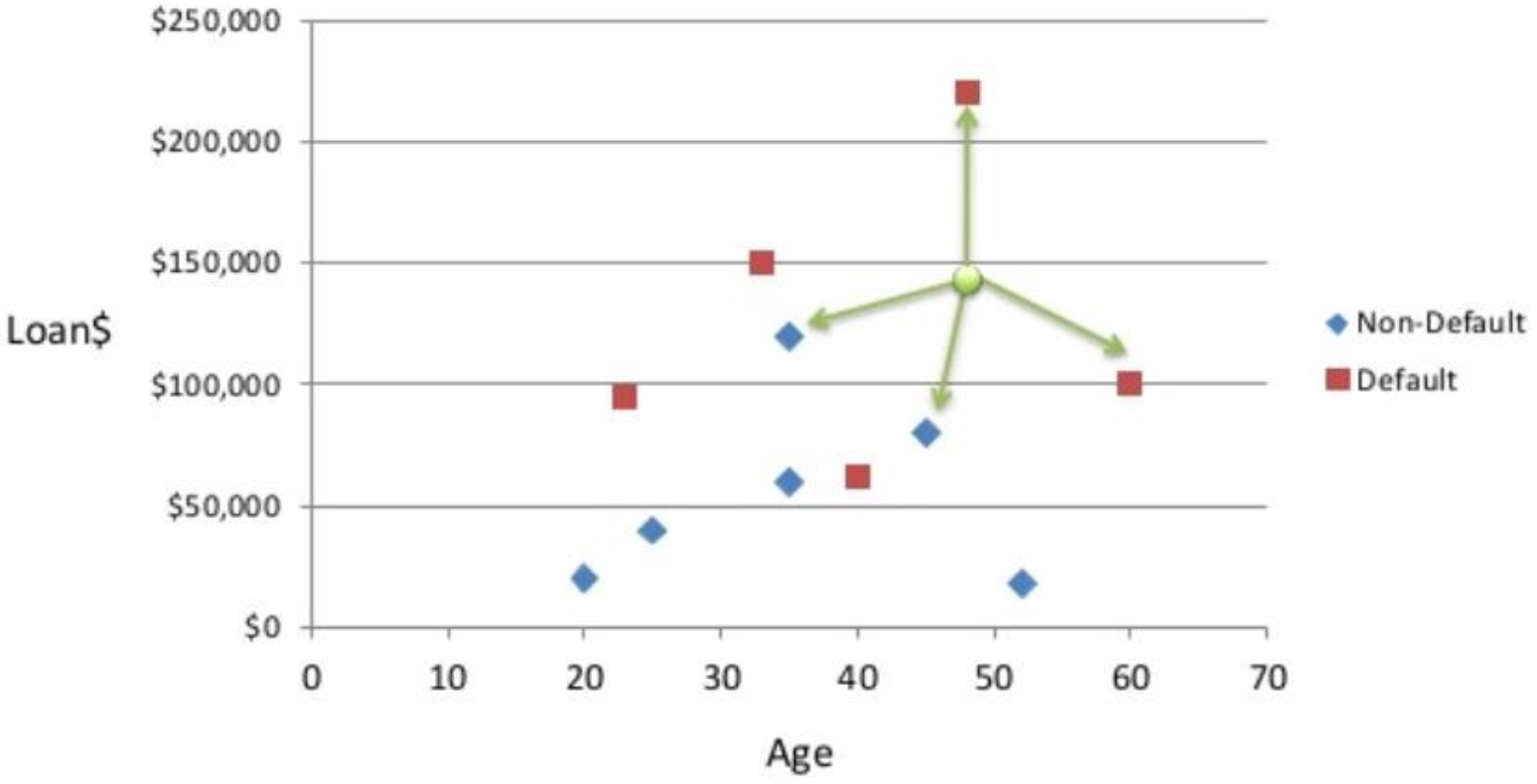
CASE 1: SIMPLE BANK-LOAN MODEL USING K-NEAREST NEIGHBOURS

OXFORD BROOKES UNIVERSITY

- Application of historical customers' information, accumulated by banks overtime, to predict whether a customer applying for a loan item will default, or otherwise, is the magic to maintain a "clean" book.
- The ML tools can also be used to maintain favorable PAR (Portfolio at risk) levels
- **Task:** to calculate the likelihood if a customer will be a defaulter or not.
- **Data:** We will use a historical bank data to construct a predictive loan model.

OXFORD BROOKES UNIVERSITY

IF A CUSTOMER WILL BE A DEFAULTER?



https://www.slideshare.net/tilanigunawardena/k-nearest-neighbors

OXFORD BROOKES UNIVERSITY

CALCULATE DISTANCE

Age	Loan	Default	Distance	
25	$40,000	N	102000	
35	$60,000	N	82000	
45	$80,000	N	62000	
20	$20,000	N	122000	
35	$120,000	N	22000	2
52	$18,000	N	124000	
23	$95,000	Y	47000	
40	$62,000	Y	80000	
60	$100,000	Y	42000	3
48	$220,000	Y	78000	
33	$150,000	Y	8000	1
48	**$142,000**	**?**		

Euclidean Distance

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

D = Sqrt[(48-33)^2 + (142000-150000)^2] = 8000.01 >> Default=Y

https://www.slideshare.net/tilanigunawardena/k-nearest-neighbors

OXFORD BROOKES UNIVERSITY

FEATURE NORMALIZATION

- Distance between neigbours could be dominated by some attributes with relatively large numbers (e.g., income of customers)

Age	Loan	Default	Distance
25	$40,000	N	102000
35	$60,000	N	82000

- It arises when two features are in different scales
- Important to normalize those features

- Mapping values to numbers between 0..1

Age	
25	0.125
35	0.375
45	0.625
20	0
35	0.375
52	0.8
23	0.075
40	0.5
60	1
48	0.7
33	0.325

$$a = \frac{v_i - min_v}{max_v - min_v}$$

Loan	
$40,000	0.11
$60,000	0.21
$80,000	0.31
$20,000	0.01
$120,000	0.50
$18,000	0.00
$95,000	0.38
$62,000	0.22
$100,000	0.41
$220,000	1.00
$150,000	0.65

CALCULATE DISTANCE NORMALIZED VARIABLES

OXFORD BROOKES UNIVERSITY

Age	Loan	Default	Distance	
0.125	0.11	N	0.7652	
0.375	0.21	N	0.5200	
0.625	0.31	N	0.3160	1
0	0.01	N	0.9245	
0.375	0.50	N	0.3428	2
0.8	0.00	N	0.6220	
0.075	0.38	Y	0.6669	
0.5	0.22	Y	0.4437	
1	0.41	Y	0.3650	3
0.7	1.00	Y	0.3861	
0.325	0.65	Y	0.3771	
0.7	0.61	?		

Standardized Variable

$$X_s = \frac{X - Min}{Max - Min}$$

Using the standardized distance on the same training set, the unknown case returned a different neighbor which is not a good sign of robustness.

https://www.slideshare.net/tilanigunawardena/k-nearest-neighbors

CASE 2: TIME-SERIES ANALYSIS: WEARABLE DEVICES USING DTW AND KNN

OXFORD BROOKES UNIVERSITY

Fitbits, Apple watches, Google Glass etc all use time-series measurements to track a user's movements.

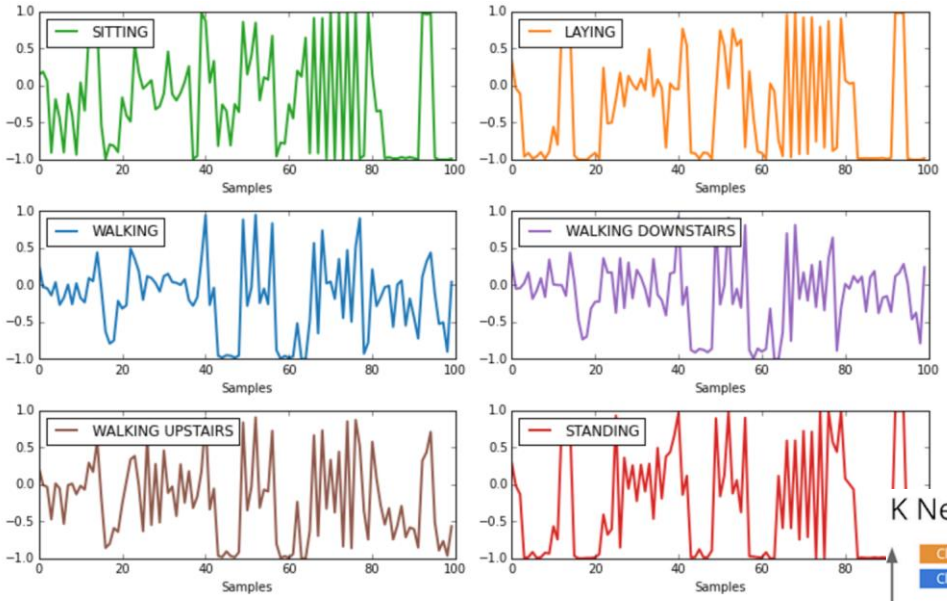
The general idea is to classify known movements into categories: sitting, standing, walking, lying down etc.

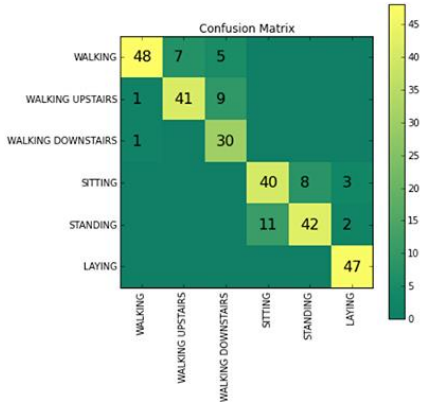
https://sflscientific.com/data-science-blog/2016/6/4/time-series-analysis-fitbit-using-dtw-and-knn

CASE 2: TIME-SERIES ANALYSIS: WEARABLE DEVICES USING DTW AND KNN

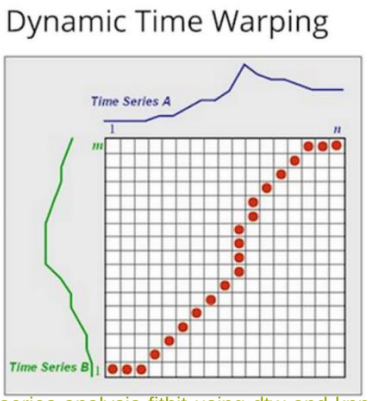
OXFORD BROOKES UNIVERSITY

- Using template waves for each of these activities we can compare using kNN and DTW a new sequence





	precision	recall	f1-score	support
WALKING	0.96	0.80	0.87	60
WALKING UPSTAIRS	0.85	0.80	0.83	51
WALKING DOWNSTAIRS	0.68	0.97	0.80	31
SITTING	0.78	0.78	0.78	51
STANDING	0.84	0.76	0.80	55
LAYING	0.90	1.00	0.95	47
avg / total	0.85	0.84	0.84	295



https://sflscientific.com/data-science-blog/2016/6/4/time-series-analysis-fitbit-using-dtw-and-knn

OXFORD BROOKES UNIVERSITY

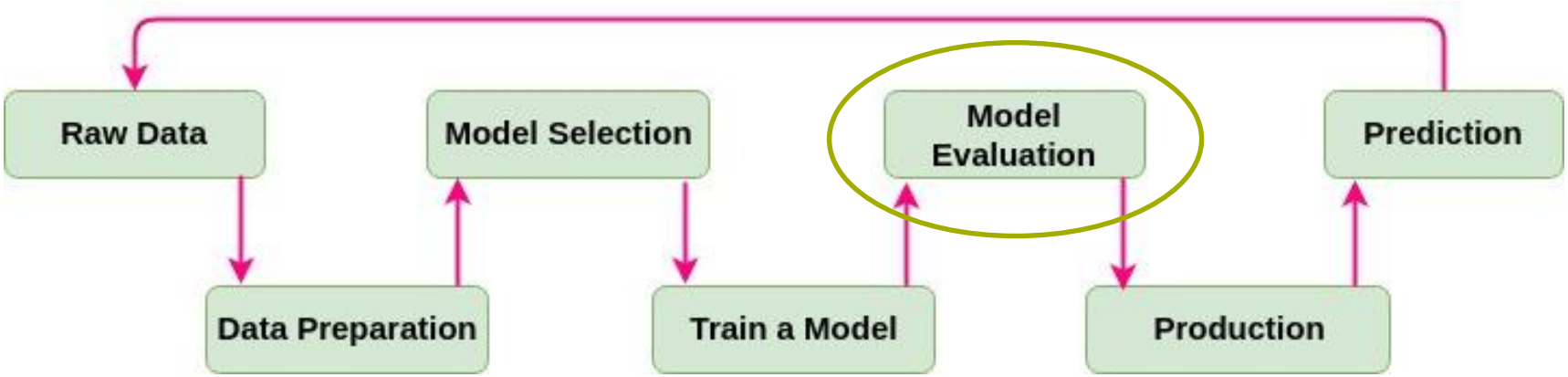
K-NEAREST NEIGHBOURS

- Pros:
  - Simple technique that is easily implemented
  - Can model very complicated, multi-modal datasets
- Cons:
  - Computationally expensive to run for new data (go through all past data)
  - Does not do well with noise k-nearest neighbors

OXFORD BROOKES UNIVERSITY

MODEL EVALUATION: METRICS

1. Root Mean Squared Error (RMSE)
2. Confusion Matrix
3. F1 Score
4. Area Under the ROC curve (AUC – ROC)

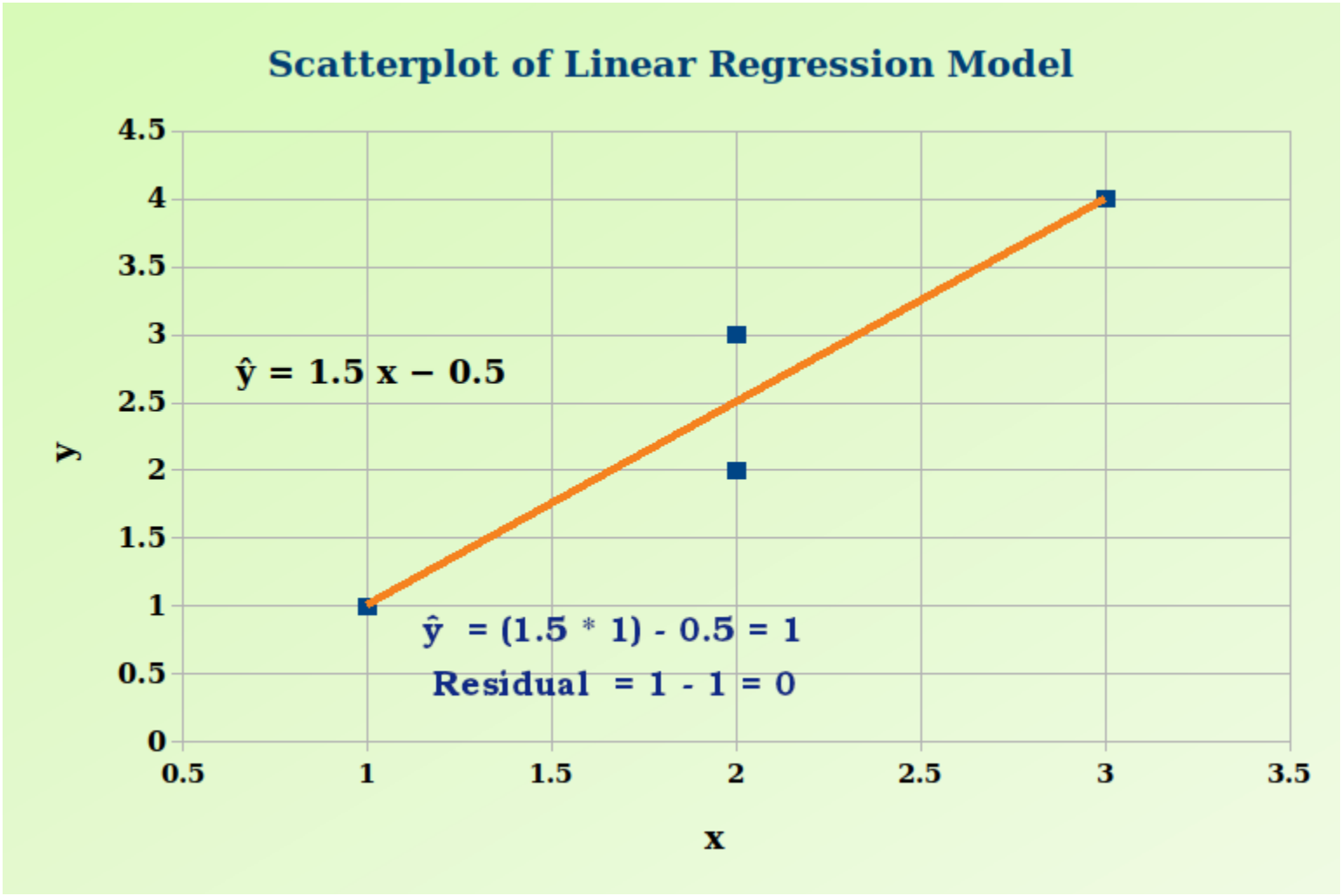


https://manisha-sirsat.blogspot.com/2019/04/machinelearning-process.html

1. RMSE

Residual

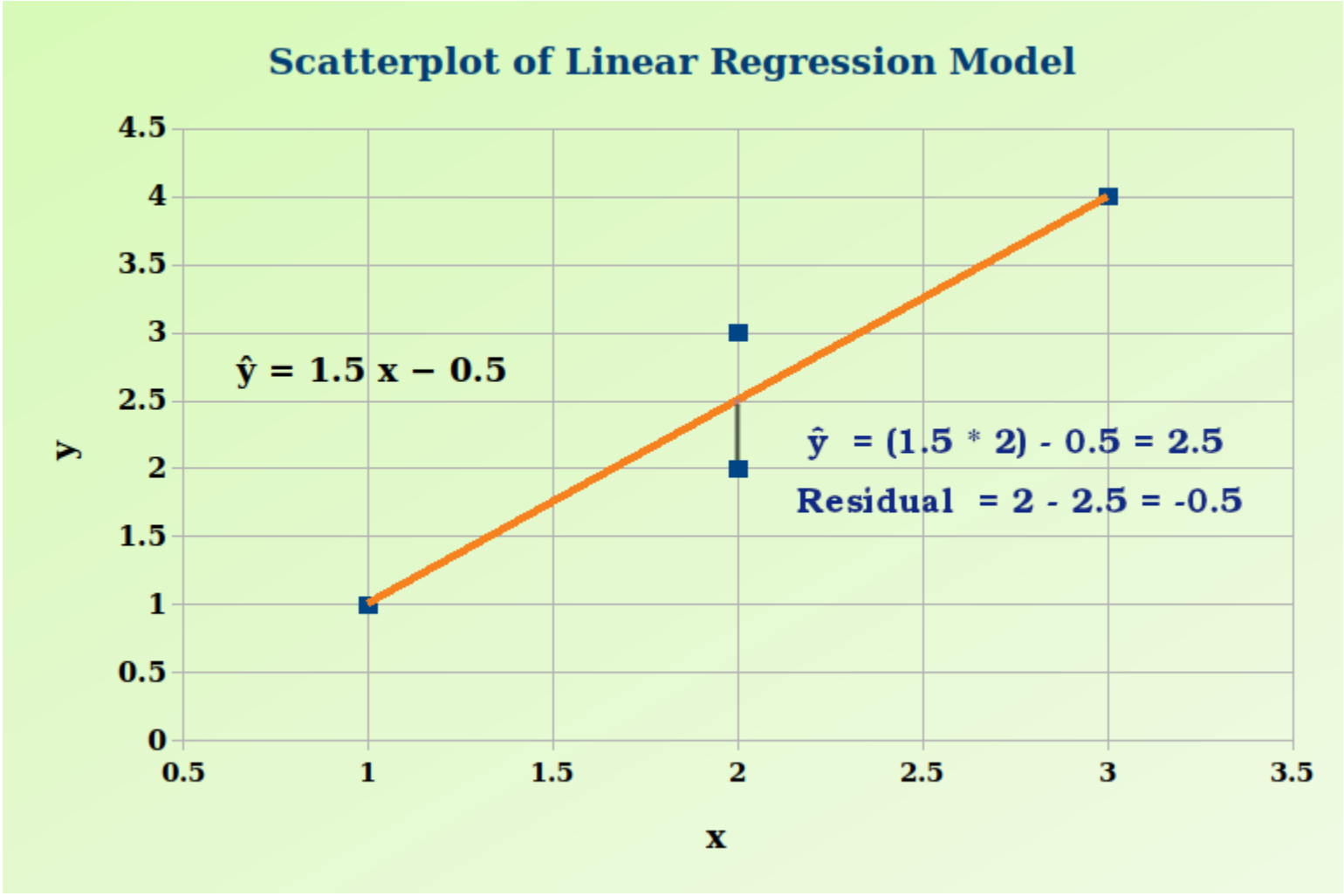
$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$$



OXFORD BROOKES UNIVERSITY

1. RMSE

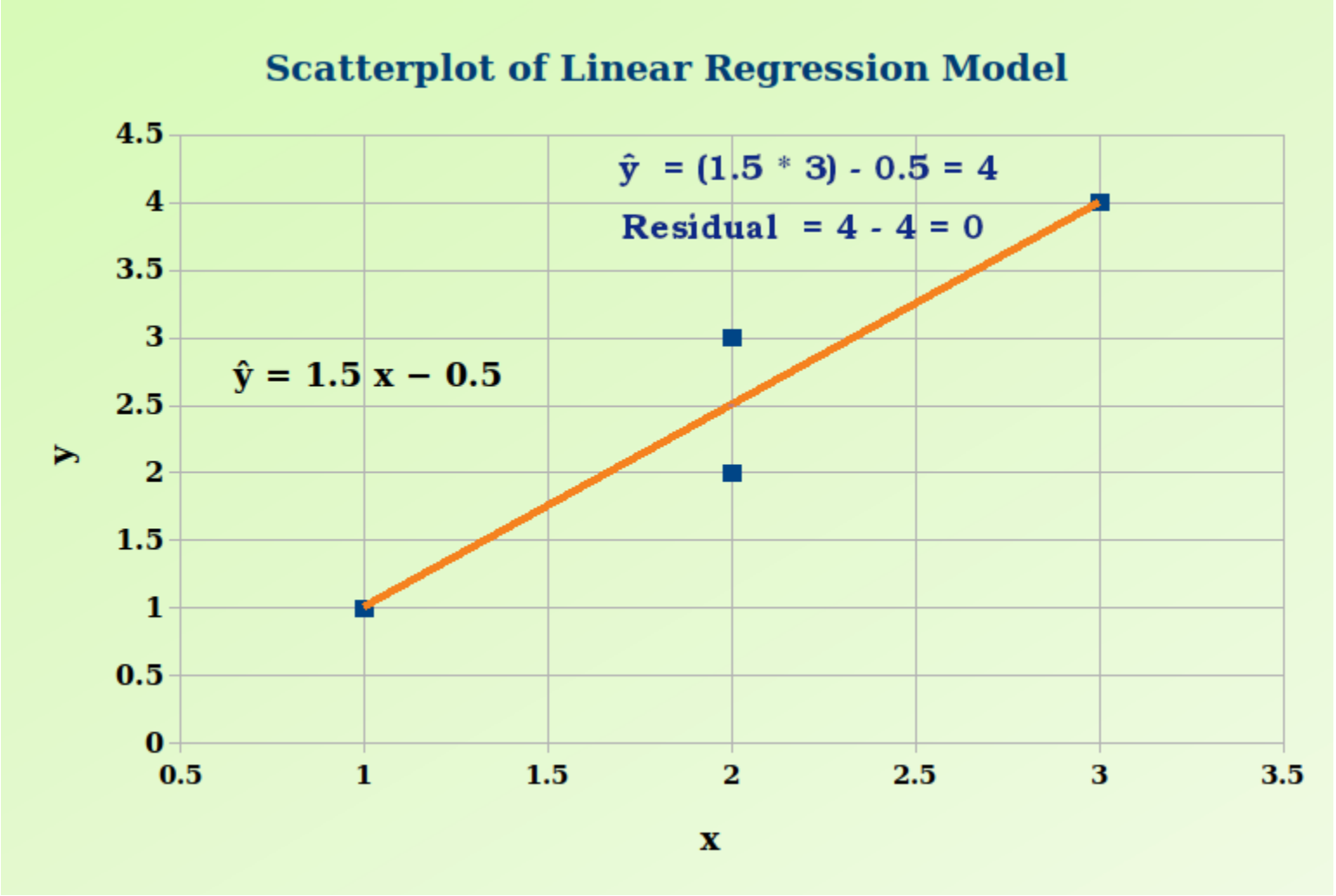
$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$$



OXFORD BROOKES UNIVERSITY

1. RMSE

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$$



OXFORD BROOKES UNIVERSITY

1. RMSE

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}$$

take the error of each one of the data points, square it, then add them together, and take the average.

$$RMSE = \sqrt{\frac{(0)^2+(-0.5)^2+(0.5)^2+(0)^2}{4}} = \sqrt{0.125} = 0.3535$$

This way we can know how much our machine learning model disagrees with the actual data.

OXFORD BROOKES UNIVERSITY

2. CONFUSION MATRIX

- A confusion matrix is an N X N matrix, where N is the number of classes being predicted.

Confusion Matrix		Target			
		Positive	Negative		
Model	Positive	a	b	*Positive Predictive Value*	a/(a+b)
	Negative	c	d	*Negative Predictive Value*	d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)	
		a/(a+c)	d/(b+d)		

- **Accuracy** : the proportion of the total number of predictions that were correct.
- **Positive Predictive Value or Precision** : the proportion of positive cases that were correctly identified.
- **Negative Predictive Value** : the proportion of negative cases that were correctly identified.
- **Sensitivity or Recall** : the proportion of actual positive cases which are correctly identified.
- **Specificity** : the proportion of actual negative cases which are correctly identified.

2. CONFUSION MATRIX

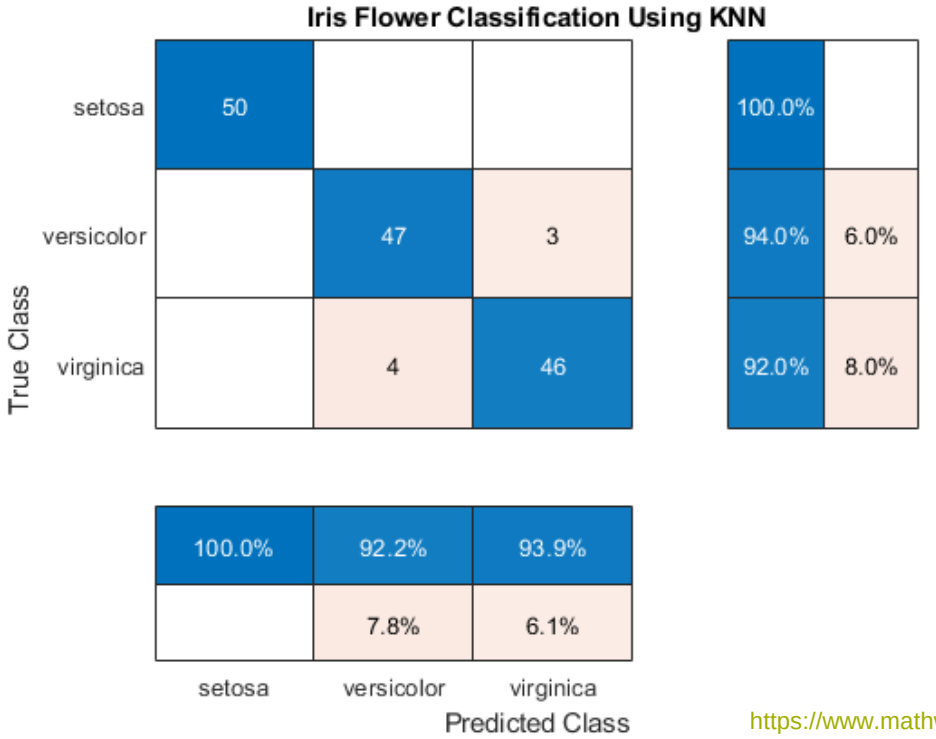
Confusion Matrix		Target			
		Positive	Negative		
Model	Positive	a	b	*Positive Predictive Value*	a/(a+b)
	Negative	c	d	*Negative Predictive Value*	d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)	
		a/(a+c)	d/(b+d)		

Count of ID	Target			
Model	**1**	**0**	**Grand Total**	
1	3,834	639	4,473	85.7%
0	16	951	967	1.7%
Grand Total	**3,850**	**1,590**	**5,440**	
	99.6%	40.19%		88.0%

https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/

2. CONFUSION MATRIX

Confusion Matrix		Target: Positive	Target: Negative		
Model	Positive	a	b	*Positive Predictive Value*	a/(a+b)
	Negative	c	d	*Negative Predictive Value*	d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)	
		a/(a+c)	d/(b+d)		



https://www.mathworks.com/help/stats/confusionchart.html

OXFORD BROOKES UNIVERSITY

3. F1-SCORE

F1-Score is the harmonic mean of precision and recall values for a classification problem

Confusion Matrix		Target		
		Positive	Negative	**Precision**
Model	Positive	a	b	*Positive Predictive Value* a/(a+b)
	Negative	c	d	*Negative Predictive Value* d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)
		a/(a+c)	d/(b+d)	

Recall

$$F_1 = \left(\frac{\mathrm{recall}^{-1} + \mathrm{precision}^{-1}}{2} \right)^{-1} = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}.$$

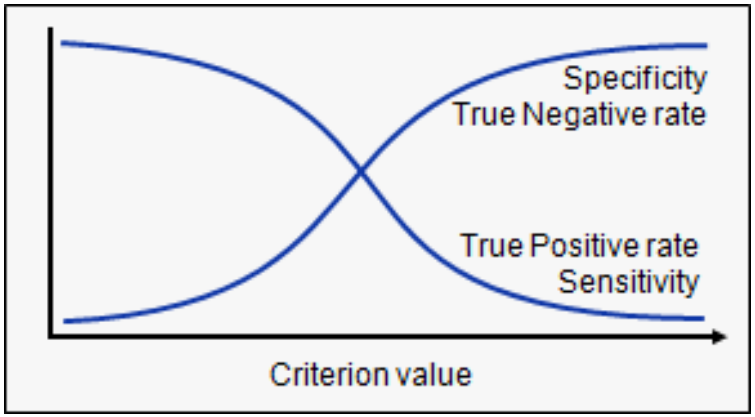
4. ROC: RECEIVER OPERATING CHARACTERISTIC

OXFORD BROOKES UNIVERSITY

- If we look at the confusion matrix, we observe that for a probabilistic model, we get different value for each metric.

Confusion Matrix		Target			
		Positive	Negative		
Model	Positive	a	b	*Positive Predictive Value*	a/(a+b)
	Negative	c	d	*Negative Predictive Value*	d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)	
		a/(a+c)	d/(b+d)		

For each sensitivity, we get a different specificity

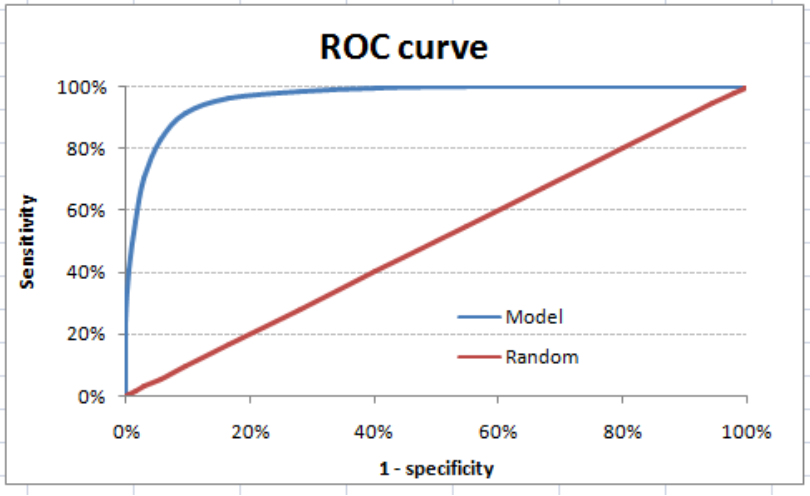


4. AREA UNDER THE ROC CURVE (AUC – ROC)

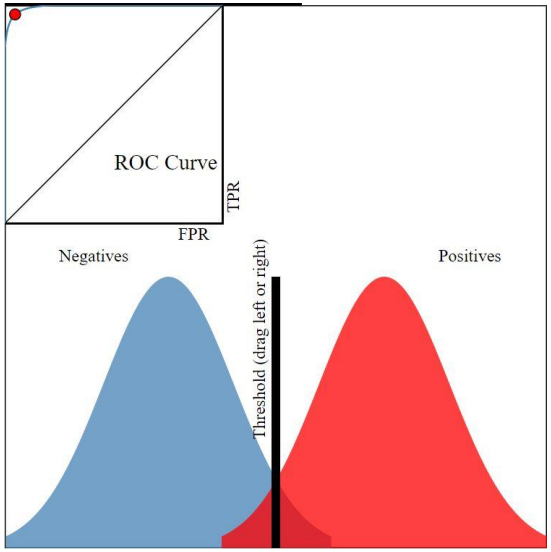
OXFORD BROOKES UNIVERSITY

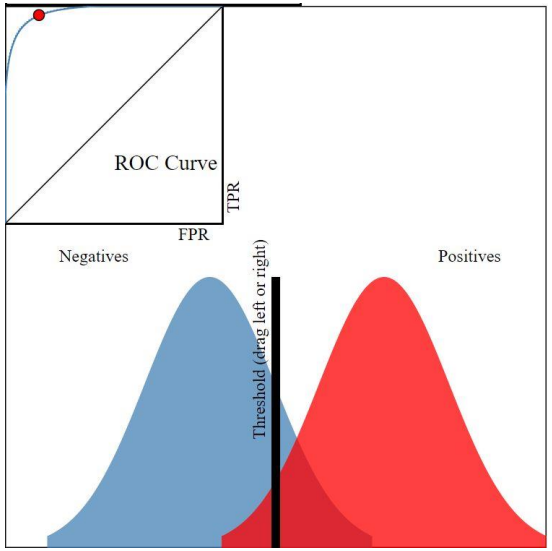
The ROC curve is the plot between sensitivity and (1- specificity)

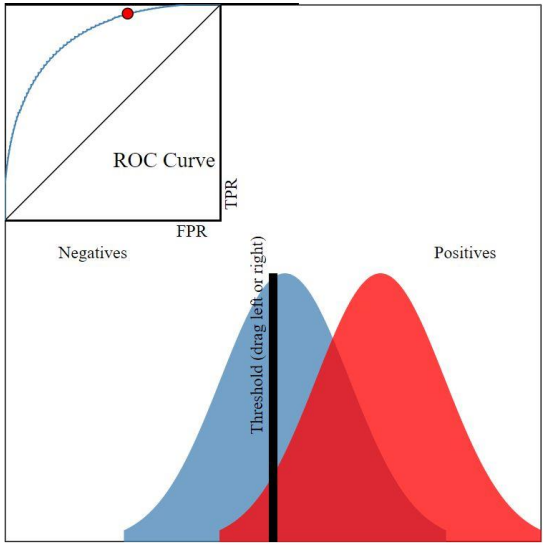
Confusion Matrix		Target			
		Positive	Negative		
Model	Positive	a	b	*Positive Predictive Value*	a/(a+b)
	Negative	c	d	*Negative Predictive Value*	d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)	
		a/(a+c)	d/(b+d)		
		TPR	**FPR**		

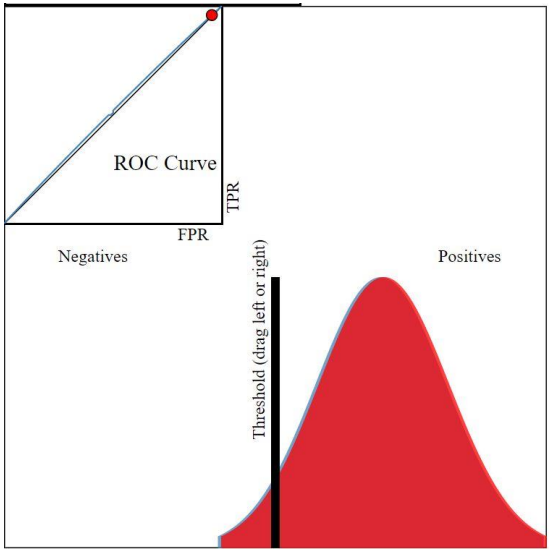


AUC – ROC









ROC is a probability curve and AUC represents degree or measure of separability.

It tells how much model is capable of distinguishing between classes.

http://www.navan.name/roc/

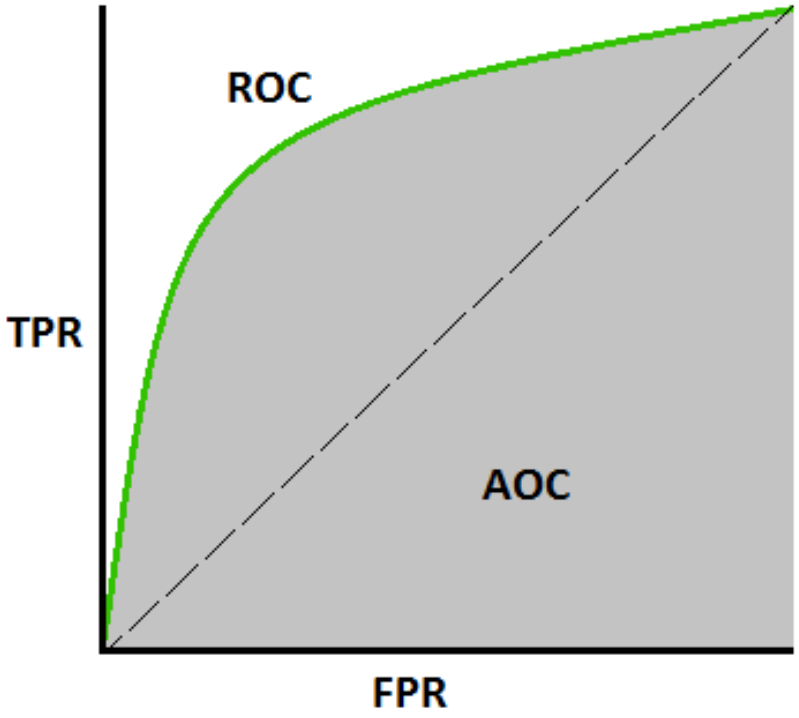
4. AUC – ROC

Area of entire square is 1*1 = 1.

AUC is the ratio under the curve and the total area.

A few thumb rules:

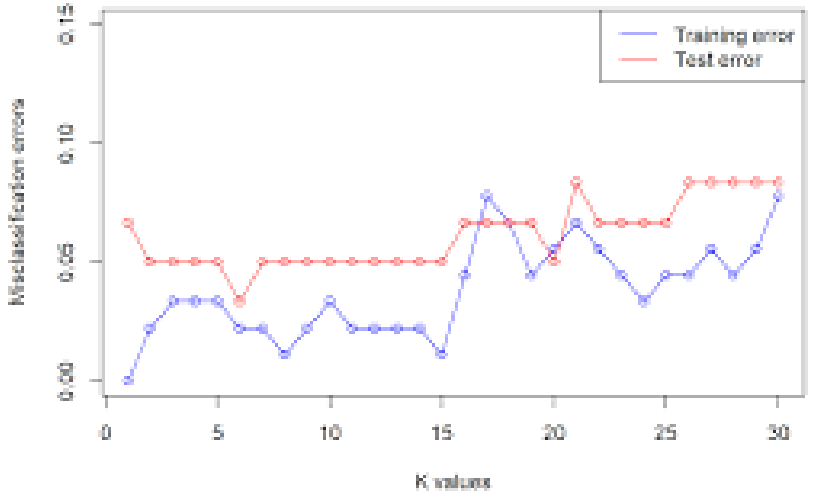
- .90-1 = excellent (A)
- .80-.90 = good (B)
- .70-.80 = fair (C)
- .60-.70 = poor (D)
- .50-.60 = fail (F)



https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/

TEST ERROR

- Models do not always behave well on new data
- Performance on new data is the whole point of machine learning
- Hence meaningful evaluation must measure the error on new data
- Test data → Test error



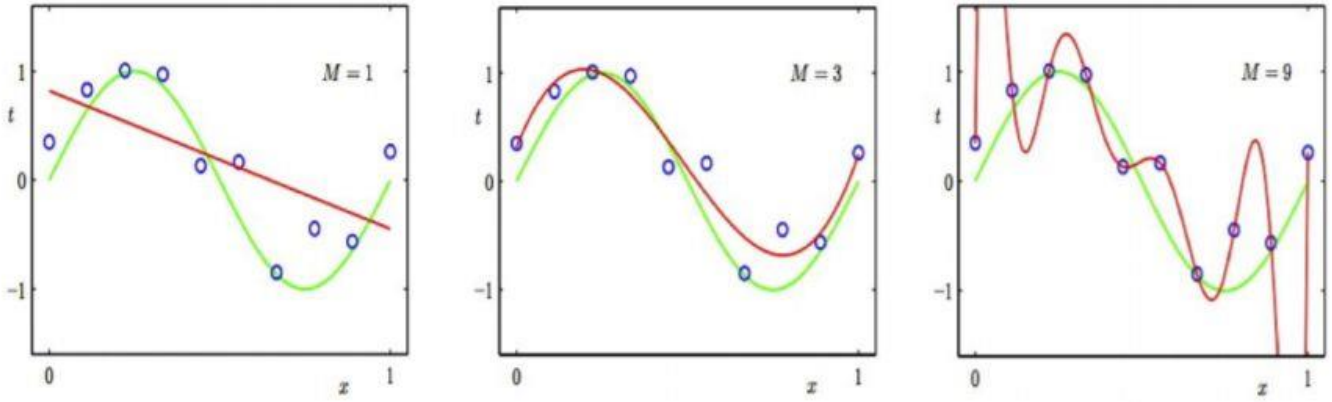
OXFORD BROOKES UNIVERSITY

OVER AND UNDERFITTING

Underfitting

Overfitting

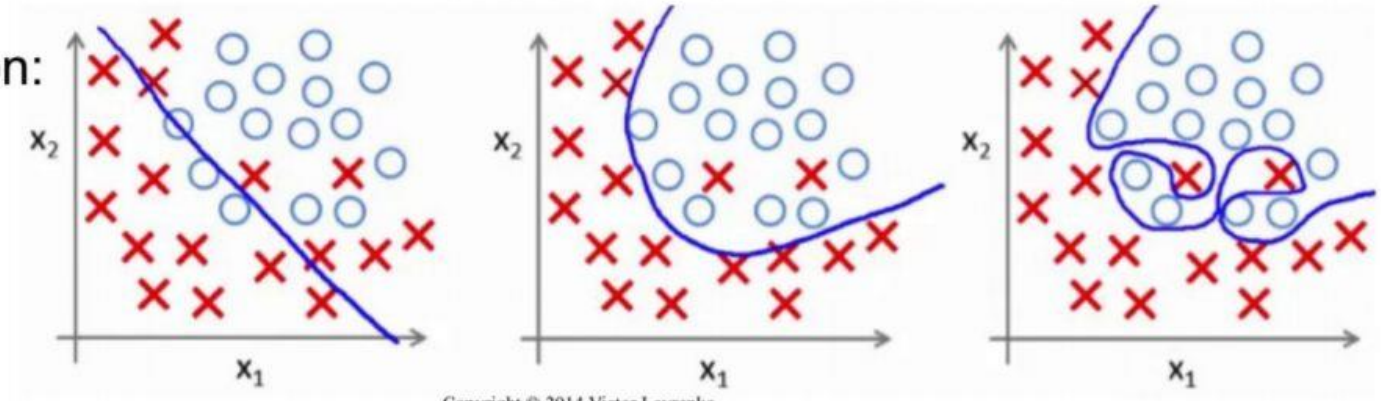
Regression:



predictor too inflexible:
cannot capture pattern

predictor too flexible:
fits noise in the data

Classification:



OXFORD BROOKES UNIVERSITY

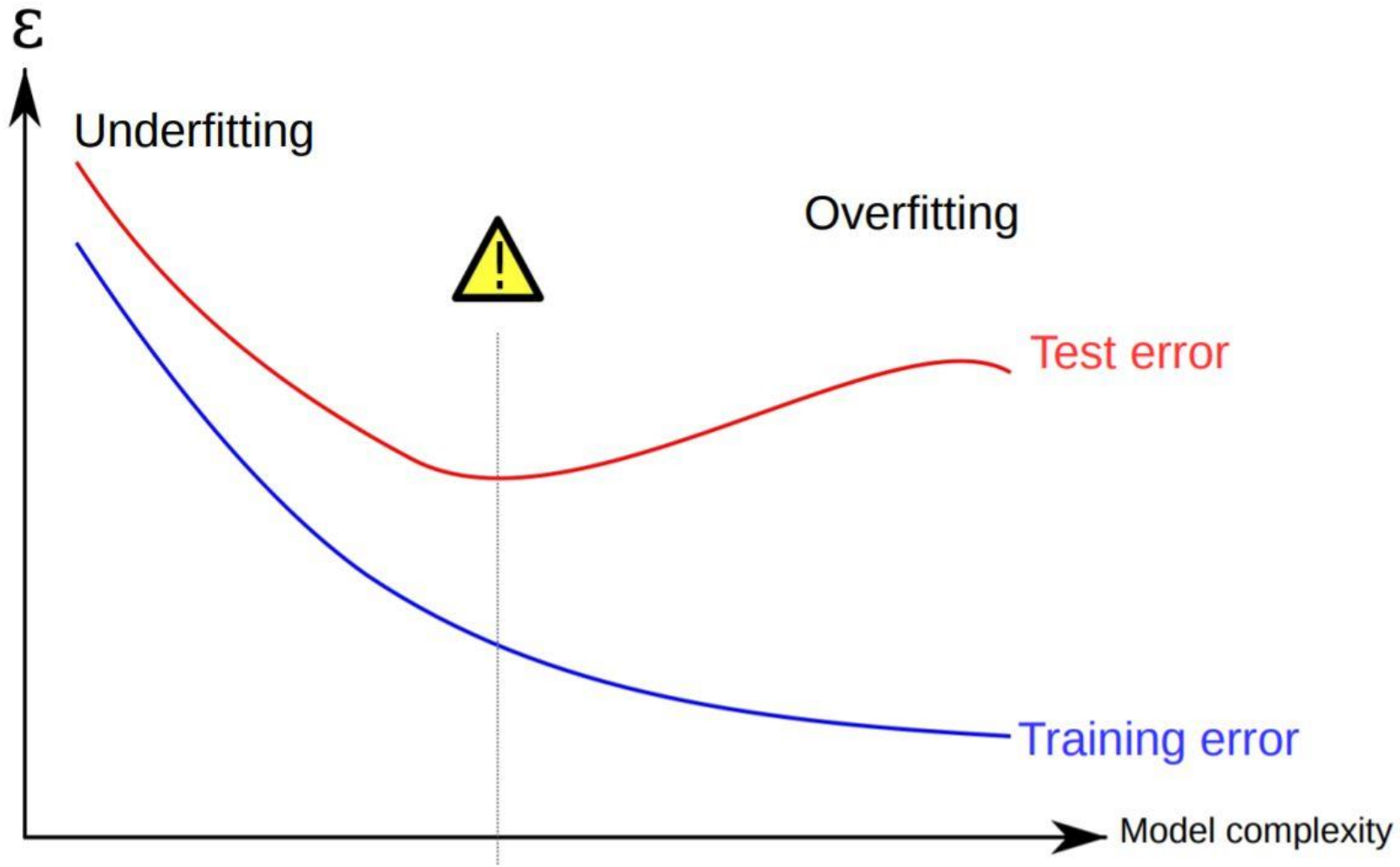
OVER AND UNDERFITTING

- *The **bias** is an error from erroneous assumptions in the learning algorithm.*
- the difference between the true label and our prediction

- *The **variance** is an error from sensitivity to small fluctuations in the training set.*
- the expectation of the squared deviation of a random variable from its mean

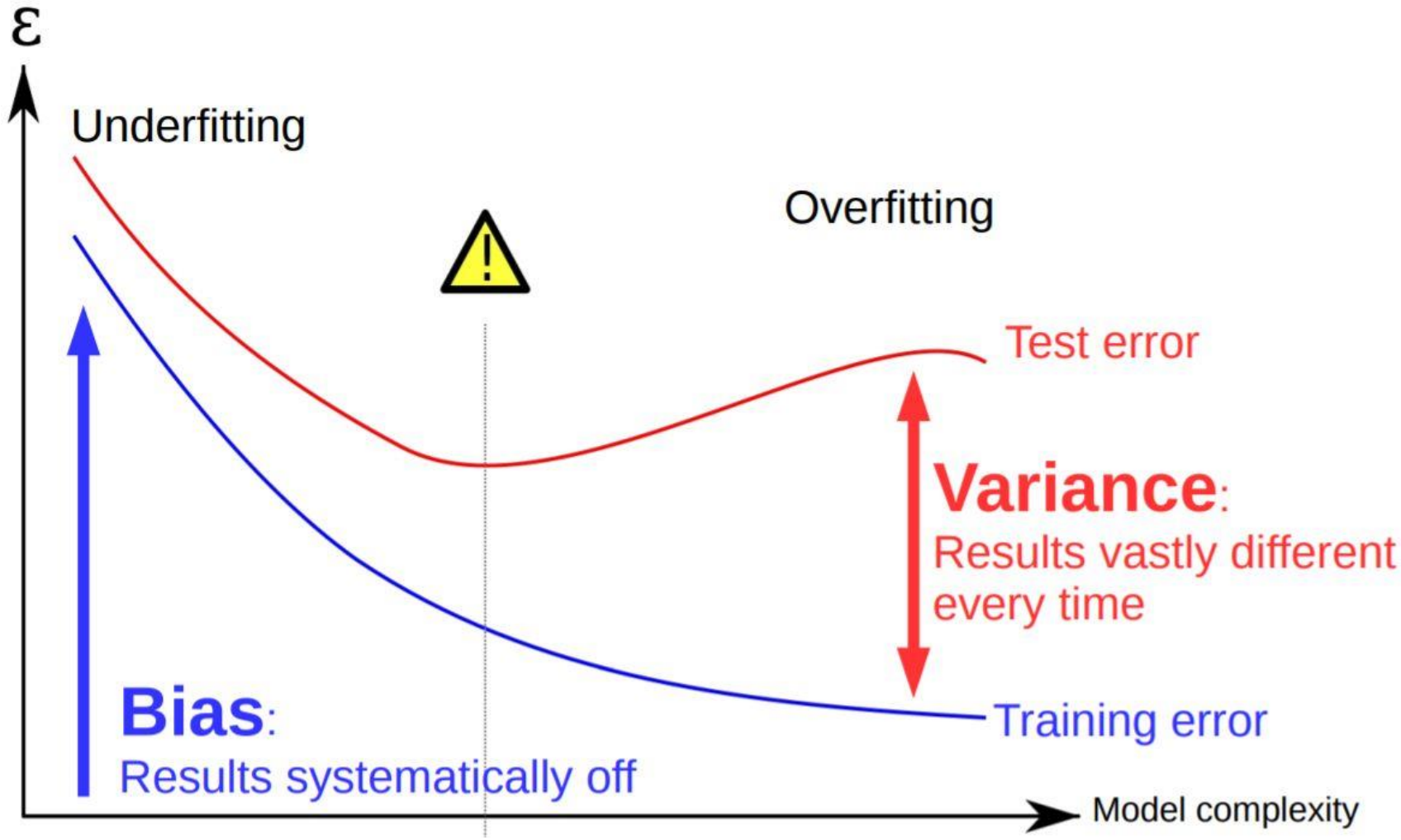
OXFORD BROOKES UNIVERSITY

OVER AND UNDERFITTING



OXFORD BROOKES UNIVERSITY

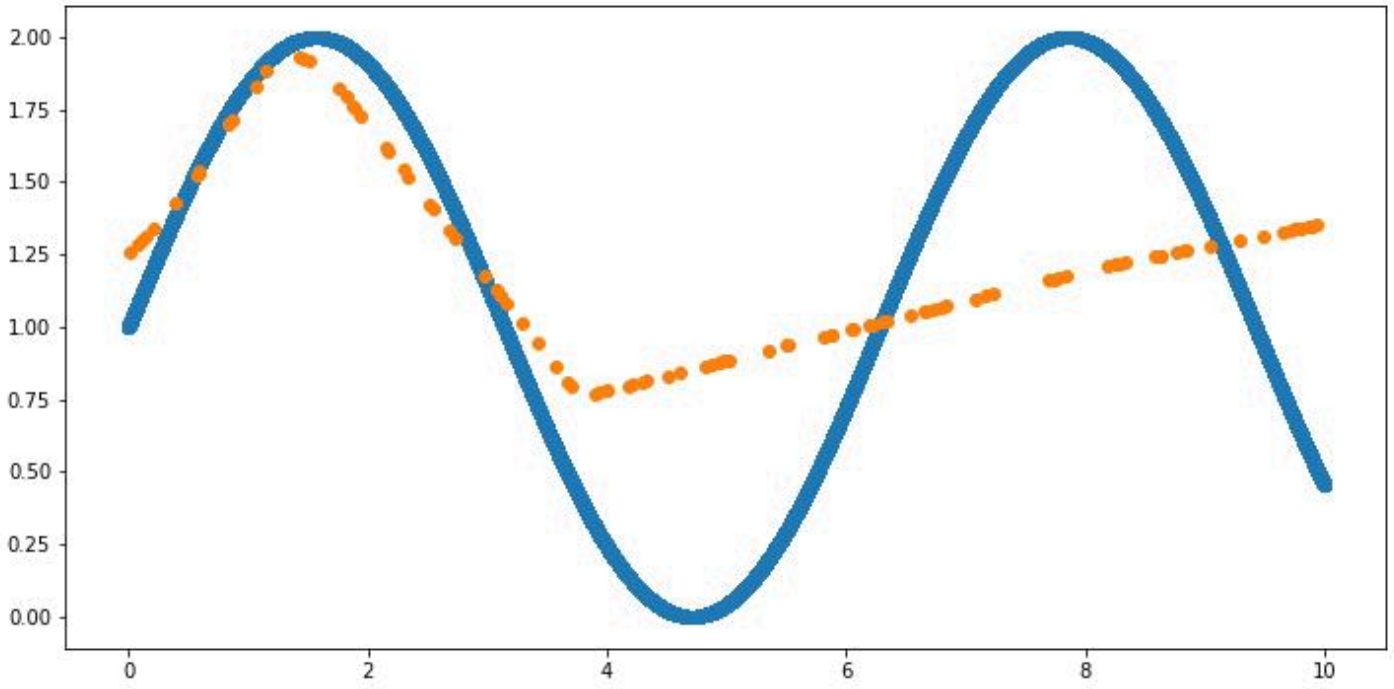
OVER AND UNDERFITTING



OXFORD BROOKES UNIVERSITY

OVER AND UNDERFITTING

Even simple models can over-fit with high variance if too little data

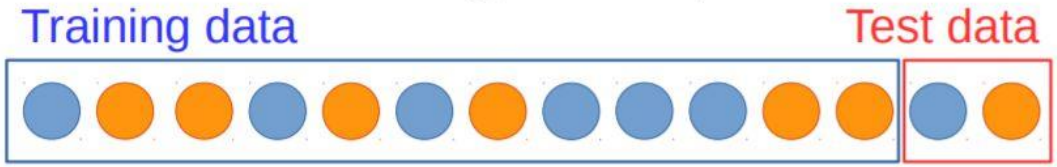


OXFORD BROOKES UNIVERSITY

TEST ERROR

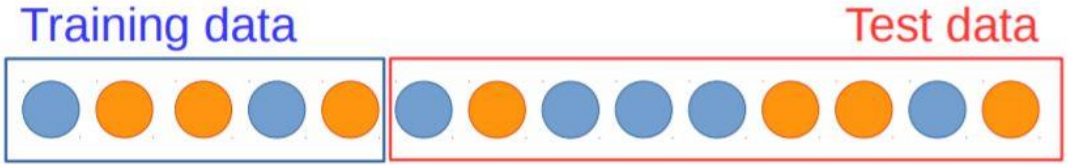
- Maximize amount of training data

  → Better training, but poor evaluation

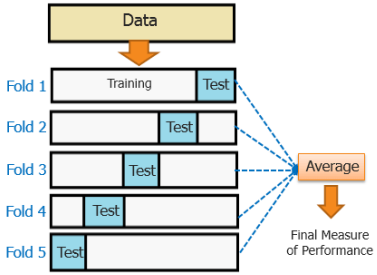


- But also enough test data?

  → Better evaluation, poor training

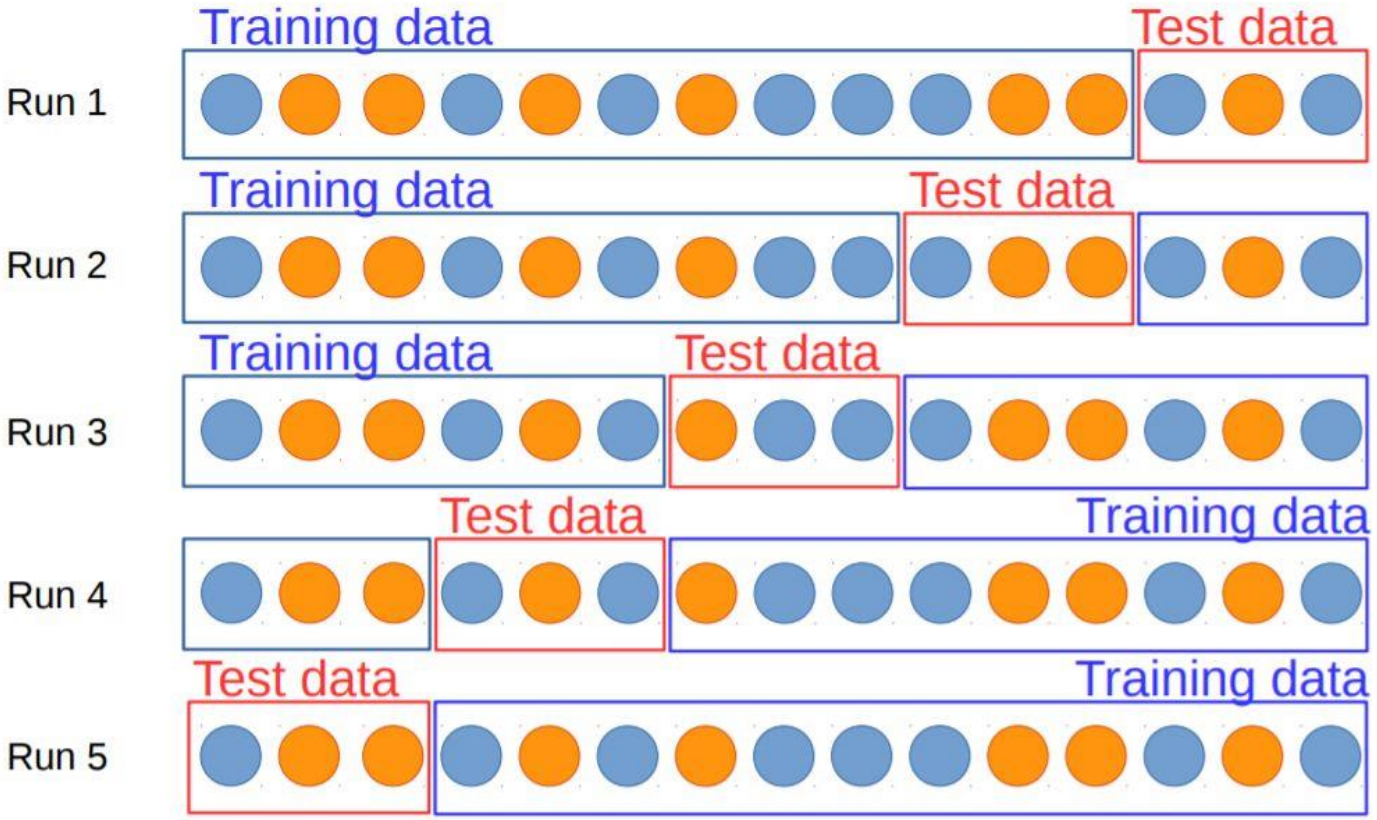


K-FOLD CROSS VALIDATION



OXFORD BROOKES UNIVERSITY

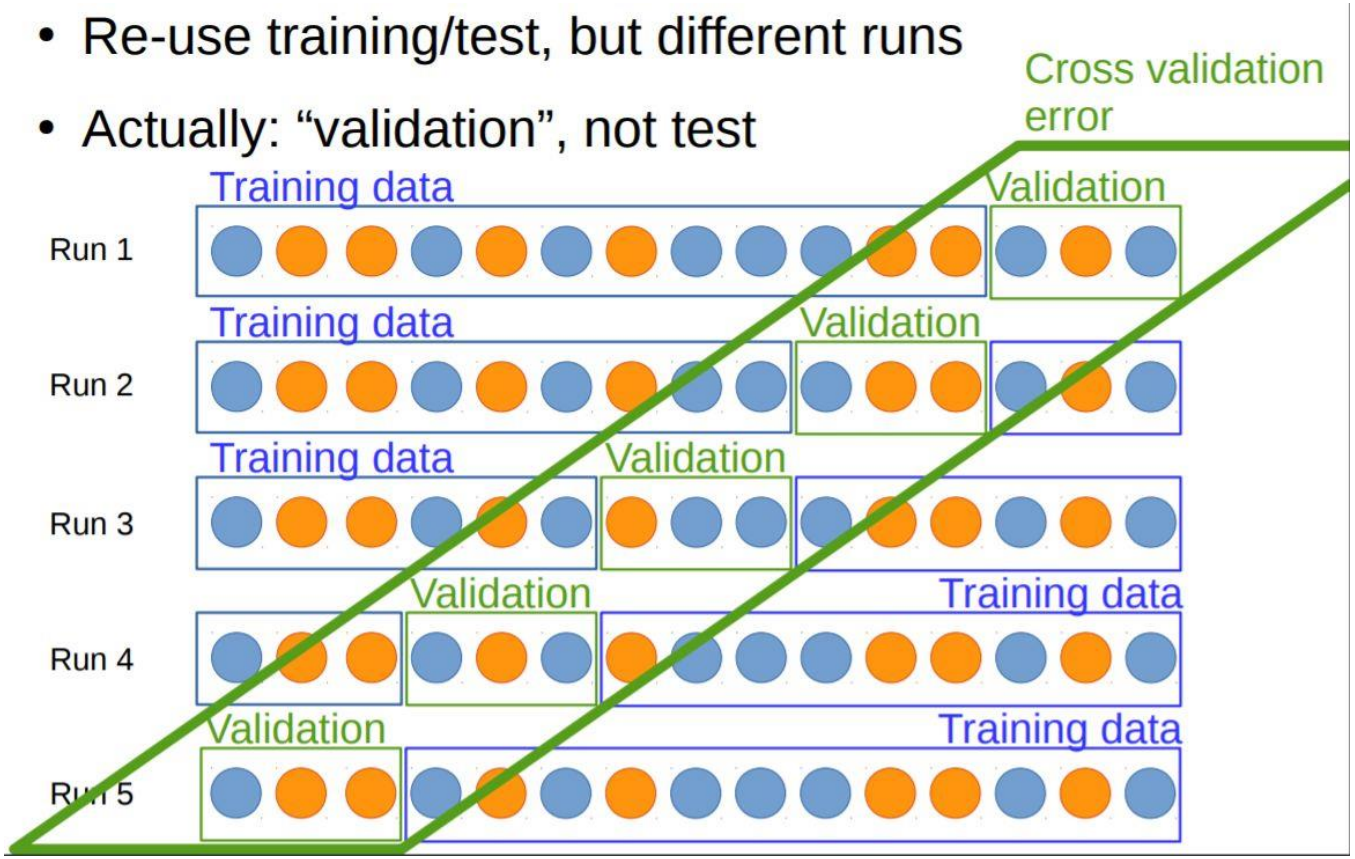
- Re-use training/test, but different runs



OXFORD BROOKES UNIVERSITY

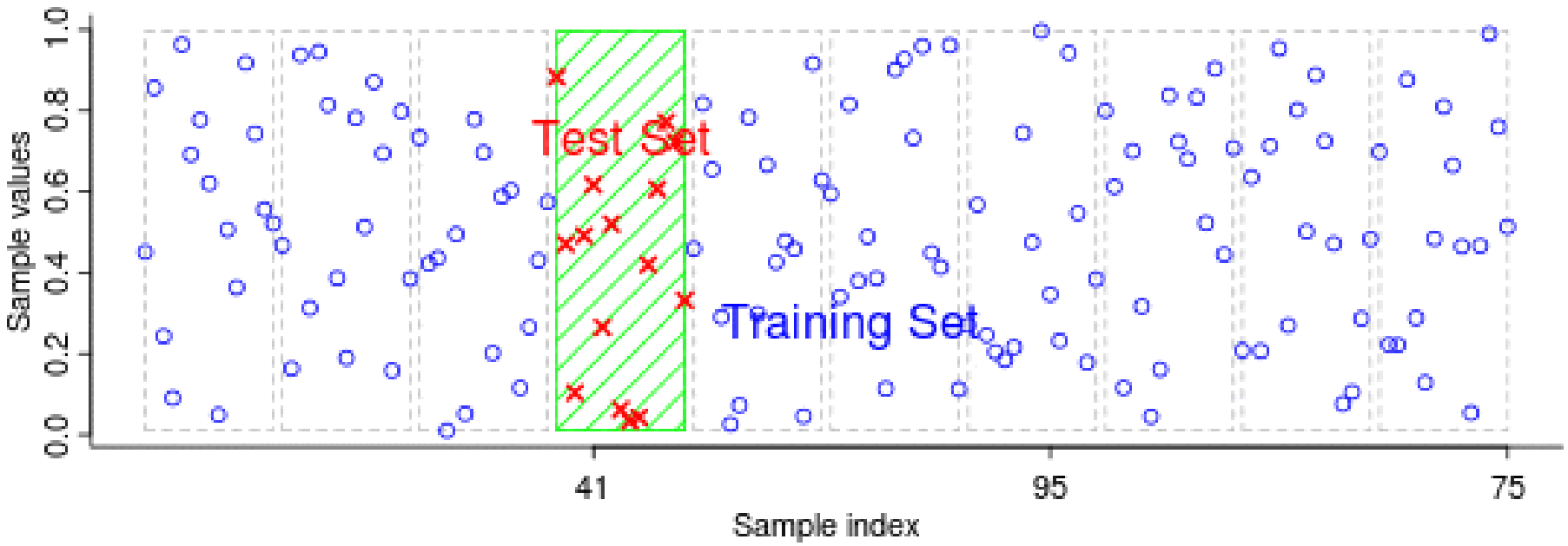
K-FOLD CROSS VALIDATION

- Re-use training/test, but different runs
- Actually: “validation”, not test



OXFORD BROOKES UNIVERSITY

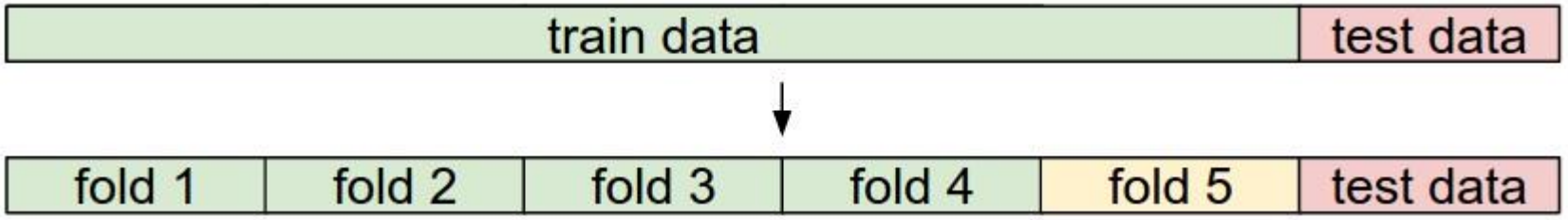
K-FOLD CROSS VALIDATION



OXFORD BROOKES UNIVERSITY

K-FOLD CROSS VALIDATION

- If the number of hyperparameters is large you may prefer to use bigger validation splits.
- If the number of examples in the validation set is small (perhaps only a few hundred or so), it is safer to use cross-validation.
- Typical number of folds you can see in practice would be 3-fold, 5-fold or 10-fold cross-validation.

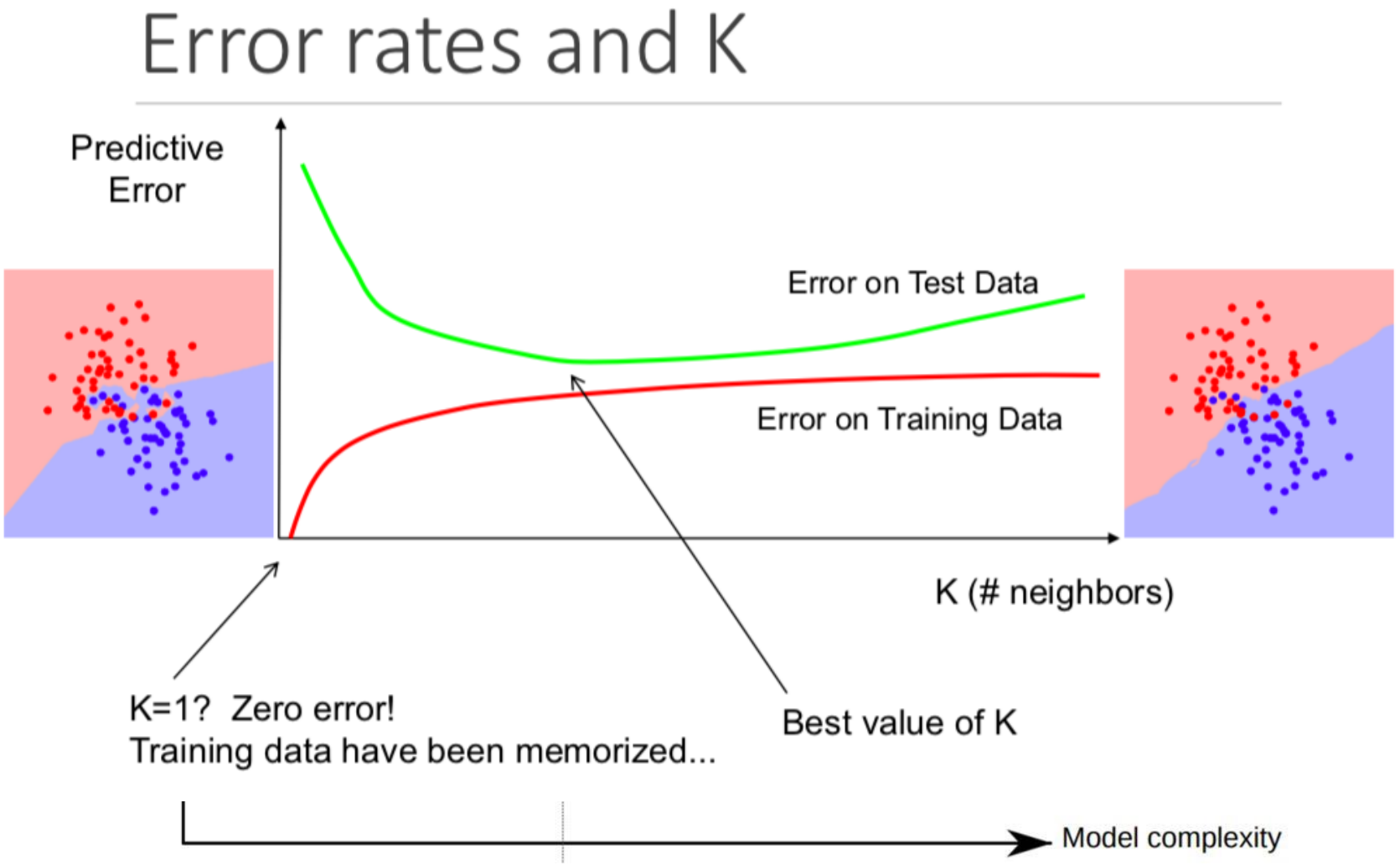


OXFORD BROOKES UNIVERSITY

HOW TO CHOOSE K?

- For a small k, we have a higher selection bias but low variance in the performances.
- For a large k, we have a small selection bias but high variance in the performances.
- Two extreme cases :
- *k = 2: two samples similar to 50-50 example. We build model only on 50% of the population each time. But as the validation is a significant population, the variance of validation performance is minimal.*
- *k = number of observations (n): "Leave one out". We have n samples and modelling repeated n number of times leaving only one observation out for cross validation. Hence, the selection bias is minimal but the variance of validation performance is very large.*
- Generally a value of k = 10 is recommended for most purpose.

MODEL SELECTION



OXFORD BROOKES UNIVERSITY

VALIDATION AND TESTING

- Keep **completely separate** test data
- Select model based on training/validation
- Possibly cross validation
- Determine eventual performance on final test data