Aufgabe 1: Erzeugen Sie einfache Mustererkennungsprobleme, wenden Sie ein vorwärtsgerichtetes neuronales Netz darauf an und untersuchen Sie, inwieweit sich die Gewichtsvektoren der trainierten Neuronen interpretieren lassen.

Aufgabe 2: Wenden Sie Support Vector Mashines auf Mustererkennungsprobleme an. Untersuchen Sie, welche Kernels und welche Kernelparameter die besten die Vorhersagen liefern. Stellen Sie dabei auch Soft-Margin und Hard-Margin gegenüber. Untersuchen Sie, wieviele und möglichst auch welche Support-Vektoren jeweils verwendet werden.

Aufgabe 3: Zeigen Sie, dass man für gegebene $\gamma \in \mathbb{R}$, $y \in \mathbb{R}^m$ und $x_1, .., x_m \in \mathbb{R}^n$ die Berechnung von

$$\arg \min_{\beta \in \mathbb{R}^n, \beta_0 \in \mathbb{R}, \delta \in [0,1]^m} \left\{ \frac{1}{2} \|\beta\|^2 + \gamma \sum_{i=1}^{m} \delta_i^2 \;\middle|\; \forall i \in \{1, ..., m\} \;:\; y_i(\langle x_i, \beta \rangle + \beta_0) \geq 1 - \delta_i \text{ und } \delta_i \geq 0 \right\}$$

zurückführen kann auf die Maximierung von

$$g(\xi) = \sum_{j=1}^{m} \xi_j - \frac{1}{2} \sum_{j,k=1}^{m} \xi_j \xi_k y_j y_k \left(\langle x_j, x_k \rangle + \frac{\delta_{ij}}{\gamma} \right)$$

mit den Nebenbedingungen $\xi_j \geq 0$, $\sum_{j=1}^{m} \xi_j y_j = 0$, sowie $\delta_{ii} = 1$ und $\delta_{ij} = 0$ für $i \neq j$.