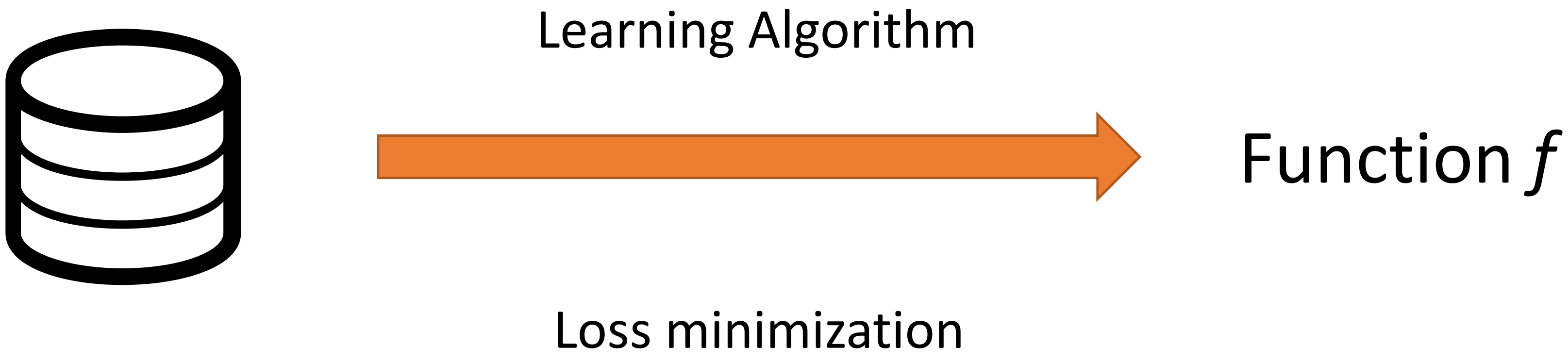
Train In Data

Parameters and Hyperparameters

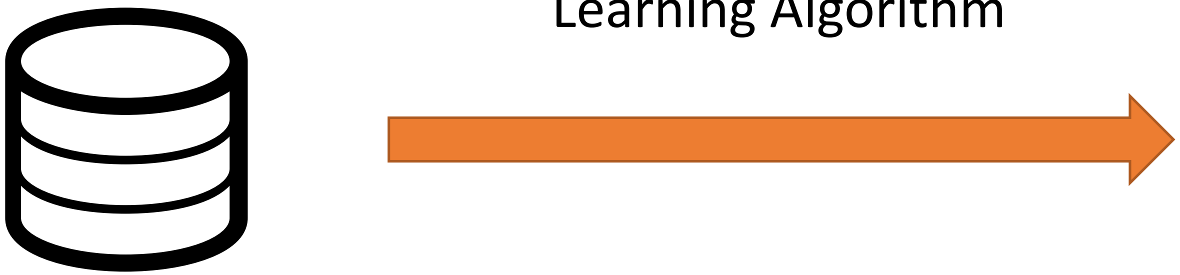
Parameters in ML models

- The objective of a typical **learning algorithm** is to find a function ***f*** that minimizes a certain **loss** over a **dataset**.



Parameters in ML models

- The **learning algorithm** produces ***f*** through the optimization of a training criterion with respect to a set of **parameters**.



Respect to certain parameters

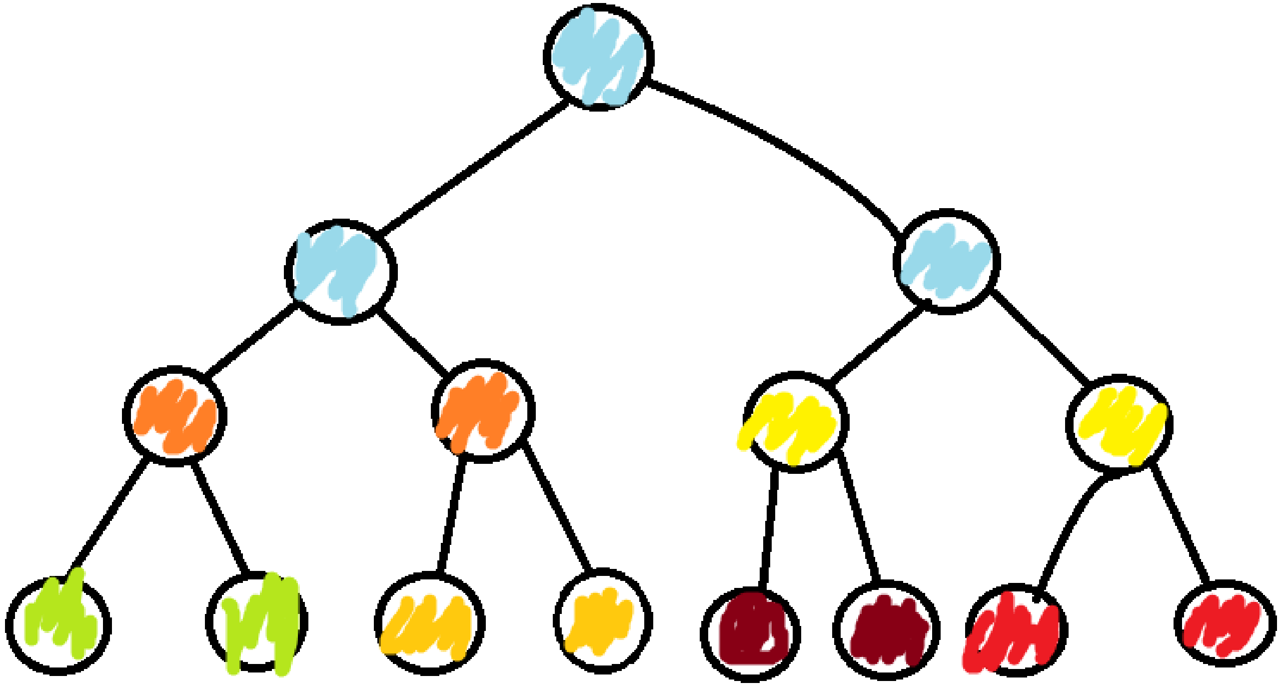
Linear Regression Parameters

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \ldots + \beta_n X_{ni} + \varepsilon_i$$

$$\begin{aligned} \mathrm{RSS}(\beta) &= \sum_{i=1}^{N} (y_i - f(x_i))^2 \\ &= \sum_{i=1}^{N} \Big(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \Big)^2. \end{aligned}$$

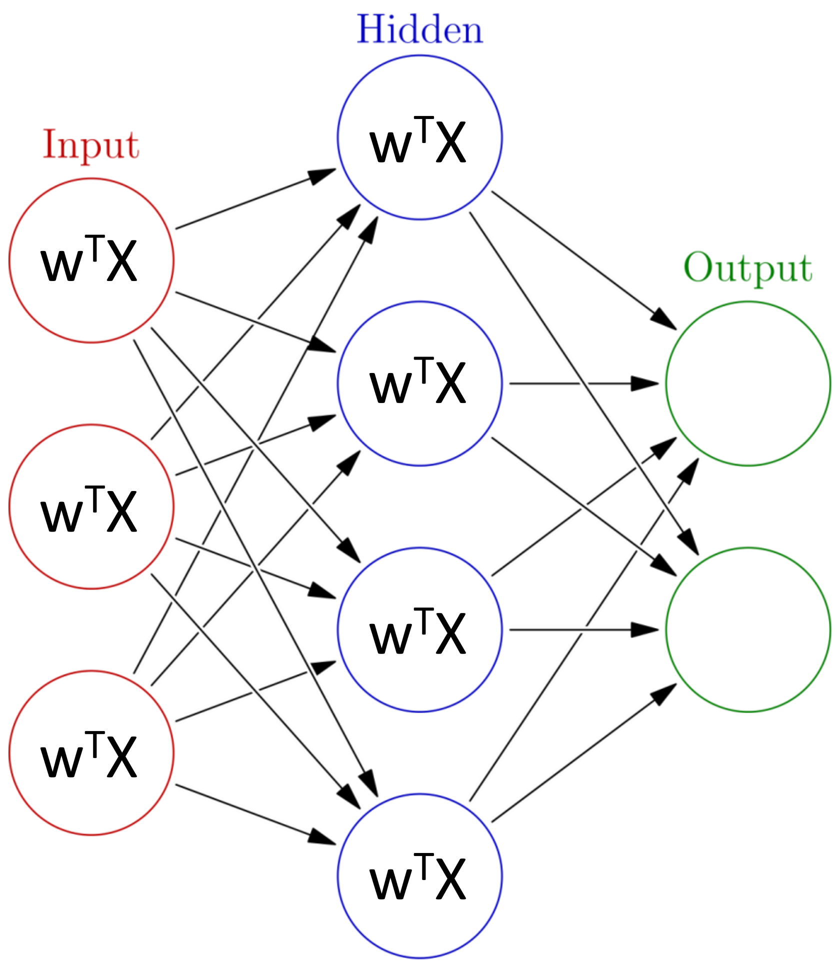
β, the coefficients of the linear function, are **the parameters** to find or optimise by the algorithm

Decision Tree Parameters



- The variable
- The split value
- The height in the tree

Neural Network Parameters



- The weights at each neuron

Hyperparameters in ML models

- Hyperparameters are parameters that are not directly learnt by the learning algorithm.

- Hyperparameters are **specified outside** of the training procedure.

- Hyperparameters **control** the *capacity* of the model, i.e., how flexible the model is to fit the data

- Prevent over-fitting

Linear Regression Hyperparameters

$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \ldots + \beta_n X_{ni} + \varepsilon_i$

$$\begin{aligned}\mathrm{RSS}(\beta) &= \sum_{i=1}^{N} (y_i - f(x_i))^2 \\ &= \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2.\end{aligned}$$

- Vanilla Linear Regression

  ➔ no hyperparameters

Regularized Linear Regression

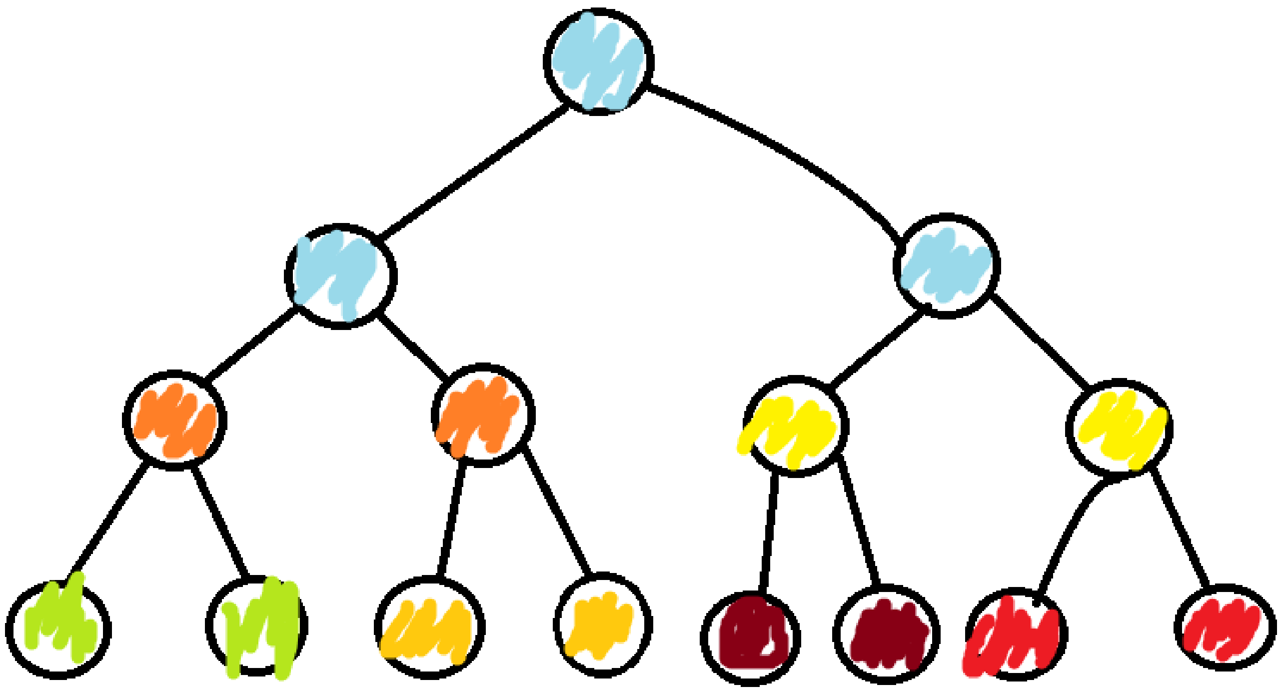
$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \ldots + \beta_n X_{ni} + \varepsilon_i$

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}.$$

$$\hat{\beta}^{\text{lasso}} = \underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}.$$

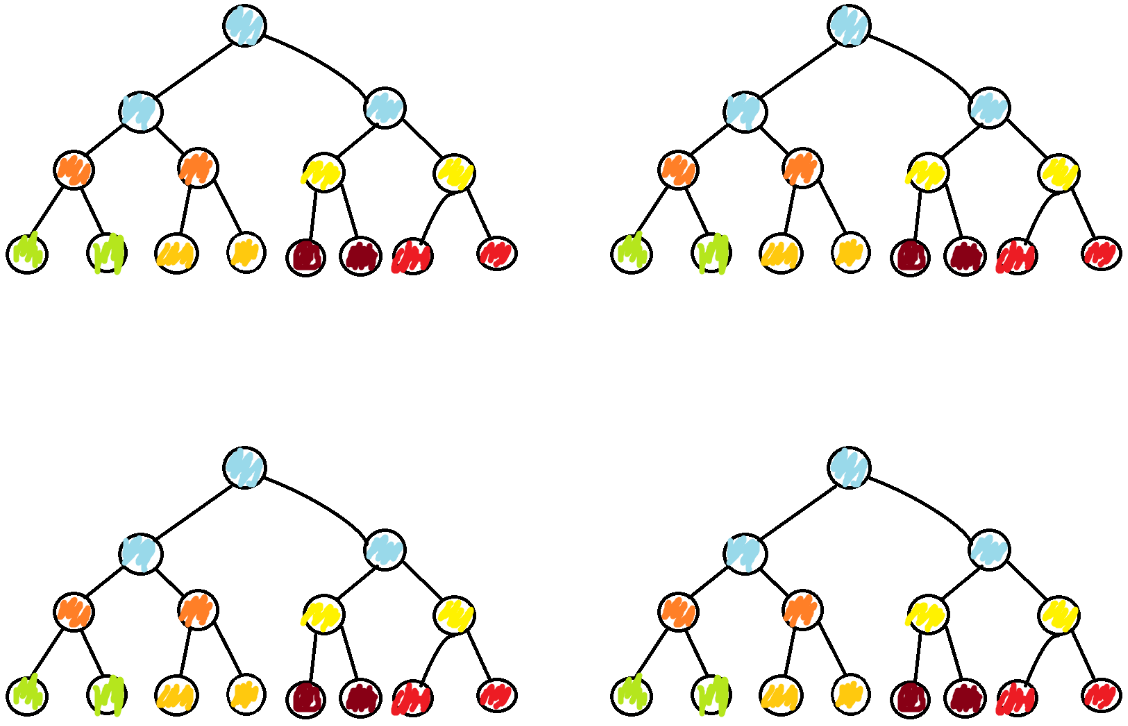
- The regularization method:
  - Lasso
  - Ridge
  - Elastic net
- The regularization penalty

Decision Tree Hyperparameters



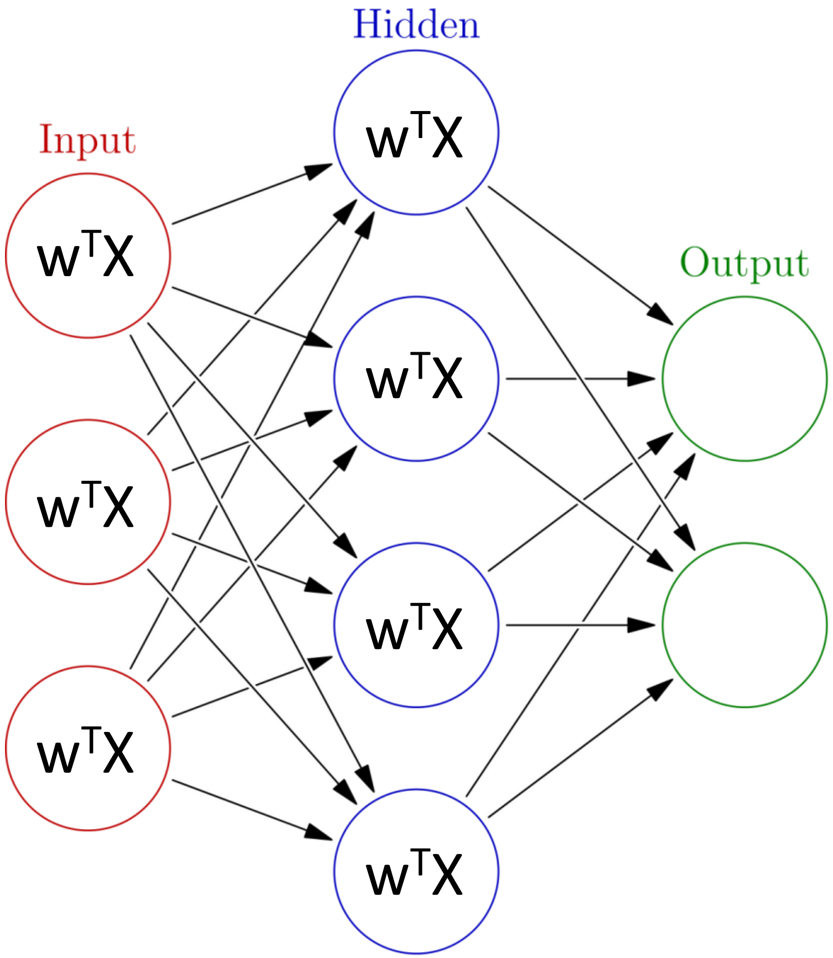
- The metric to measure the quality of the split
- The number of features to evaluate at each node
- The depth of the tree
- The minimum number of samples required to split the data further,
- More…

Random Forests and GBMs



- Number of trees (or estimators)
- Learning rate (GBMs)

Neural Network Hyperparameters



- Number of layers
- Number of neurons per layer
- The activation function
- The dropout rate
- More…

Other model Hyperparameters

- Nearest neighbours ➔ the number of neighbours
- Support vector machines ➔ the kernel function

Hyperparameters in ML models

- Hyperparameters could have a big impact on the performance of the learning algorithm.

- Optimal hyperparameter settings often differ for different datasets.

- Therefore they should be optimized for each dataset.

THANK YOU

www.trainindata.com