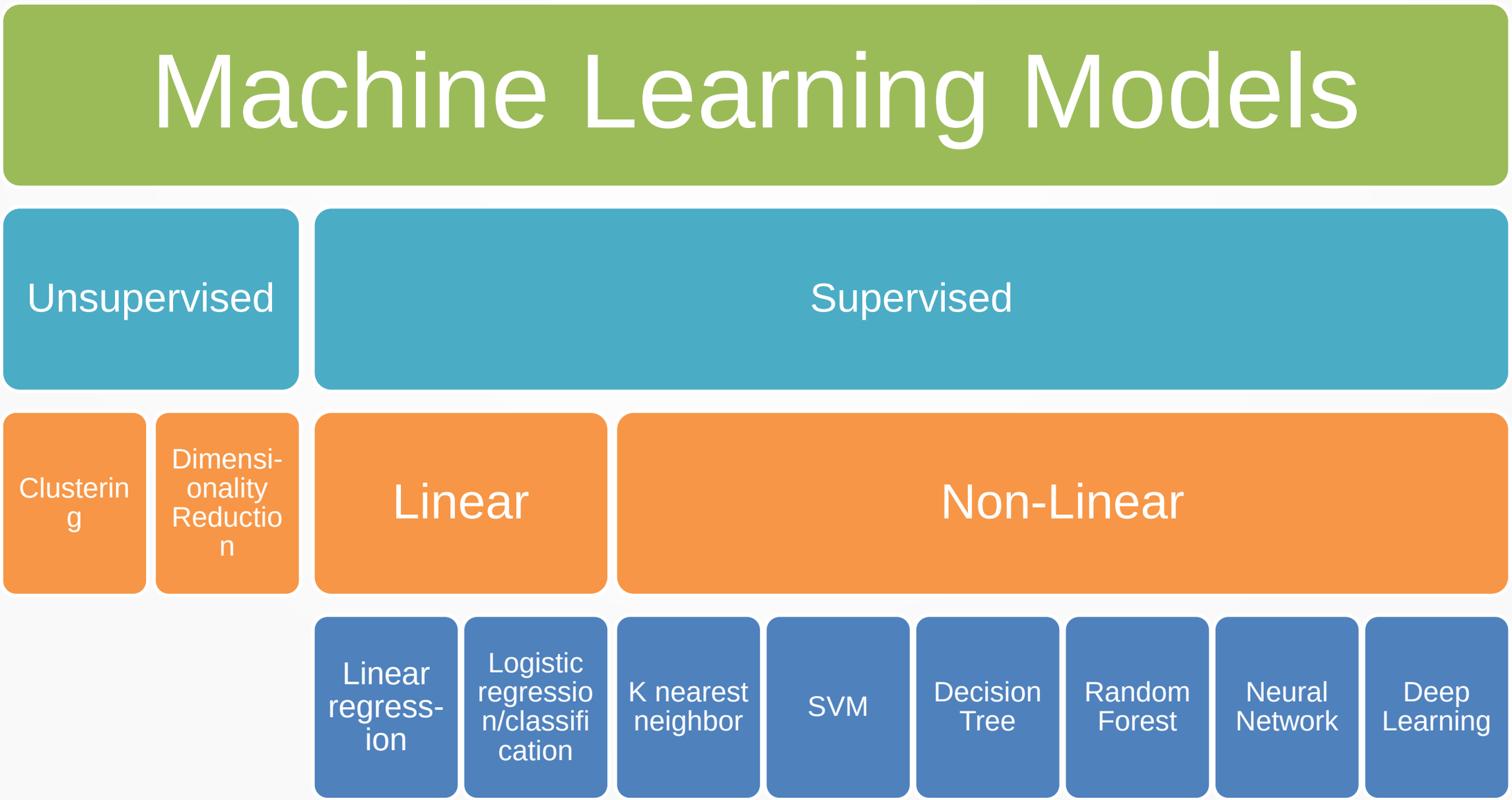
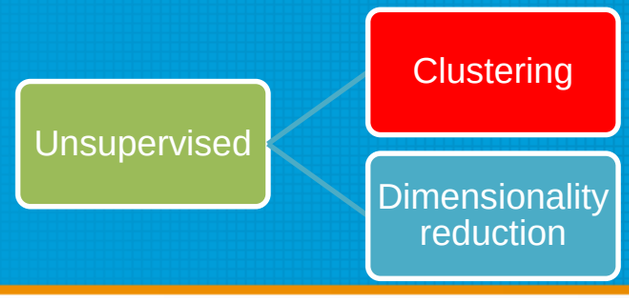
Machine Learning – Take home messages

Week 11

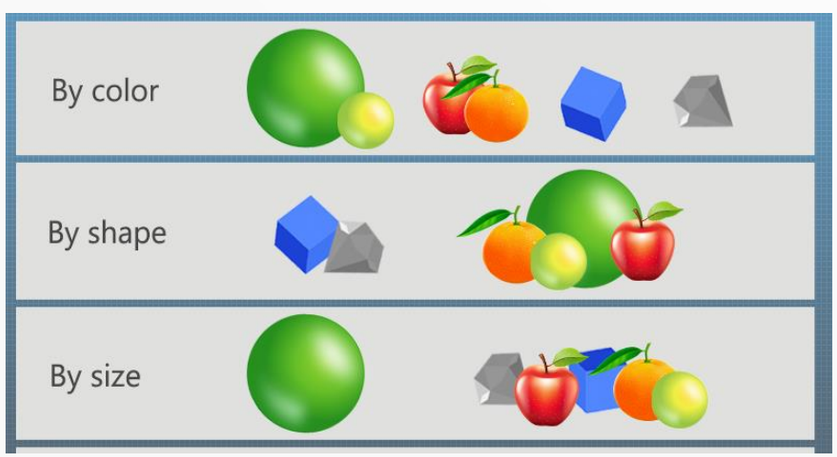
Recap of the unit ...



Recap of the unit ...

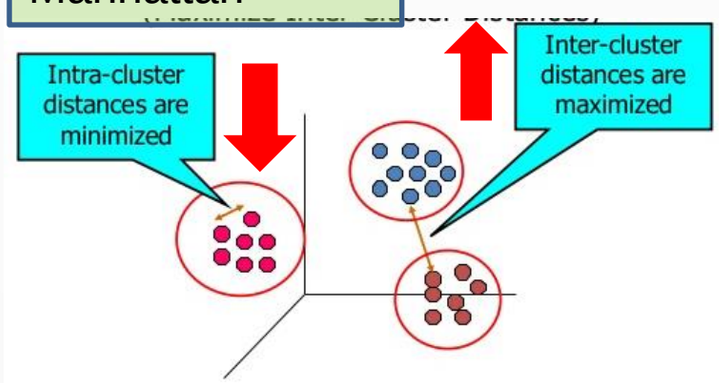


Cluster can vary based on the properties



Distance measures

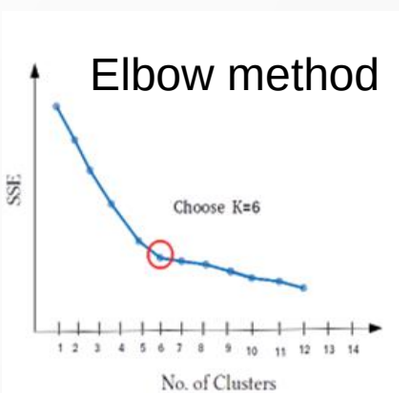
Euclidean
Mahalanobis
Jaccard
Cosine
Manhattan



Algorithms

Kmeans++

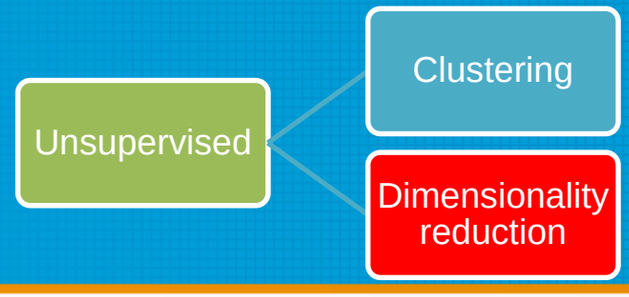
KMeans



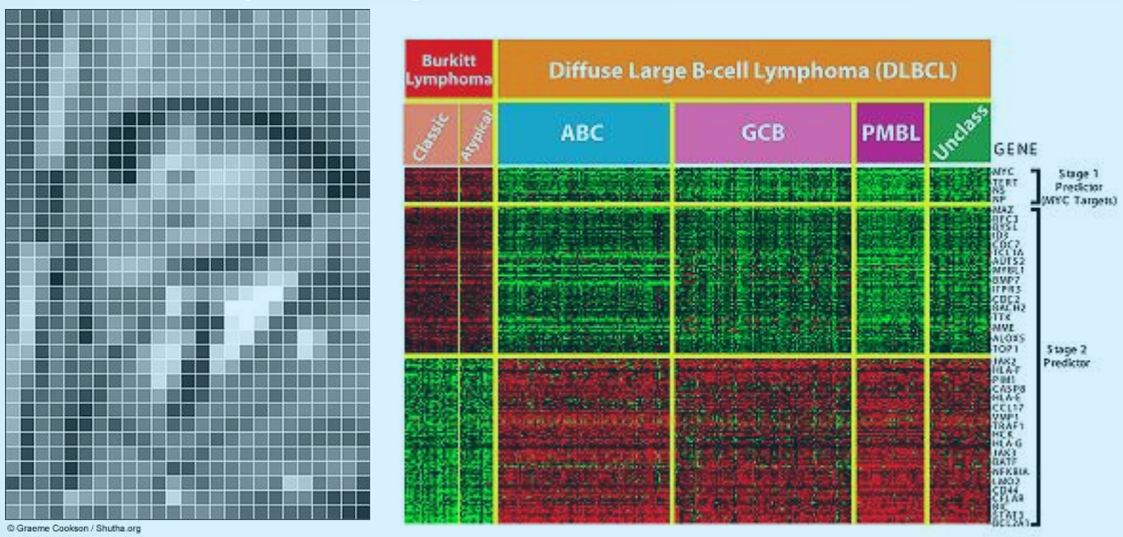
Performance measures

Rand Index
Purity
Mutual Information
Silhouette Coefficient

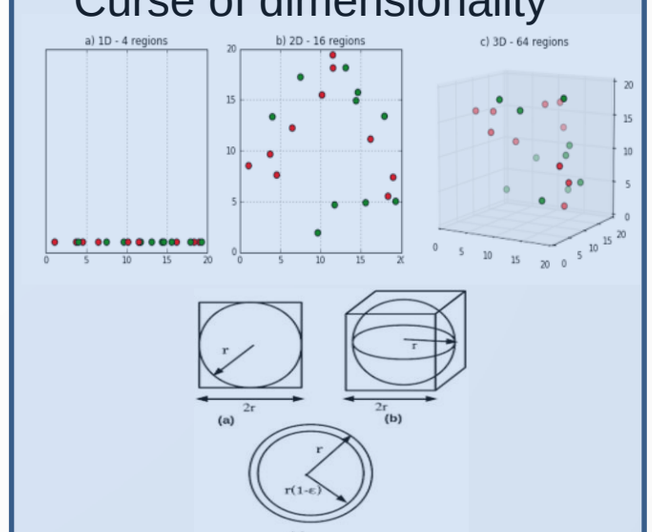
Recap of the unit ...



Example of high dimensional data



Curse of dimensionality



Principal Component Analysis

Correlated High Dimensional Data → Compute Covariance Matrix → Eigenvectors + Eigenvalues → Pick n<d eigenvectors with highest eigenvalues → Project data points to those eigenvectors → Uncorrelated Low Dimensional Data

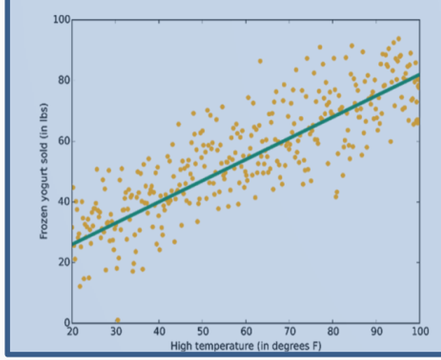
- Linear combinations of all dimensions (w>0)
- Ranked based on described variance
- Drop dimension with very low variance

- Impossible to find effect of individual attributes
- Limits importance of hand-crafted features
- Reduce clarity of domain knowledge

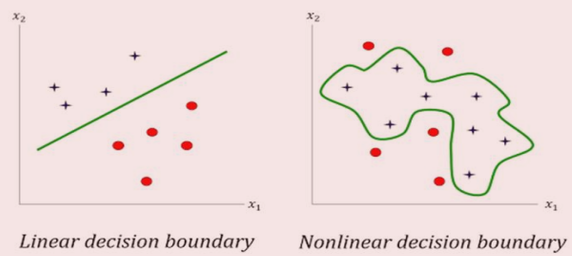
Recap of the unit ...

- Supervised
  - Linear
  - Nonlinear

Regression



Classification



What we want to learn?

$$h : X \rightarrow Y$$

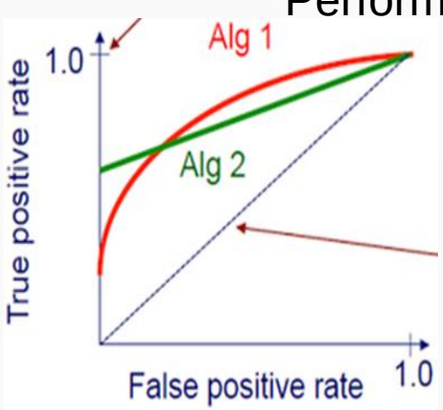
How the machine learns?

$$\min_{h \in H} \frac{1}{n} \sum_{i=1}^{n} L(y_i, h(x_i))$$

Model complexity

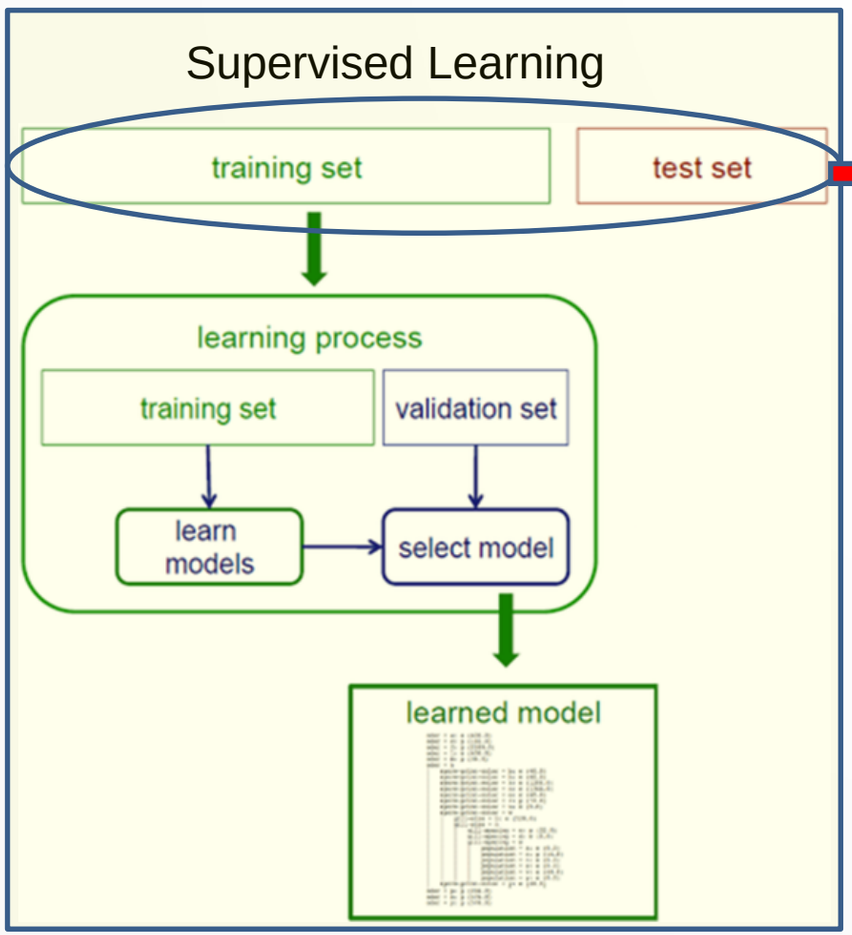
$$R_{str}(h) = R_{emp}(h) + \lambda C(h)$$

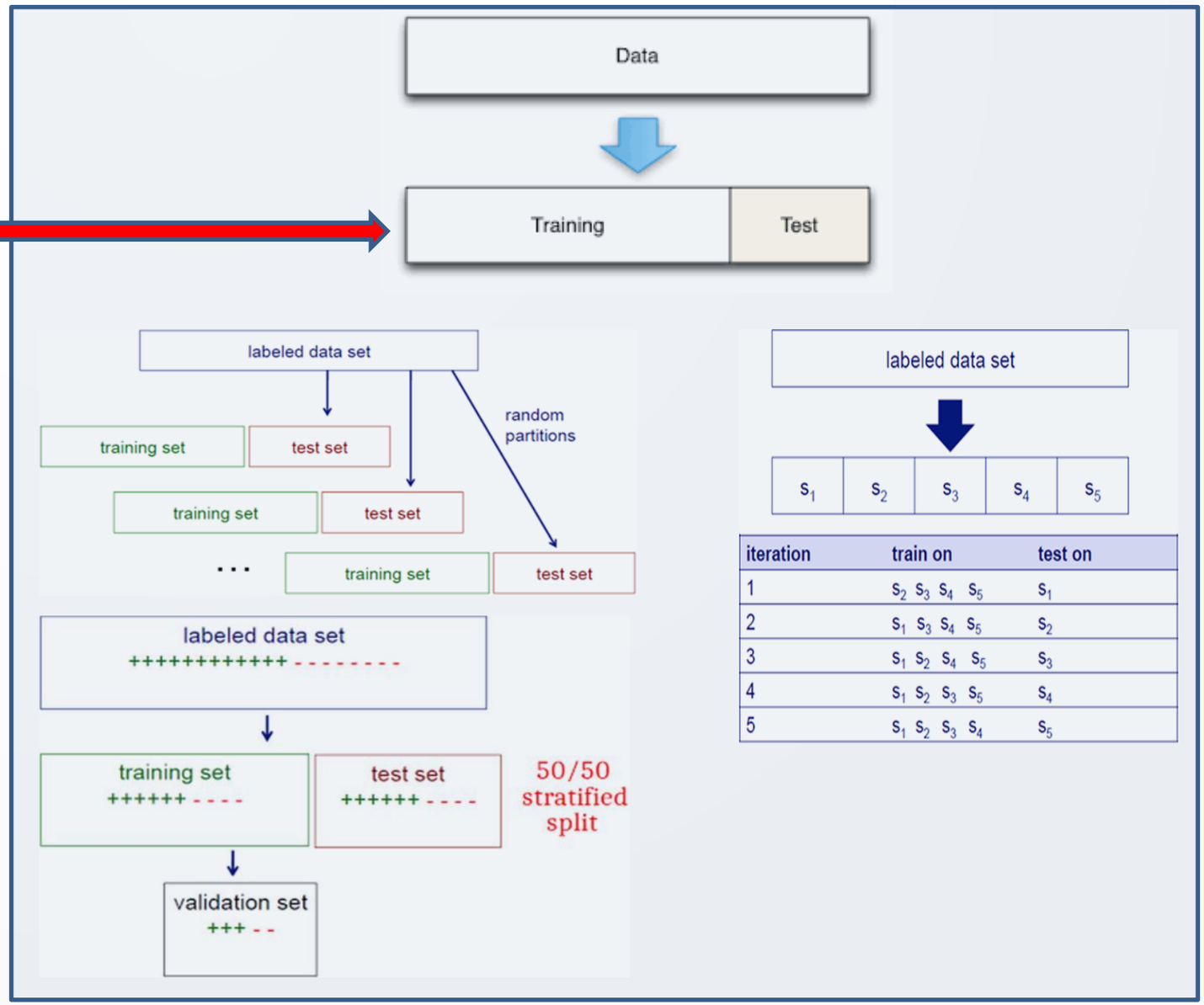
Performance Measurements



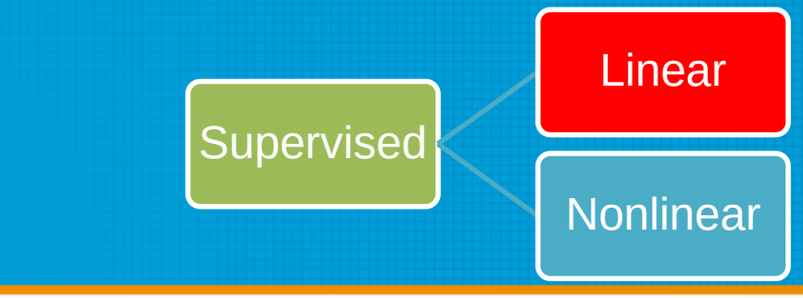
		Predicted Class: Positive	Predicted Class: Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error	Sensitivity $\frac{TP}{(TP+FN)}$
	Negative	False Positive (FP) Type I Error	True Negative (TN)	Specificity $\frac{TN}{(TN+FP)}$
		Precision $\frac{TP}{(TP+FP)}$	Negative Predictive Value $\frac{TN}{(TN+FN)}$	Accuracy $\frac{TP+TN}{(TP+TN+FP+FN)}$

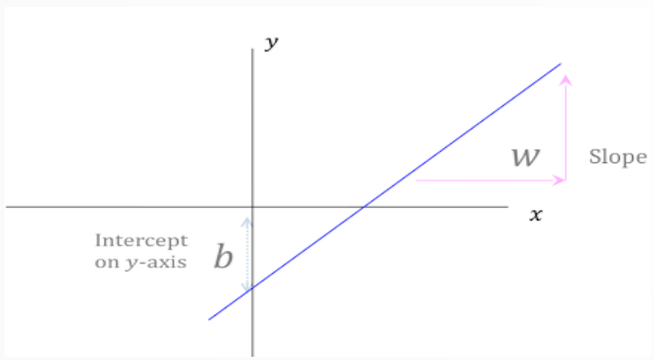
Recap of the unit ...





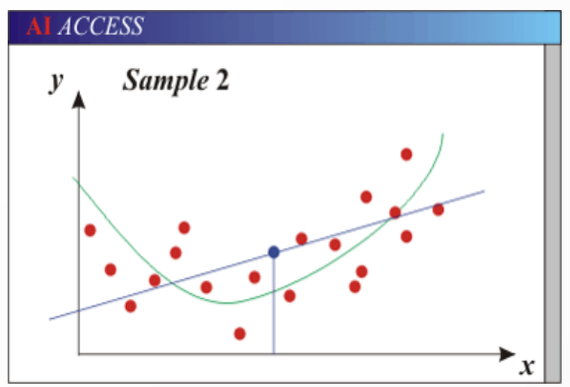
Recap of the unit ...

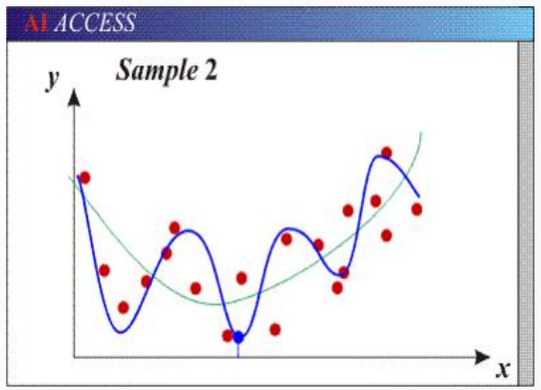




$$L(y_i, \mathbf{x}_i^T\mathbf{w}) = (y_i - \mathbf{x}_i^T\mathbf{w})^2$$

$$L(y_i, \mathbf{x}_i^T\mathbf{w}) = log(1 + exp(-y_i\mathbf{x}_i^T\mathbf{w}))$$

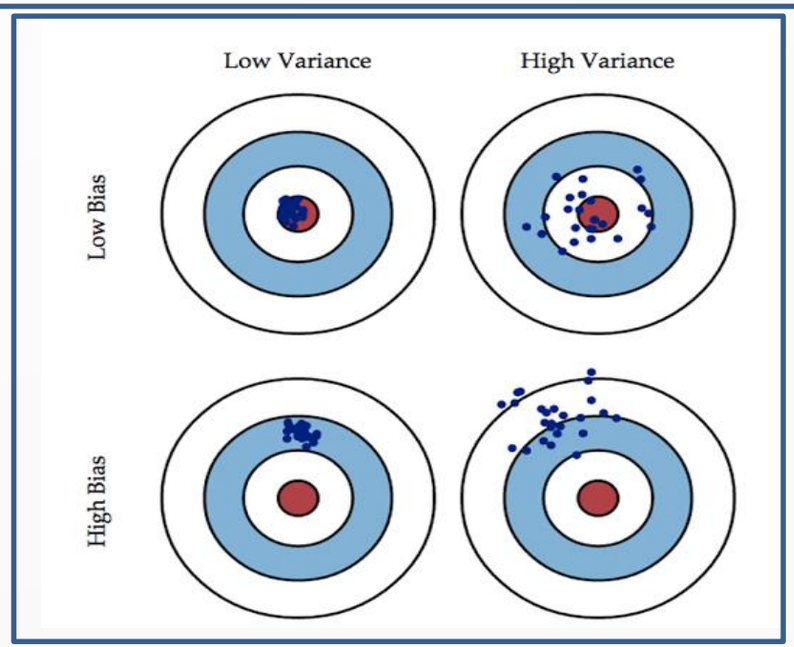




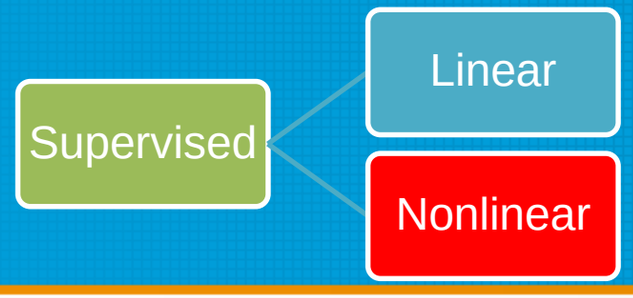
$$\min_w \frac{1}{n}\sum_i L(y_i, \mathbf{x}_i^T\mathbf{w}) + \lambda_1||\mathbf{w}||_1$$

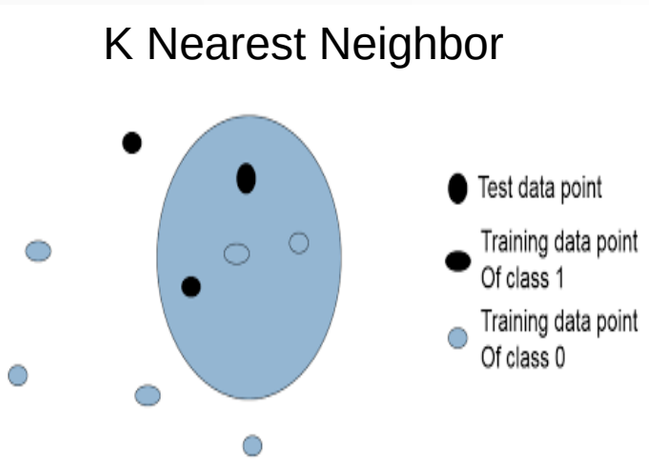
$$\min_w \frac{1}{n}\sum_i L(y_i, \mathbf{x}_i^T\mathbf{w}) + \lambda_2||\mathbf{w}||_2^2$$

$$\min_w \frac{1}{n}\sum_i L(y_i, \mathbf{x}_i^T\mathbf{w}) + \lambda_1||\mathbf{w}||_1 + \lambda_2||\mathbf{w}||_2^2$$



Recap of the unit ...

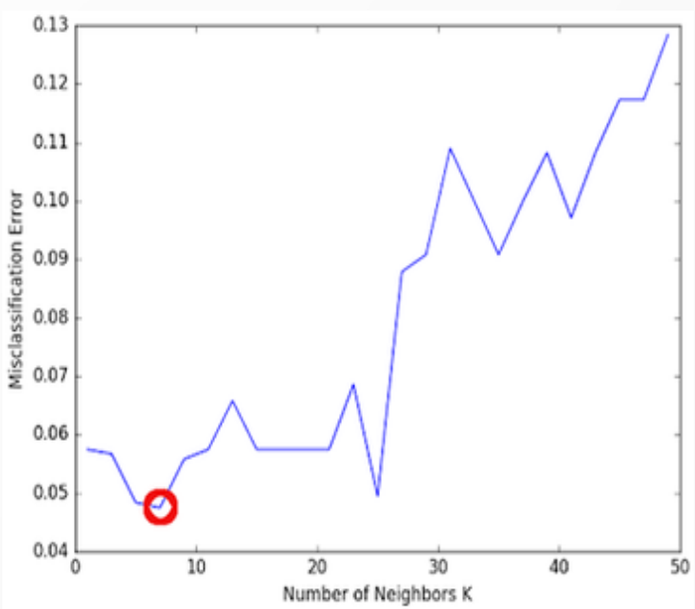




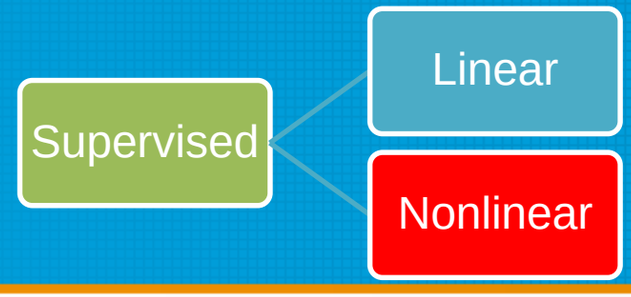
- Small values of K
  - Restrains the region of a given prediction
  - Forces classifier to be more focused on the close regions and neighbours
    - This will result in a low bias and high variance

- Higher values of K
  - Asking for more information from distant training points
  - Smoother decision boundaries
    - Lower variance but increases bias

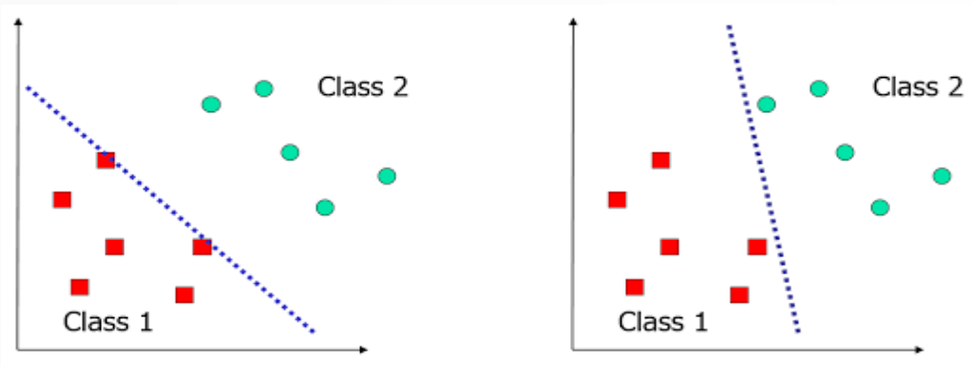
- Finding the best K
  - There is no rule of thumb in selecting K_{max} since it depends on your desired rate of exploration for K
  - A simple and handy method
    - Cross-validation to partition your data into test and training samples
    - Evaluate model with different ranges of K values
    - The misclassification error can be used as a measurement of performance

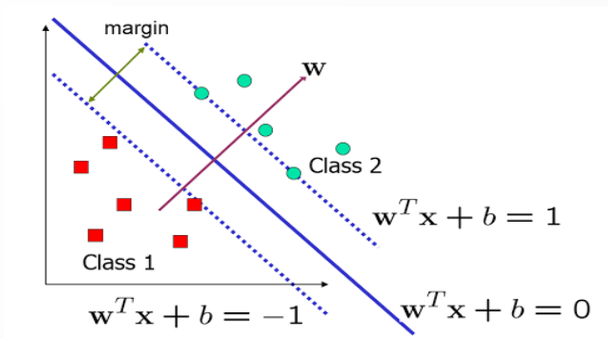


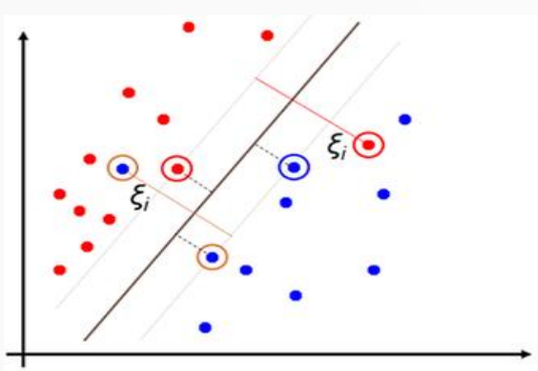
Recap of the unit …



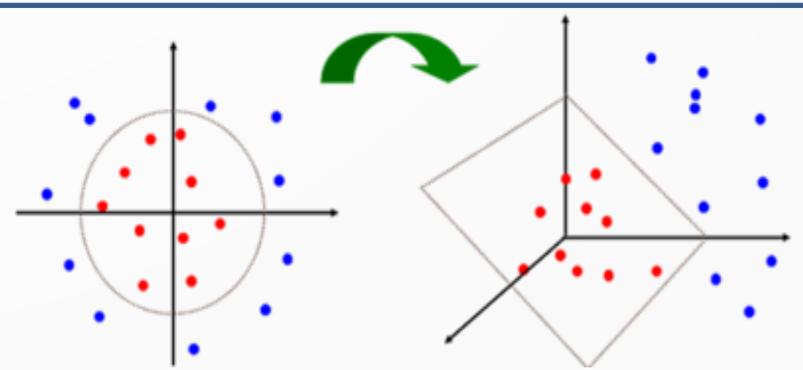
$$\min_\alpha \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{i=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$
$$\text{subject to: } \sum_{i=1}^{n} \alpha_i y_i = 0 \quad and \quad \alpha_i \geq 0 \quad for\ all\ i$$







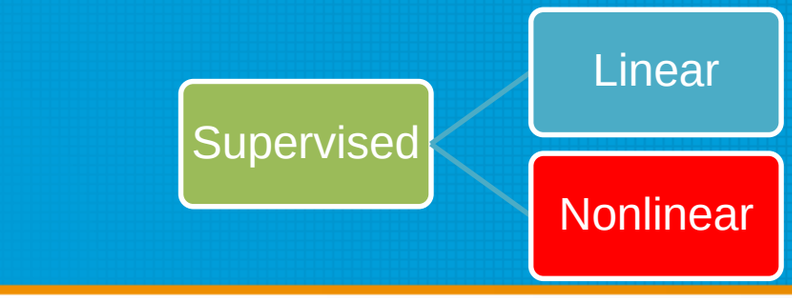
- Nonlinearly separable data
- Kernel



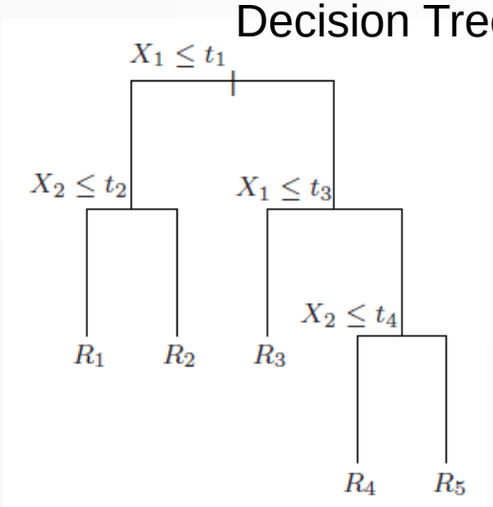
$$\min_\alpha \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{i=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \Leftrightarrow K(X_i, X_j)$$

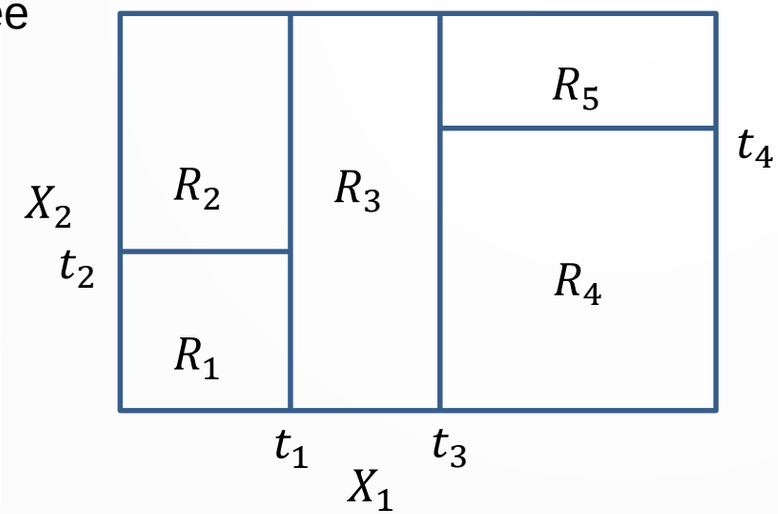
$$Subject\ to: \sum_{i=1}^{n} \alpha_i y_i = 0 \quad and \quad 0 \leq \alpha_i \leq C \quad \forall i$$

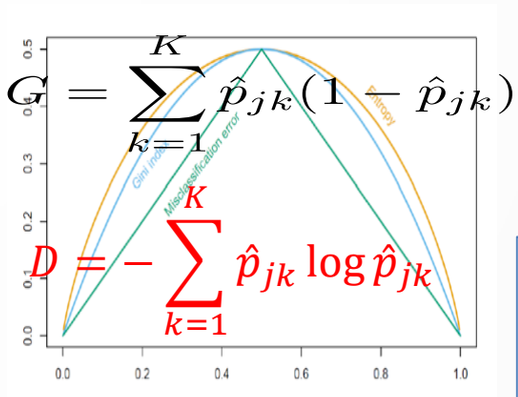
Recap of the unit ...



Decision Tree

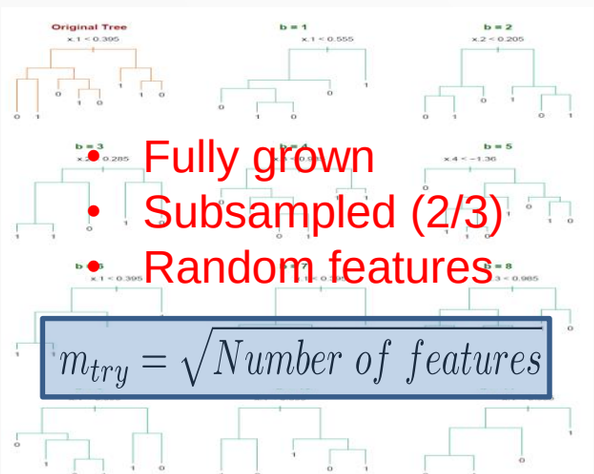






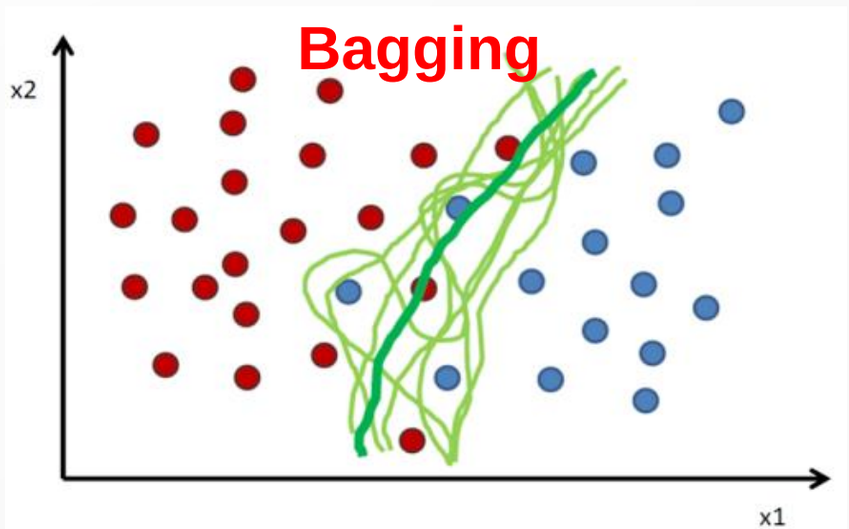
- Simple
- Explainable
- Overfitting

- Pruning
  - Pre-pruning
  - Post-pruning

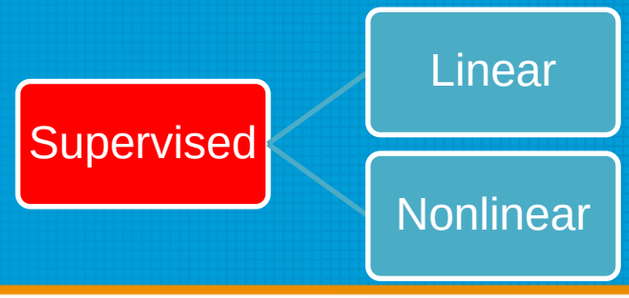


- Fully grown
- Subsampled (2/3)
- Random features

$$m_{try} = \sqrt{Number\ of\ features}$$



Recap of the unit ...



Hyperparameter

Parameter

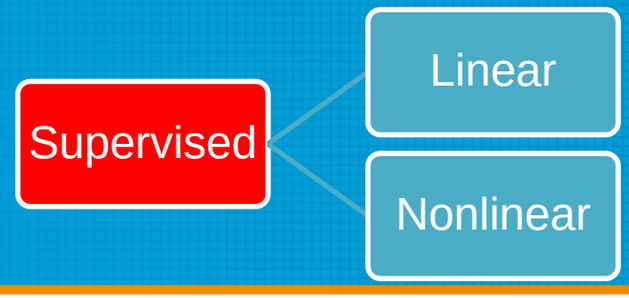
Performance measurement

Data splitting

Overfitting

Model generalisation

Recap of the unit ...



Design questions

Which ML model to use?

- Sample size
- Class balance
- Dimensionality of the data
- Application environment
- Realtime or asynchronous

How to optimize the model?

- Identify hyperparameters (especially sensitive ones)
- Do not try to use any rule of thumb
- Do not use the test data in the model selection process
- Select the best model based on cross-validation results
- Generate multiple test performance and report the aggregated results

eVALUate

- Completing the survey is an important way that Deakin listens to student voices about their experience (especially during this time)
- Responses are confidential, students are not identified, and eVALUate reports are only sent out after student results are released
- You must only give respectful and polite feedback to better represent your views
  - There is a useful set of guidelines on constructive feedback for eVALUate developed by the University of Tasmania – here.
- You can complete your eVALUate surveys on any device, from anywhere using this link https://deakin.is/evaluating-us

Thank You.