university of groningen

Dynamics of on–line learning in two–layer neural networks in the presence of concept drift

Michiel Straat

Fthi Abadi

Zhuoyun Kan

Christina Göpfert

Barbara Hammer

Michael Biehl

Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence

University of Groningen / NL

arXiv:2005.10531: Supervised Learning in the Presence of Concept Drift: A modelling framework

Table of contents

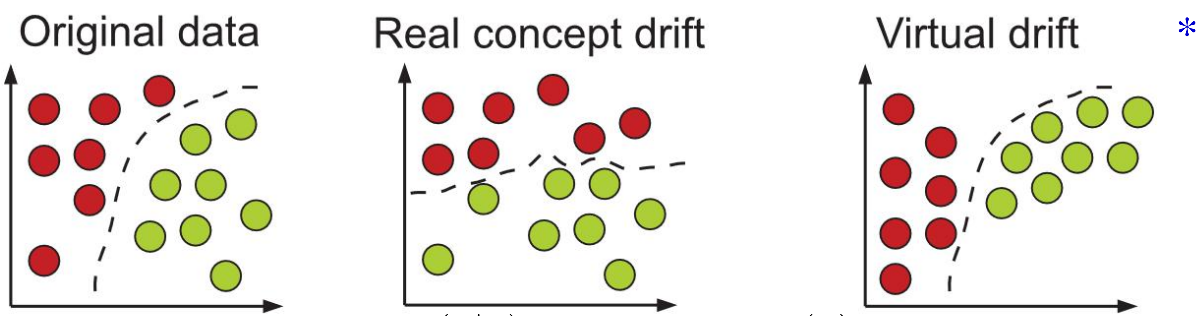
- Concept drift
- Model-scenario: student-teacher setup
- Including concept drift and weight decay
- Results for the ReLU- and Erf SCM, similarities and differences

Learning under concept drift

Traditional assumption in ML of stationarity is often violated

- *Virtual drift:* Change in input density $p(\boldsymbol{\xi})$
- *Real drift:* Change of the target rule $y = f(\boldsymbol{\xi})$

$$\exists \boldsymbol{\xi} : p_{t_0}(\boldsymbol{\xi}, y) \neq p_{t_1}(\boldsymbol{\xi}, y)$$

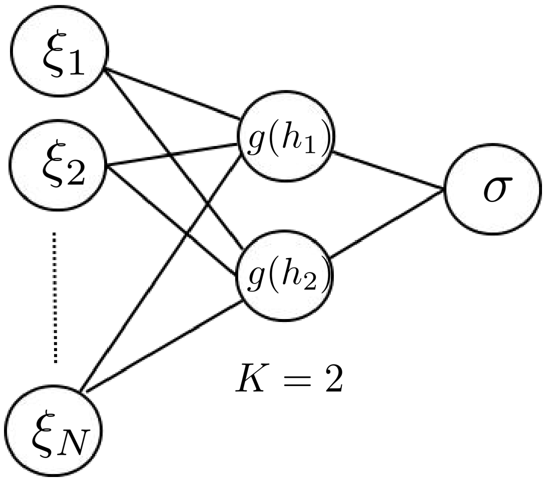


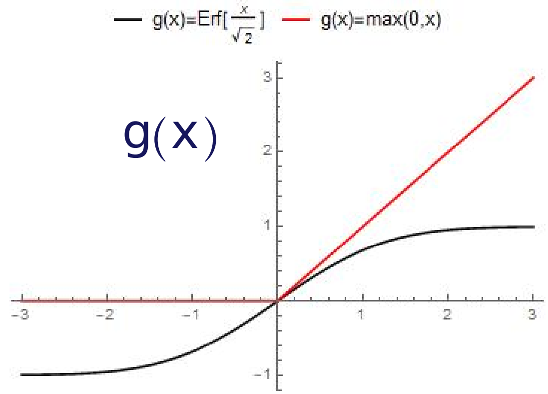
$p(y|\boldsymbol{\xi})$ changes $p(\boldsymbol{\xi})$ changes, but not $p(y|\boldsymbol{\xi})$

*J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, A. Bouchachia. 2014. A survey on concept drift adaptation. *ACM Comput. Surv.* 46, 4, Article 44 (April 2014)

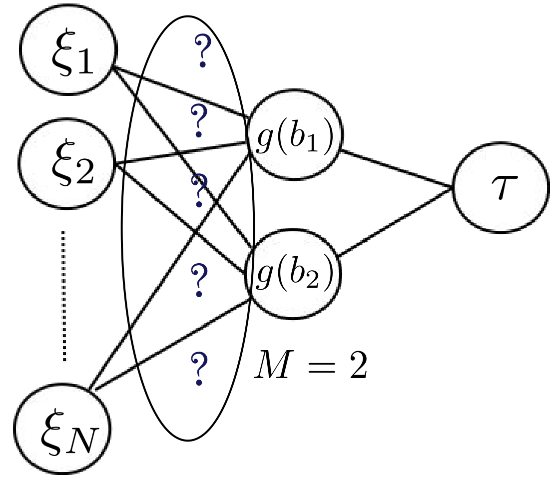
Model scenario: On–line learning of a drifting rule

Student hypothesis





Teacher rule



$\sigma(\boldsymbol{\xi}) = \sum_{i=1}^{K} g(h_i)$, with $h_i = \mathbf{w}_i \cdot \boldsymbol{\xi}$

$\tau(\boldsymbol{\xi}) = \sum_{m=1}^{M} g(b_m)$, with $b_m = \mathbf{B}_m \cdot \boldsymbol{\xi}$

- Order parameters: $Q_{ik} = \mathbf{w}_i \cdot \mathbf{w}_k, \quad R_{im} = \mathbf{w}_i \cdot \mathbf{B}_m, \quad T_{nm} = \mathbf{B}_n \cdot \mathbf{B}_m = \delta_{nm}$
- The student learns from random i.i.d. examples $(\boldsymbol{\xi}^\mu \in \mathbb{R}^N, \tau(\boldsymbol{\xi}^\mu) \in \mathbb{R})$

  $\langle \xi_i \rangle = 0, \quad \langle \xi_i \xi_j \rangle = \delta_{ij}$

- CLT for large N: (h_i, b_m) are zero-mean correlated Gaussian variables with $\langle h_i h_k \rangle = Q_{ik}, \langle b_n b_m \rangle = T_{nm}, \langle h_i b_m \rangle = R_{im}$

Learning dynamics of the networks (stationary)

Order parameters: $Q_{ik} = \mathbf{w}_i \cdot \mathbf{w}_k, \quad R_{im} = \mathbf{w}_i \cdot \mathbf{B}_m, \quad T_{nm} = \mathbf{B}_n \cdot \mathbf{B}_m = \delta_{nm}$

A stream of random i.i.d. examples $\boldsymbol{\xi}^1, \boldsymbol{\xi}^2, \boldsymbol{\xi}^3, \ldots$ (discrete time $\mu = 1, 2, 3, \ldots$)

At the presentation of one example $\boldsymbol{\xi}^\mu$

1. Quadratic error: $\epsilon^\mu = \frac{1}{2}(\sigma^\mu - \tau^\mu)^2$
2. Update student weights with gradient descent:

$$\mathbf{w}_i^{\mu+1} = \mathbf{w}_i^\mu + \frac{\eta}{N}\rho_i^\mu \boldsymbol{\xi}^\mu, \quad \text{with } \rho_i^\mu = (\tau^\mu - \sigma^\mu) g'(x_i^\mu)$$

3. Recursions of order parameters and example average

Closed form available for ReLU and Erf
Only available for Erf

$$R_{im}^{\mu+1} = R_{im}^\mu + \frac{\eta}{N}\langle \rho_i^\mu b_m^\mu \rangle$$

$$Q_{ik}^{\mu+1} = Q_{ik}^\mu + \frac{\eta}{N}\langle h_i^\mu \rho_k^\mu \rangle + \frac{\eta}{N}\langle h_k^\mu \rho_i^\mu \rangle + \frac{\eta^2}{N}\langle \rho_i^\mu \rho_k^\mu \rangle$$

(Saad & Solla, 95)

4. Consider the limit $N \to \infty$ and $\eta \to 0$ and scaled time:

$\widetilde{\alpha} = \eta\mu/N \qquad d\widetilde{\alpha} = \eta/N$ (continuous in the limits)

$$\left[\frac{dR_{im}}{d\widetilde{\alpha}}\right]_{stat} = \langle \rho_i b_m \rangle$$

$$\left[\frac{dQ_{ik}}{d\widetilde{\alpha}}\right]_{stat} = \langle h_i \rho_k \rangle + \langle h_k \rho_i \rangle$$

Introducing a random real drift

- Random change of the teacher vectors, while keeping orthonormality

$$\mathbf{B}_m^{\mu+1} \cdot \mathbf{B}_m^{\mu} = 1 - \widetilde{\delta}/N$$
$$T_{nm}^{\mu+1} = \mathbf{B}_n^{\mu+1} \cdot \mathbf{B}_m^{\mu+1} = \delta_{nm}$$

- Weight decay as a mechanism of *forgetting* older examples

$$\mathbf{w}_i = (1 - \widetilde{\gamma}/N)\, \mathbf{w}_i$$

$$\left[\frac{dR_{im}}{d\widetilde{\alpha}}\right]_{drift} = \left[\frac{dR_{im}}{d\widetilde{\alpha}}\right]_{stat} - (\widetilde{\delta} + \widetilde{\gamma}) R_{im}$$
$$\left[\frac{dQ_{ik}}{d\widetilde{\alpha}}\right]_{drift} = \left[\frac{dQ_{ik}}{d\widetilde{\alpha}}\right]_{stat} - 2\widetilde{\gamma}\, Q_{ik}$$

Generalization error:

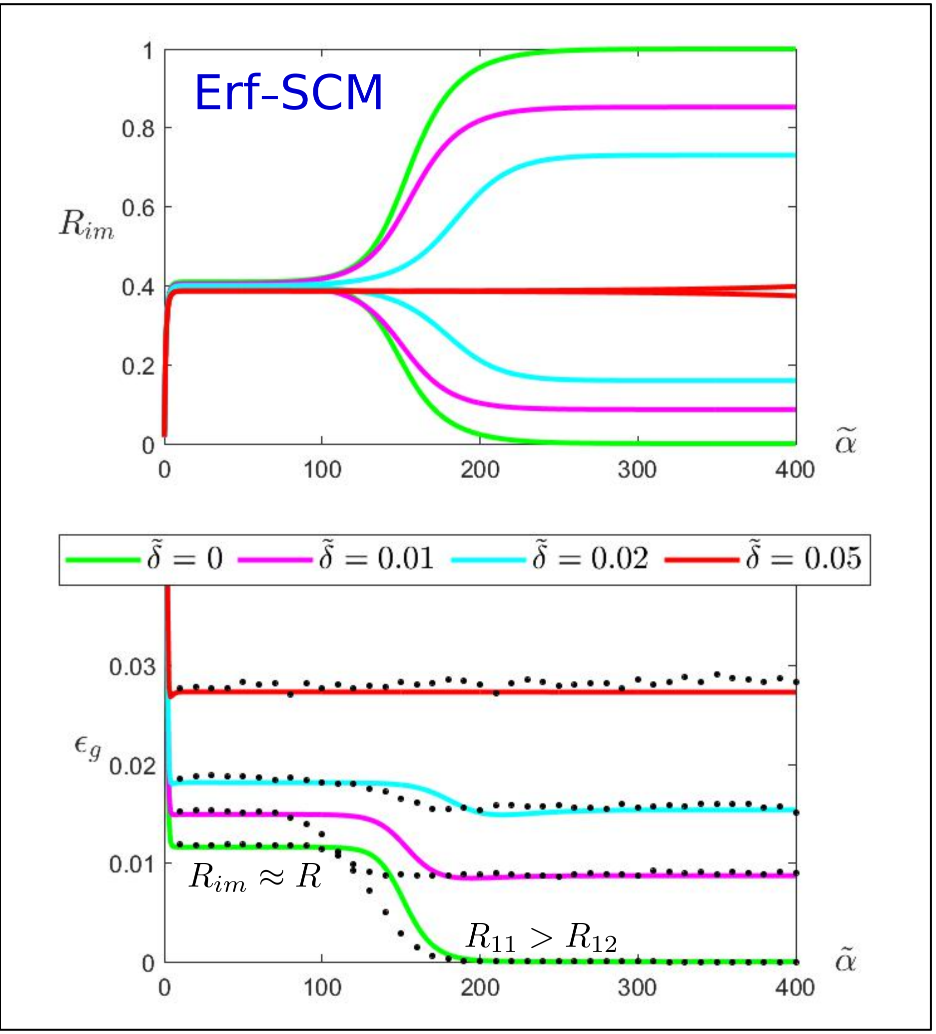
$$\epsilon_g = \frac{1}{2}\langle(\sigma^\mu - \tau^\mu)^2\rangle$$

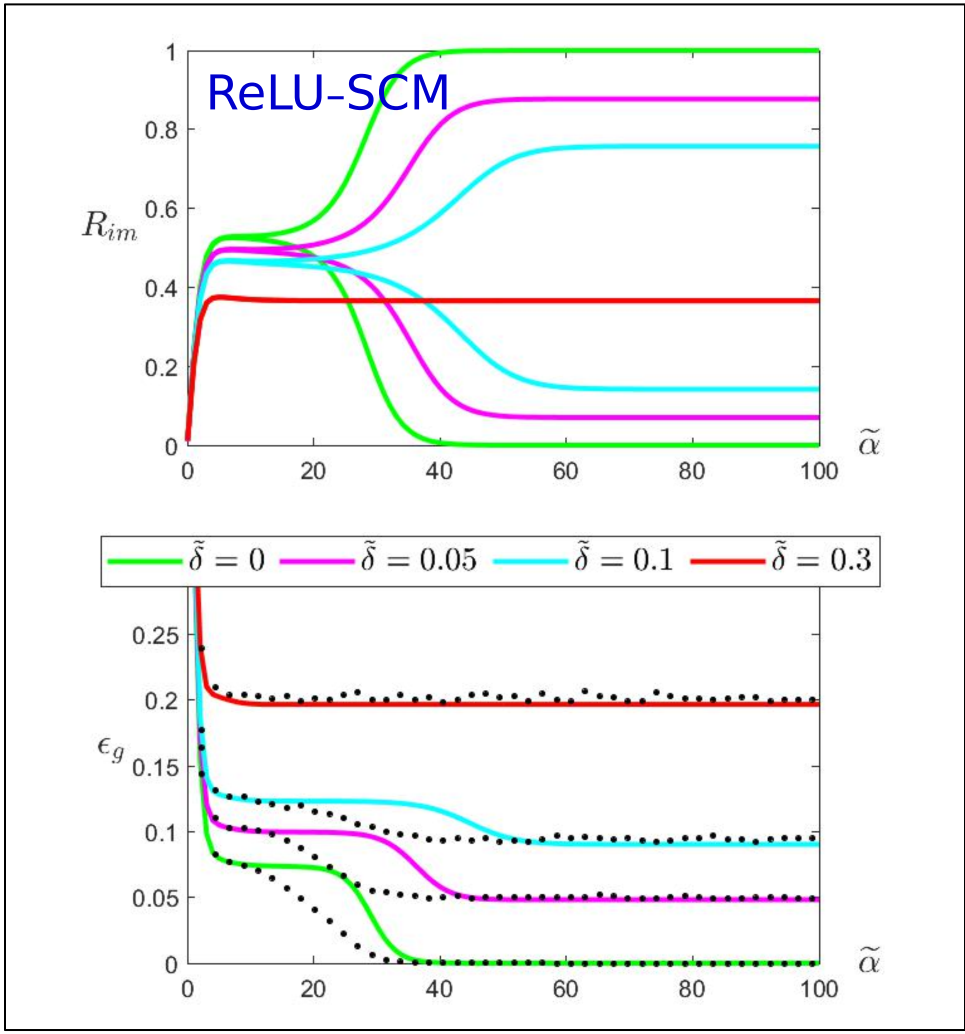
Closed form expression $\epsilon_g(R_{im}, Q_{ik})$ available for ReLU and Erf

Results matching student and teacher (M=K=2)

Initial conditions corresponding to no prior information about the rule:

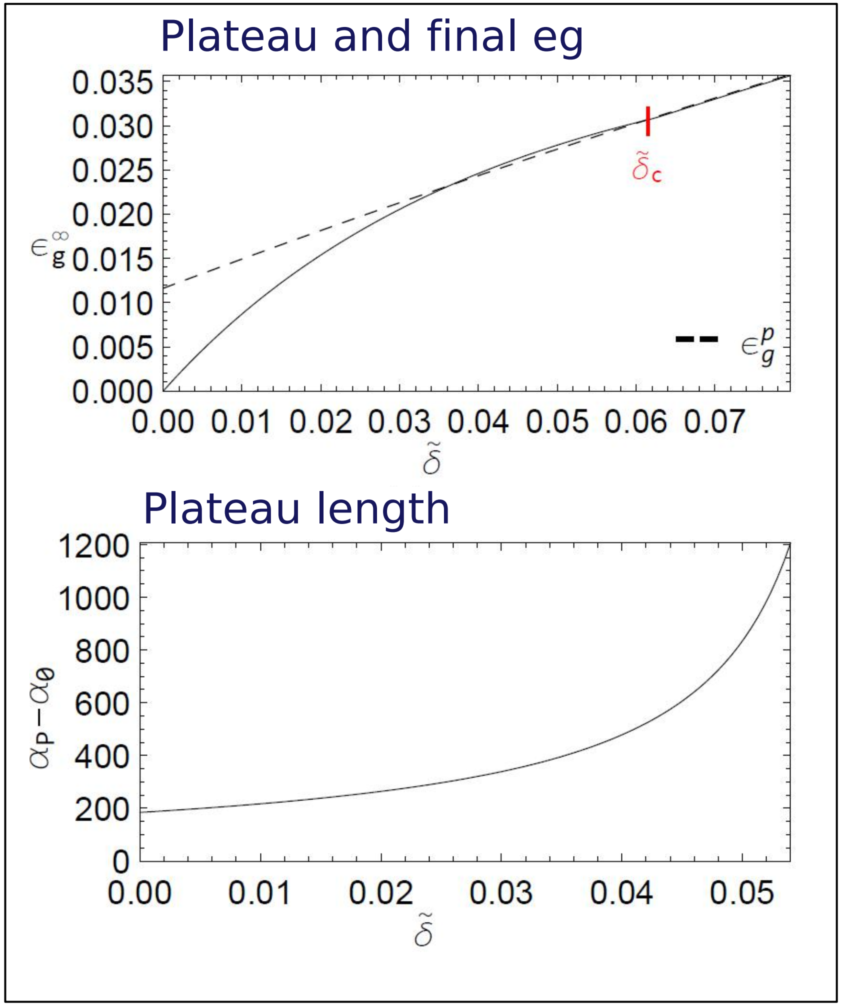
$$R_{im} \approx 0 \text{ and choose } Q_{11} = Q_{22} = 0.5,\ Q_{12} = 0.49$$



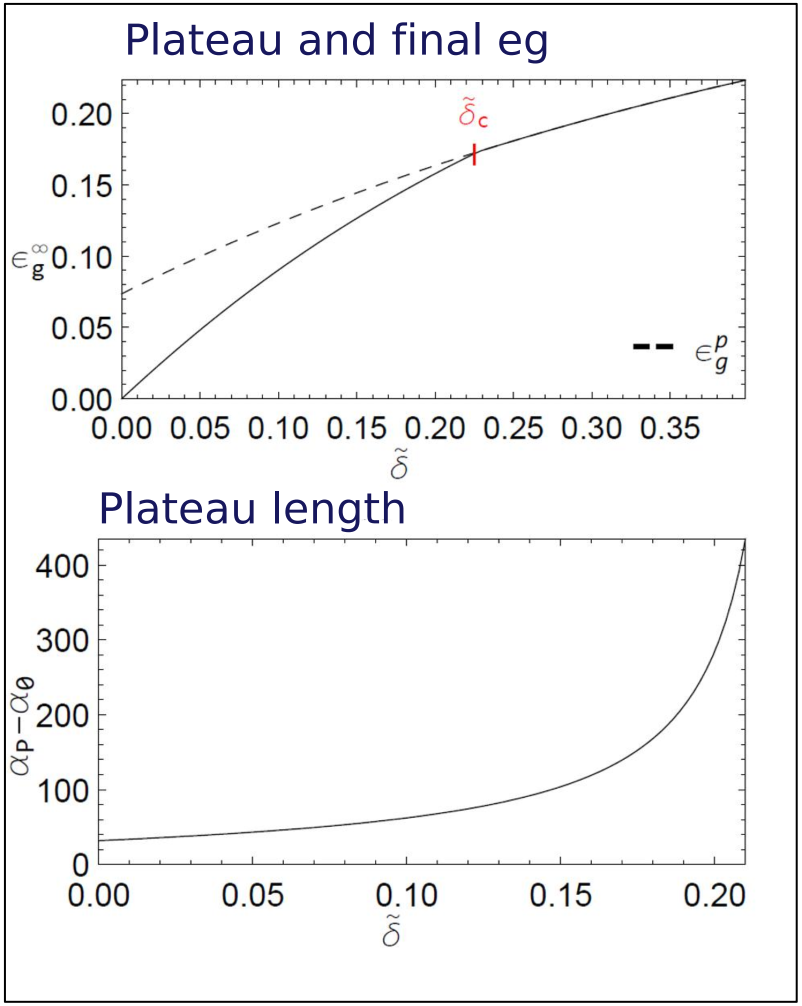


Dots: simulations for $N = 500, \eta = 0.05$ (avg. of 10 runs)

Erf–SCM

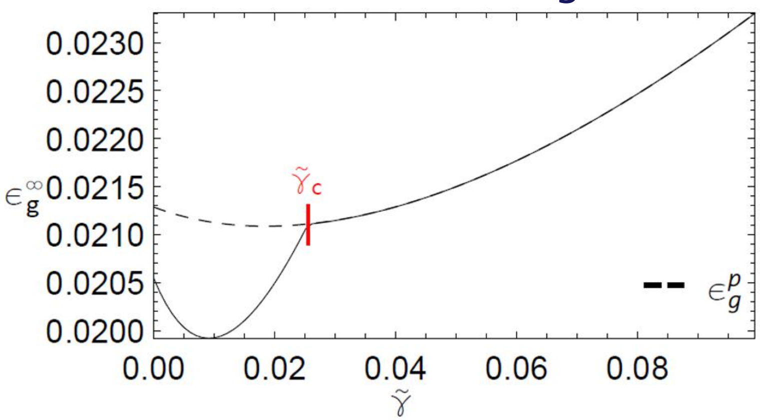


ReLU–SCM

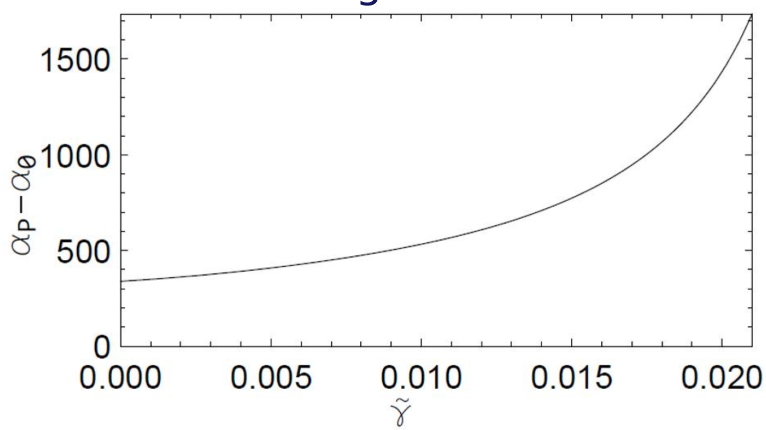


Erf-SCM $\tilde{\delta} = 0.03$

Plateau and final eg

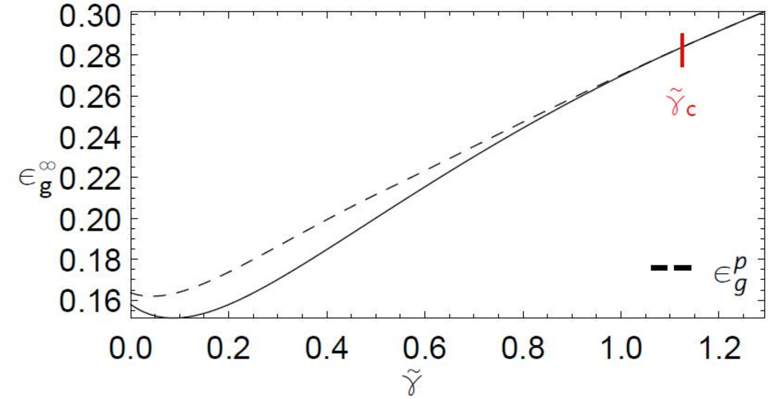


Plateau length

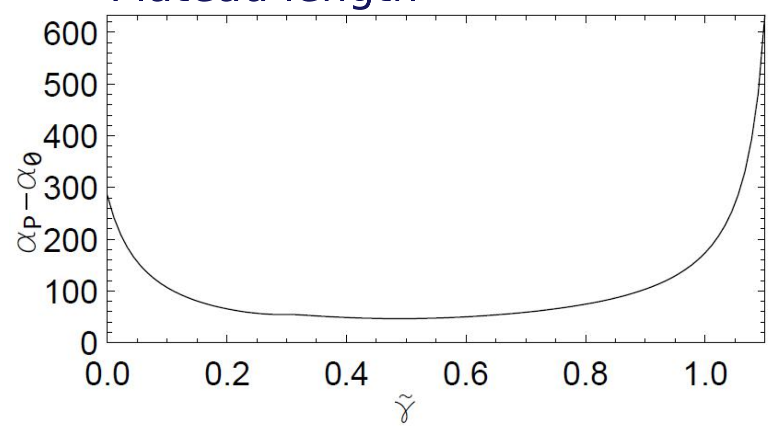


ReLU-SCM $\tilde{\delta} = 0.2$

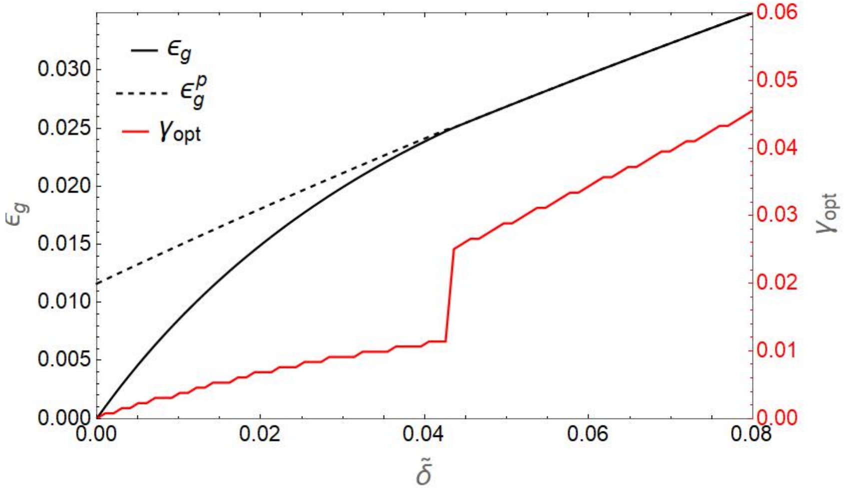
Plateau and final eg

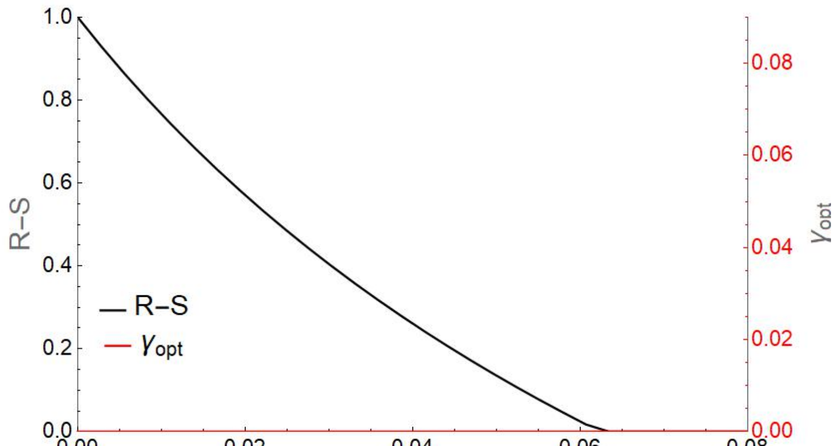


Plateau length



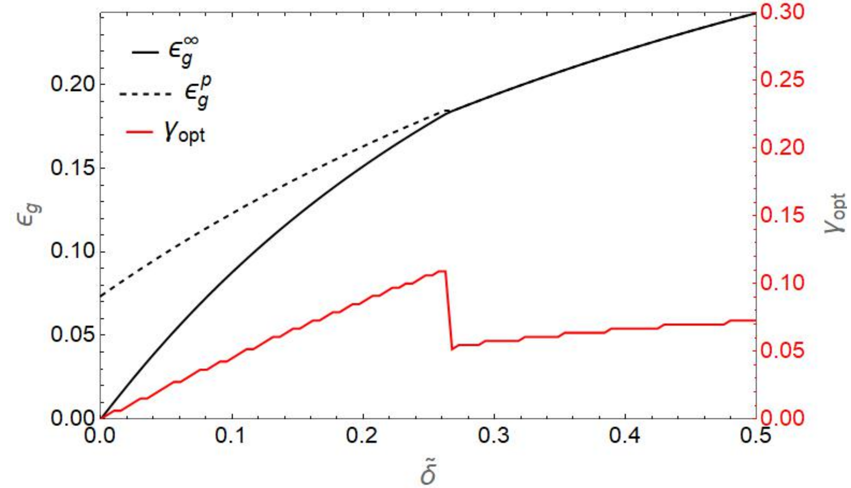
Erf-SCM

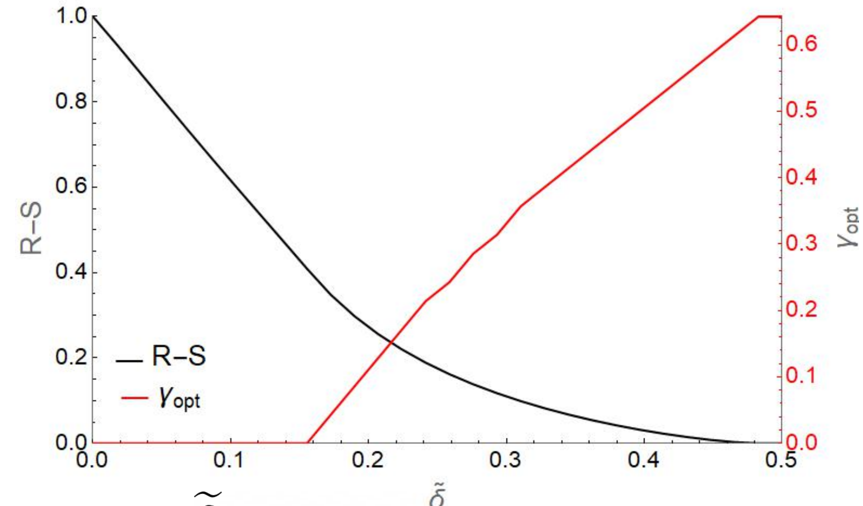




$$\widetilde{\delta}_c = 0.0615$$

ReLU-SCM





$$\widetilde{\delta}_c = 0.225$$

In the ReLU SCM, weight decay also optimizes the specialization.

Main findings

Common to both SCM...

- In the presence of concept drift, specialization possible uptill $\widetilde{\delta}_c$
- Drift increases the length of the plateau
- Weight decay could improve the final generalization error.

Differences between the SCM...

- Weight decay increased specialization for the ReLU SCM, while it always deteriorated specialization in the Erf SCM.
- Weight decay reduces the plateau length for the ReLU SCM, while it increases the plateau length in the Erf SCM.

Outlook

- Other types of real drift, e.g. a changing complexity of the rule by (de)-aligning teacher vectors
- Virtual drift by changing the density of the input data
- Increasing number of hidden units, mismatched student and teacher.
- Universal approximators: Adaptive thresholds and hidden to output weights
- Deep networks, tree-like architectures