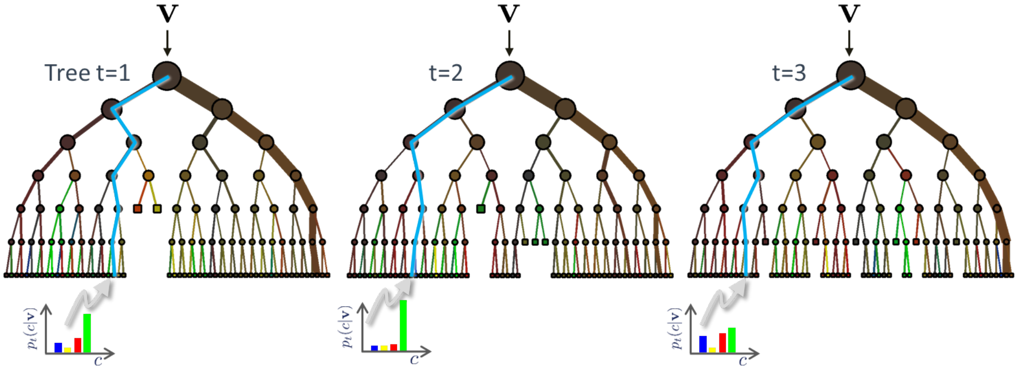

Fig. 3.2: **Classification forest testing.** During testing the same unlabelled test input data $\mathbf{v}$ is pushed through each component tree. At each internal node a test is applied and the data point sent to the appropriate child. The process is repeated until a leaf is reached. At the leaf the stored posterior $p_t(c|\mathbf{v})$ is read off. The forest class posterior $p(c|\mathbf{v})$ is simply the average of all tree posteriors.

sample $\rho = 1000$ parameter values out of possibly billions or even infinite possibilities. It is important to point out that it is not necessary to have the entire set $\mathcal{T}$ pre-computed and stored. We can generate each random subset $\mathcal{T}_j$ as needed before starting training the corresponding node.

The leaf and ensemble prediction models. Classification forests produce probabilistic output as they return not just a single class point prediction but an entire class distribution. In fact, during testing, each tree leaf yields the posterior $p_t(c|\mathbf{v})$ and the forest output is simply:

$$p(c|\mathbf{v}) = \frac{1}{T}\sum_{t}^{T} p_t(c|\mathbf{v}).$$

This is illustrated with a small, three-tree forest in fig. 3.2.

The choices made above in terms of the form of the objective function and that of the prediction model characterize a classification forest. In later chapter we will discuss how different choices lead to different