# STATISTICAL MACHINE LEARNING (WS2024/25)
# HOMEWORK: GENERALIZATION BOUND

**Assignment (2 points)** *Assume a predictor $h_m \colon \mathcal{X} \to \mathcal{Y}$, where $\mathcal{Y} = \{0, 1, \ldots, Y-1\}$, trained on i.i.d. sample $\mathcal{T}^m = ((x^i, y^i) \in \mathcal{X} \times \mathcal{Y} \mid i = 1, \ldots, m)$ by a learning algorithm $A \colon \cup_{m=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^m \to \mathcal{H}$, where $\mathcal{H} = \{h_1, \ldots, h_H\}$ is a finite hypothesis space. The trained predictor $h_m = A(\mathcal{T}^m)$ is evaluated on the same training sample $\mathcal{T}^m$.*

*Given:*

- *A sequence of class predictions $(h_m(x^1), \ldots, h_m(x^m))$ on the training sample $\mathcal{T}^m$,*
- *A sequence of true class labels $(y^1, \ldots, y^m)$,*
- *A loss function $\ell \colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$,*
- *An error level $\delta \in (0, 1)$,*
- *The number of hypotheses $H$,*

*your tasks is to compute an upper bound $R_{\mathrm{UB}}$ such that $R(h_m) \leq R_{\mathrm{UB}}$ holds with probability $1 - \delta$ at least.*

*You have to fill in code for the following Python function:*

```
def generalization_bound(true_y, pred_y, loss, delta, H ):
    # Input:
    #  true_y is np.array(l,dtype=int) representing the true class labels;
    #   a label is an integer from 0 to Y-1
    #  pred_y is np.array(l,dtype=int) representing the predicted class labels;
    #   a label is an integer from 0 to Y-1
    #  Loss is np.array((Y,Y)) whose Loss[y,yy] represents  the loss
    #   incurred when the true label is y and prediction is yy
    #  delta is a scalar from (0,1) representing the probability of failure
    #  H  is a positive integer representing the number of hypothesis.
    #
    # Output:
    #  R_UB the upper bound on the expected risk

    return R_UB
```