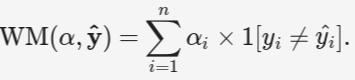
**2. Recall that the classification error for unweighted data is defined as follows:**



**Meanwhile, the weight of mistakes for weighted data is given by**



**If we set the weights α=1 for all data points, how is the weight of mistakes WM(α,ŷ)related to the classification error?**

**WM(α,ŷ) = [classification error]**

**WM(α,ŷ)** = [classification error] \* [weight of correctly classified data points]

**WM(α,ŷ)** = N \* [classification error]

**WM(α,ŷ)** = 1 - [classification error]

**WM(α,ŷ)** = [classification error]

**WM(α,ŷ)** = [classification error] \* [weight of correctly classified data points]

**WM(α,ŷ) = N \* [classification error]**

**WM(α,ŷ)** = 1 - [classification error]

**3. Refer to section Example: Training a weighted decision tree.**

**Will you get the same model as small\_data\_decision\_tree\_subset\_20 if you trained a decision tree with only 20 data points from the set of points in subset\_20?**

**Yes**

No

**4. Refer to the 10-component ensemble of tree stumps trained with Adaboost.**

**As each component is trained sequentially, are the component weights monotonically decreasing, monotonically increasing, or neither?**

Monotonically decreasing

Monotonically increasing

**Neither**

**5. Which of the following best describes a general trend in accuracy as we add more and more components? Answer based on the 30 components learned so far.**

Training error goes down monotonically, i.e. the training error reduces with each iteration but never increases.

**Training error goes down in general, with some ups and downs in the middle.**

Training error goes up in general, with some ups and downs in the middle.

Training error goes down in the beginning, achieves the best error, and then goes up sharply.

None of the above

**6. From this plot (with 30 trees), is there massive overfitting as the # of iterations increases?**

Yes

**No**