Algorithm 1 Preliminary pseudocode of the expected method described in this proposal

Inputs: Given a dataset $\mathcal{D}$, a weighted kernel κ_ω and a classifier $\mathcal{A}$

Let β represents a dependency network distribution initialized with an independent joint distribution: $\beta \leftarrow$ Independent joint distribution.

repeat

Split $\mathcal{D}$ in training $\mathcal{D}_\alpha$ and testing $\mathcal{D}_\theta$ data

$\bar{\Omega} \leftarrow$ Sample k candidates from β

for $\boldsymbol{\omega}_j \in \bar{\Omega}$ **do**

Train classifier: $h_j \leftarrow \mathcal{A}(\mathcal{D}_\alpha, \kappa_{\omega j})$

Test classifier: $s_j \leftarrow \text{error}(h_j, \mathcal{D}_\theta, \kappa_{\omega j})$

end for

$\bar{\Omega}' \leftarrow \text{bestCandidates}(\bar{\Omega}, s)$

Re-estimate dependency network: $\beta \leftarrow \text{reEstimate}(\bar{\Omega}')$

until Dependency network has converge or maximum iterations reached