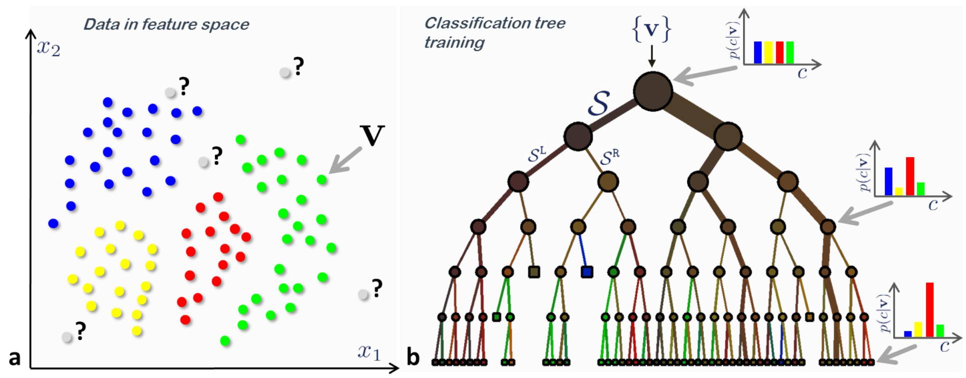

Fig. 3.1: **Classification: training data and tree training. (a)** Input data points. The ground-truth label of training points is denoted with different colours. Grey circles indicate unlabelled, previously unseen test data. **(b)** A binary classification tree. During training a set of labelled training points $\{\mathbf{v}\}$ is used to optimize the parameters of the tree. In a classification tree the entropy of the class distributions associated with different nodes decreases (the confidence increases) when going from the root towards the leaves.

Given a labelled training set learn a general mapping which associates previously unseen test data with their correct classes.

The need for a general rule that can be applied to "not-yet-available" test data is typical of *inductive* tasks.[1] In classification the desired output is of discrete, categorical, unordered type. Consequently, so is the nature of the training labels. In fig. 3.1a data points are denoted with circles, with different colours indicating different training labels. Testing points (not available during training) are indicated in grey.

More formally, during testing we are given an input test data $\mathbf{v}$ and we wish to infer a class label c such that $c \in \mathcal{C}$, with $\mathcal{C} = \{c_k\}$. More generally we wish to compute the whole distribution $p(c|\mathbf{v})$. As

[1] As opposed to *transductive* tasks. The distinction will become clearer later.