

KRONECKER KERNEL RIDGE REGRESSION

- In MTP with target features, we often use kernel methods.
- Consider the following pairwise model representation in the primal:

$$f(\mathbf{x}, \mathbf{t}) = \omega^\top (\phi(\mathbf{x}) \otimes \psi(\mathbf{t})),$$

where ϕ is feature mapping for features and ψ is feature mapping for target (features) and \otimes is Kronecker product.

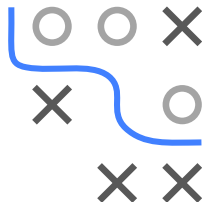
- This yields Kronecker product pairwise kernel in the dual:

$$f(\mathbf{x}, \mathbf{t}) = \sum_{(\mathbf{x}', \mathbf{t}') \in \mathcal{D}} \alpha_{(\mathbf{x}', \mathbf{t}')} \cdot k(\mathbf{x}, \mathbf{x}') \cdot g(\mathbf{t}, \mathbf{t}') = \sum_{(\mathbf{x}', \mathbf{t}') \in \mathcal{D}} \alpha_{(\mathbf{x}', \mathbf{t}')} \Gamma((\mathbf{x}, \mathbf{t}), (\mathbf{x}', \mathbf{t}')),$$

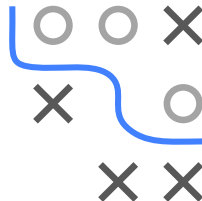
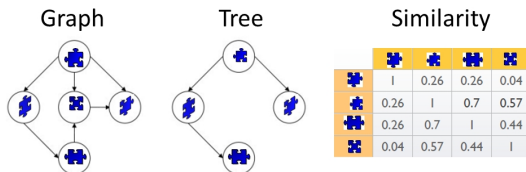
where k is kernel for feature map ϕ , g kernel for feature map ψ and $\alpha_{(\mathbf{x}', \mathbf{t}')}$ are dual parameters determined by:

$$\min_{\alpha} \|\Gamma \alpha - \mathbf{z}\|_2^2 + \lambda \alpha^\top \Gamma \alpha, \text{ where } \mathbf{z} = \text{vec}(Y)$$

- Commonly used in zero-shot learning.



EXPLOITING RELATIONS IN REGULARIZATION



- Graph-based regularization for graph-type relations in targets:

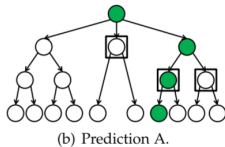
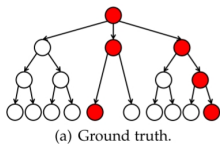
$$\min_{\Theta} \|Y - \Phi\Theta\|_F^2 + \lambda \sum_{m=1}^I \sum_{m' \in \mathcal{N}(m)} \|\theta_m - \theta_{m'}\|^2,$$

where $\mathcal{N}(j)$ is the set of targets related to target j .

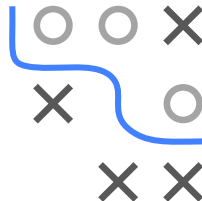
- The graph or tree is given as prior information.
- Can be extended to a weighted version aware of the similarities

Gopal and Yang, Recursive regularization for large-scale classification with hierarchical and graphical dependencies, KDD 2013.

HIERARCHICAL MULTI-LABEL CLASSIFICATION



	Tennis	Football	Biking	Movies	Tv	Belgium
01101 Text1	0	0	0	0	0	1
00111 Text2	0	0	1	0	1	1
01110 Text3	0	0	0	1	1	0
10001 Text4	0	0	1	0	1	0
01011 Text5	1	0	0	1	0	0
11110 Text6	?	?	?	?	?	?



- Hierarchies can also be used to define specific loss functions, such as the Hierarchy-loss:

$$L_{Hier}(\mathbf{y}, f) = \sum_{m: y_m \neq \hat{y}_m} c_m \mathbb{1}_{[anc(y_m) = anc(\hat{y}_m)]},$$

- This is rather common in multi-label classification problems.

Bi and Kwok, Bayes-optimal hierarchical multi-label classification, IEEE Transactions on Knowledge and Data Engineering, 2014.

PROBABILISTIC CLASSIFIER CHAINS

- Estimate the joint conditional distribution $\mathbb{P}(\mathbf{y} \mid \mathbf{x})$.
- For optimizing the subset 0/1 loss:

$$L_{0/1}(\mathbf{y}, \hat{\mathbf{y}}) = \mathbb{1}_{[\mathbf{y} \neq \hat{\mathbf{y}}]}$$

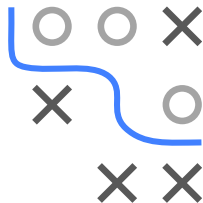
- Repeatedly apply the *product rule* of probability:

$$\mathbb{P}(\mathbf{y} \mid \mathbf{x}) = \prod_{j=m}^l \mathbb{P}(y_m \mid \mathbf{x}, y_1, \dots, y_{m-1}).$$

- Learning relies on constructing probabilistic classifiers for

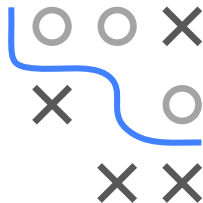
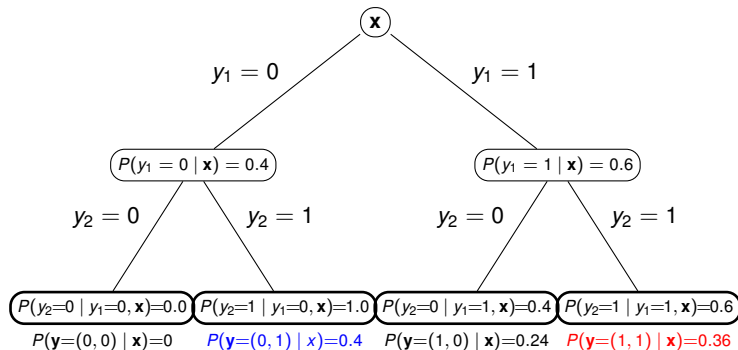
$$\mathbb{P}(y_m \mid \mathbf{x}, y_1, \dots, y_{m-1}),$$

independently for each $m = 1, \dots, l$.



PROBABILISTIC CLASSIFIER CHAINS

- Inference relies on exploiting a probability tree:

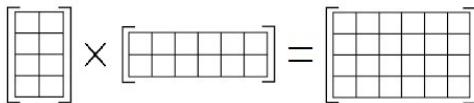


- For subset 0/1 loss one needs to find $h(\mathbf{x}) = \arg \max_{\mathbf{y}} \mathbb{P}(\mathbf{y} | \mathbf{x})$.
- Greedy and approximate search techniques with guarantees exist.
- Other losses: compute the prediction on a sample from $\mathbb{P}(\mathbf{y} | \mathbf{x})$.

LOW-RANK APPROXIMATION

- Θ : parameter matrix of dimensionality $p \times l$
- p : the number of (projected) features
- l : the number of targets
- Assume a low-rank structure of A :

$$U \times V = A$$



- We can write $\Theta = UV$ and $\Theta \mathbf{x} = UV\mathbf{x}$
- V is a $p \times \hat{l}$ matrix
- U is an $\hat{l} \times l$ matrix
- \hat{l} is the rank of Θ

