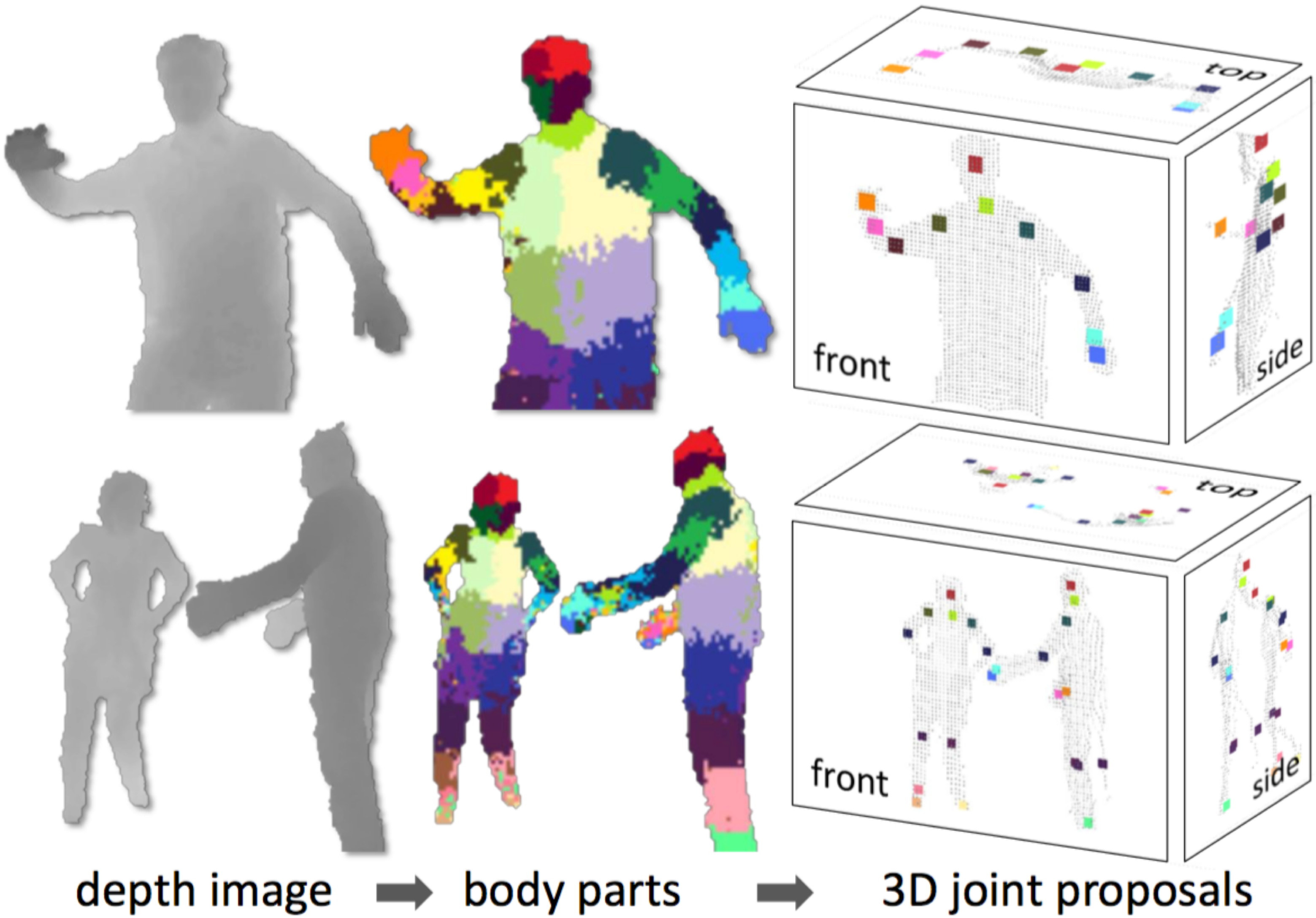
CSCM77
Structured Light & 3D Pose: Part 2
(a study on Kinect system)

Prof. Xianghua Xie

x.xie@swansea.ac.uk

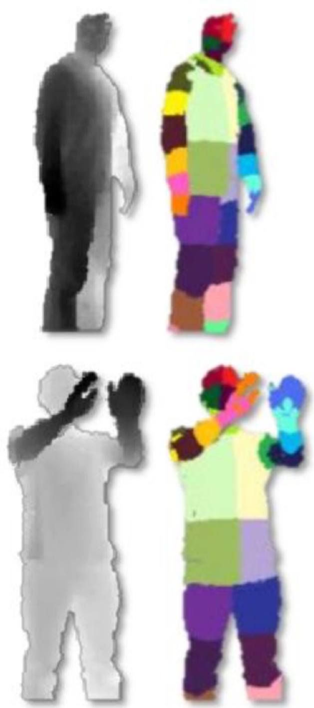
http://csvision.swan.ac.uk

From depth to 3D joint positions



Body part estimation: classification

- RECAP: Those depth feature, together with their body part labels, are fed into a Classifier
  - Random Forests classifier



Training

Testing

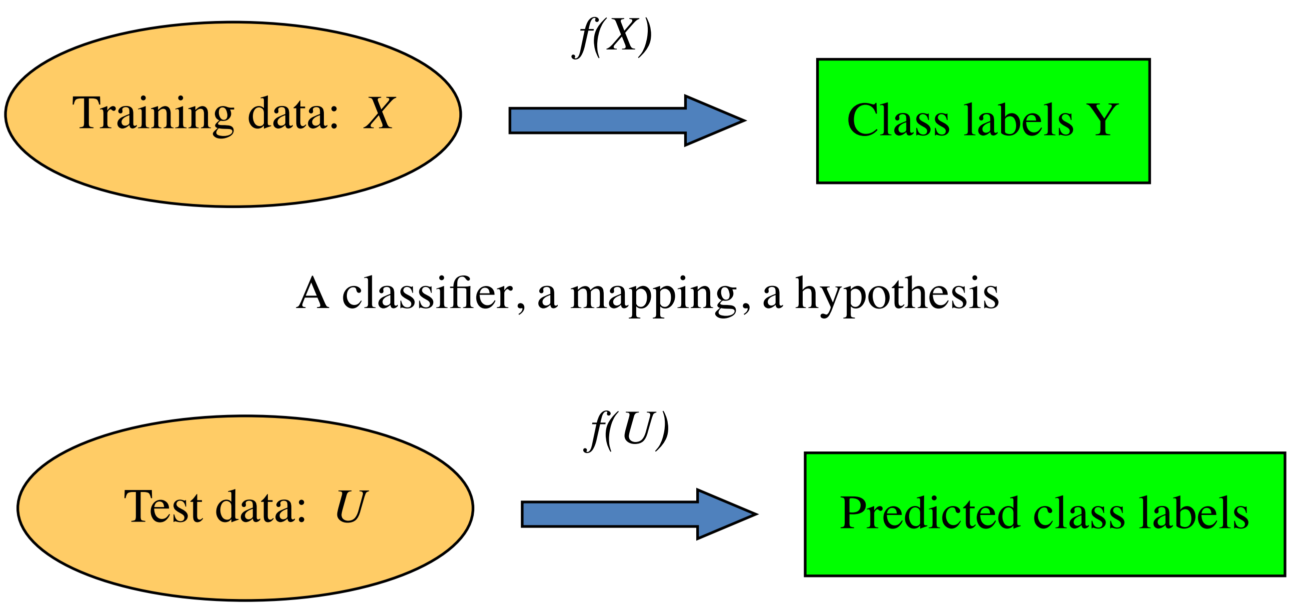
Classification

- Clustering: unsupervised learning
  - Class labels of the data are unknown
  - Given data, the task is to establish the existence of classes or clusters in the data

- Classification: supervised learning
  - Supervision: data (observations) are lablled with pre-defined classes
  - The input data (training set) consists of multiple records, each of which has multiple attributes or features
  - Given training data, the task is
    - to develop an accurate description or model for each class using the features
    - and to predict categorical class label for unseen data (test data)

Classification

- Supervised learning
  - Training data (X_i,Y_i), X_i is typically a feature vector and Y_i is the corresponding class label
  - The task of training is to find a good mapping function f
  - The derived function is then evaluated on test data (unseen data)



Classification

- Evaluation
  - Measure performance on independent blind test data
  - LOOCV: leave one out cross validation
    - Take one sample from the dataset as the test data and the remaining for training
    - Rotate the test data across the whole dataset
    - Performance is measured as the average across the rounds
  - K-fold cross validation:
    - Divide the dataset into K even parts
    - K-1 parts used for training, and one for testing
    - Rotating the test set
    - Performance is measured as the average across the rounds
  - 2-fold cross validation
    - Simplest variation of K-fold
    - Training and testing has equal number of samples

Classification

- Evaluation metrics
  - Positive: data sample that belongs to the class of interest
  - Negative: data sample that does not belong to the class
  - True positive: a positive sample correctly identified as positive
  - True negative: a negative sample correctly identified as negative
  - True positive rate: sensitivity
    - Percentage of correct prediction of positive samples
  - True negate rate: specificity
    - Percentage of correct prediction of negative samples
  - Accuracy
    - Percentage of correct prediction
  - Error rate
    - Percentage of incorrect prediction

Classification

- A computer vision example:
  - Predict gender from face detection
  - Class label, {male, female}

http://vimeo.com/51210467

Classification

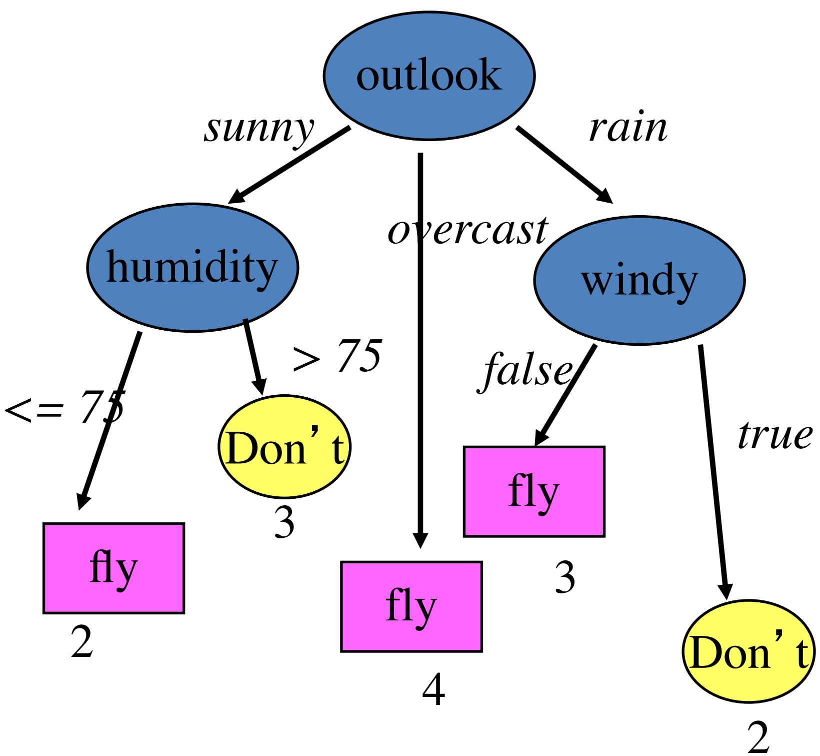
- A simple learning example: learn prediction of "safe conditions to fly"
  - Based on weather conditions, i.e. attributes
  - Classification problem, class = {yes, no}

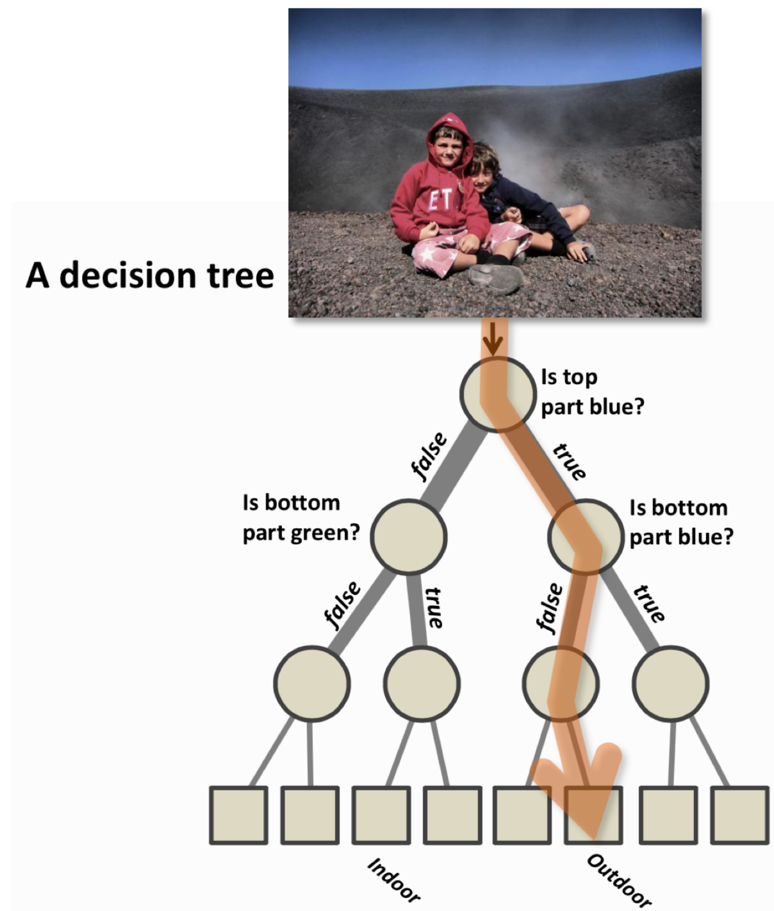
Attributes **Classification**

Outlook	Temperature	Humidity	Windy	Fly
Sunny	85	85	False	No
Sunny	80	90	True	No
Overcast	83	86	False	Yes
Rainy	75	80	False	Yes
...	...	...	...	...

Decision Trees

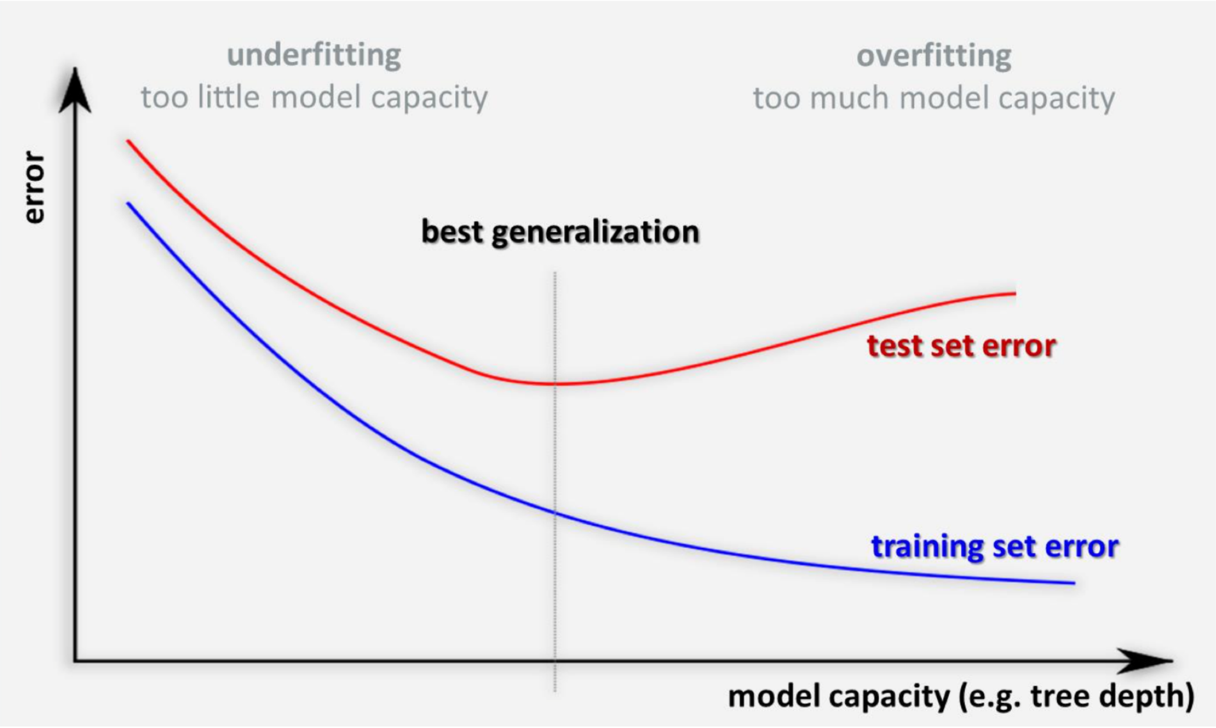
- Rule based induction algorithms
- Easy to interpret, but generally poor accuracy
  - Tend to overfit the data; poor generalisation.





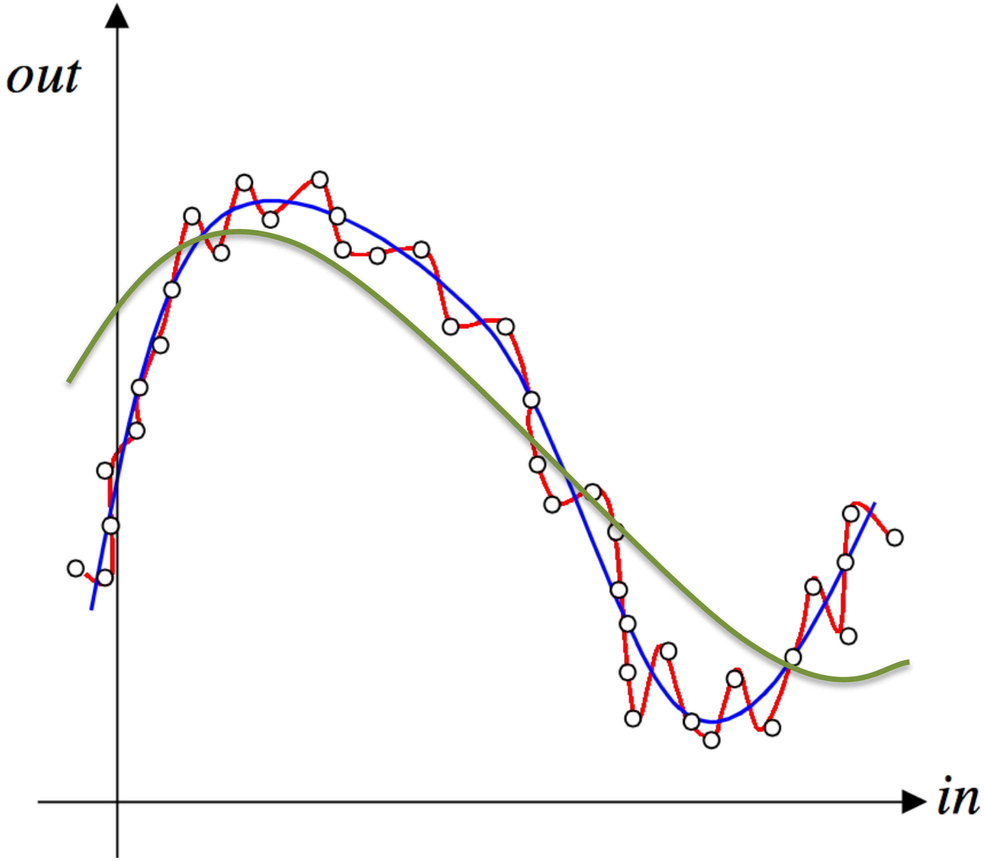
Classification

- Underfitting
  - Model is not optimised
  - Large classification errors on both training data and testing data
- Overfitting
  - Poor generalisation
  - Training error no longer providers a good estimation of testing error



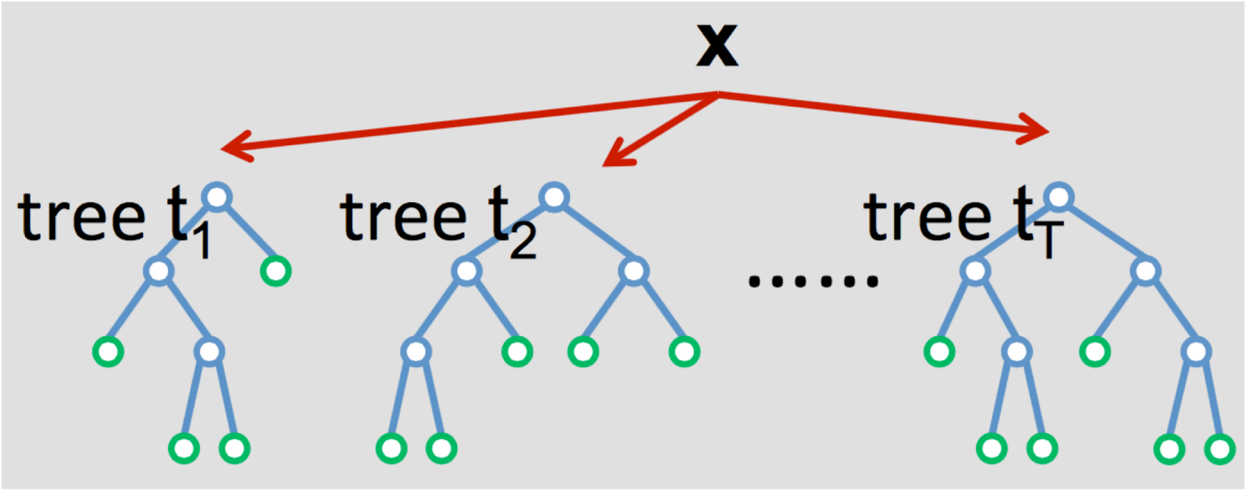
Classification

- Example
  - Which curve is underfitting the data?
  - Which one is overfitting the data?



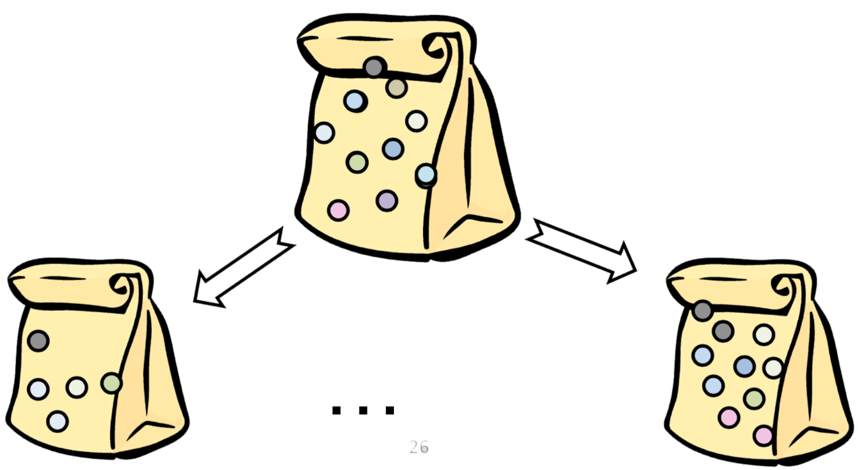
Random Forests

- Conventional decision trees are rarely used in solving computer vision problems
- Random Forests (in one sentence):
  - Ensemble of bagged decision tree learners with randomised feature selection
    - Key concepts: ensemble, bagging, randomisation
  - Efficient, good performance in general



Random Forests

- Bagging: Bootstrap AGGregatING
  - Given a dataset, it generates multiple subsets;
  - Each subset is generated by subsampling the original dataset uniformly with replacement
  - To improve stability and reduce overfitting, via randomisation on data

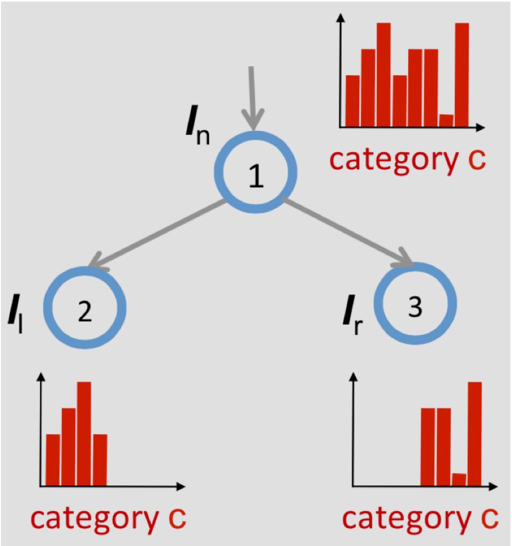


Random Forests

- Strong classifier
  - Output strongly correlates to correct classification
- Weak classifier
  - Output weakly correlates to correct classification
- Ensemble
  - Combining multiple (weak) classifiers
- How to combine?
  - Bagging
    - Train a (often large) number of classifiers on random subsets of training set; classify using majority vote of all classifiers
  - Boosting
    - As per bagging, but introduce weights for each classifier based on performance over the training set

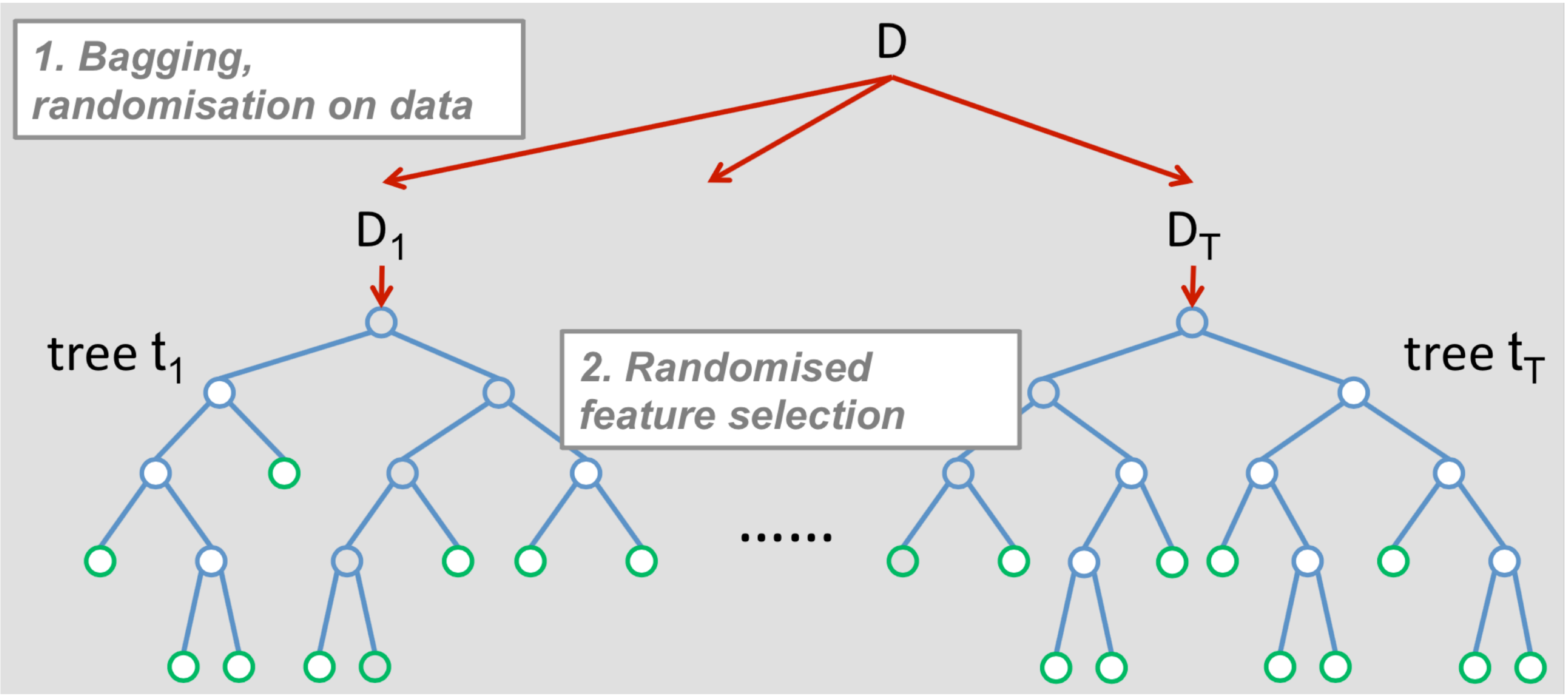
Random Forests

- Randomisation
  - Randomisation on data, at the bagging stage
  - Randomisation on feature selection
  - Both to avoid overfitting
- Randonmised feature selection
  - Split training data I_n that reaches node n, based on threshold t
  - Feature f and threshold t are chosen at random
  - Select the best (f,t) that maximises the information gain
    - Left and right nodes are mostly different to each other



Random Forests

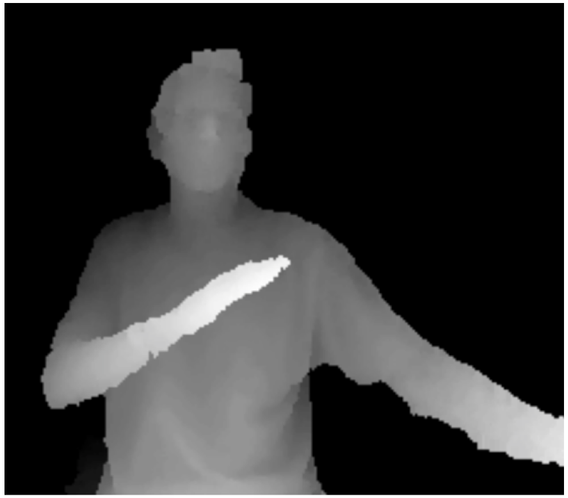
- To summarise: forest is an ensemble of bagged decision trees with randomised feature selection

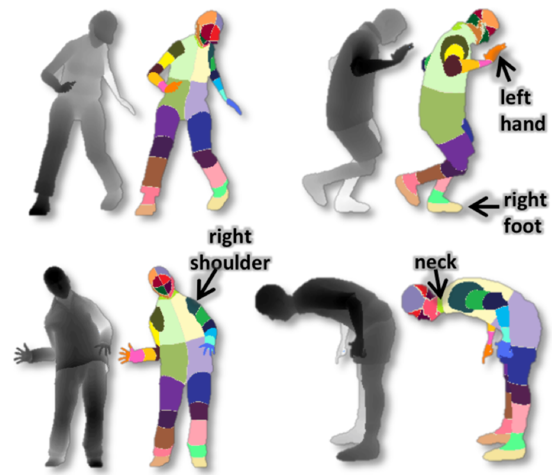


Random Forests

- Example: Kinect pose estimation

Input depth image





Labbelled training data

Testing data

Random forests result

Summary

- Clustering vs. Classification
- Evaluation methods
- Evaluation metrics
- Decision trees
- Overfitting and underfitting
- Bagging learning
- Ensemble classifier
- Radomisation to avoid overfitting
- Randomised decision trees