Hinge Loss, Support Vector Machines and the Loss of Users

Andreas Maier, Vincent Christlein, Tobias Würfl, Katharina Breininger, Shahab Zarei

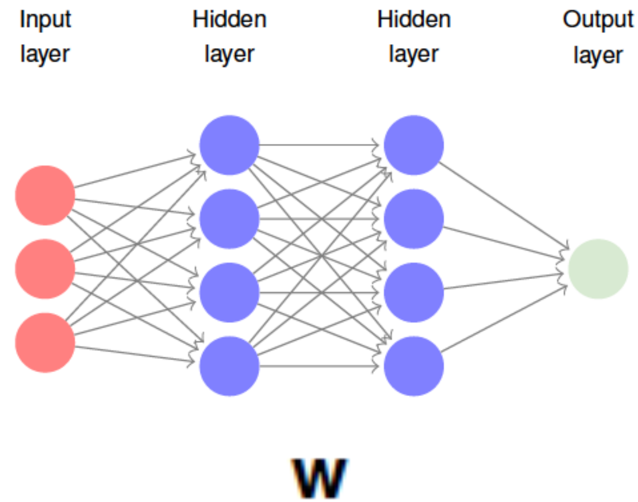
Lehrstuhl für Mustererkennung (Informatik 5),
Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany

Hinge Loss, Support Vector Machines, and the Loss of Users

- Hinge Loss
- Support Vector Machines
- Loss of Users
- Outlook

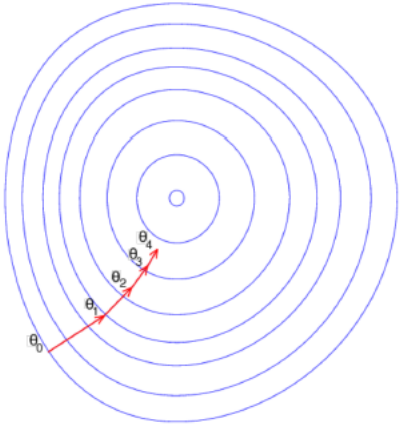
Loss Function

- **Tells a network the merit of its output**

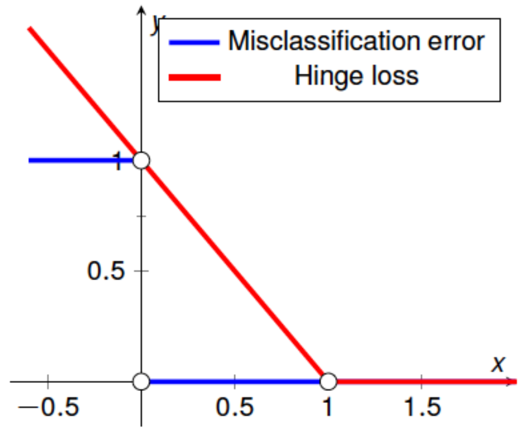


$$\underset{\mathbf{w}}{\text{minimize}} \quad \{L(\mathbf{w}, \mathbf{x}, \mathbf{y})\}$$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k} - \eta \nabla_{\mathbf{w}} \frac{1}{M} \sum_{m=1}^{M} L(\mathbf{w}, \mathbf{x}, \mathbf{y})$$



Hinge Loss



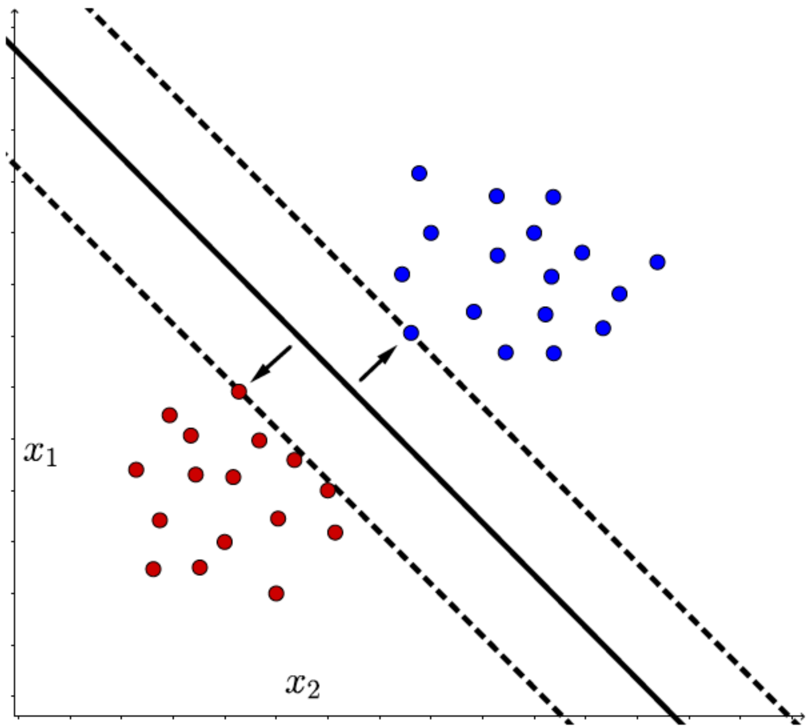
$$L(\mathbf{w}) = \sum_{m=1}^{M} \max\left(0, 1 - y_m \hat{y}_m(\mathbf{x}_m, \mathbf{w})\right)$$

- **Accepts correct output equally**
- **Punishes misclassification in a linear fashion**

Hinge Loss, Support Vector Machines, and the Loss of Users

- Hinge Loss
- Support Vector Machines
- Loss of Users
- Outlook

Support Vector Machines



$$\hat{y}(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$$

$$\arg\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2$$
$$\text{s.t. } \forall n : y_n(\boldsymbol{w}^\top \boldsymbol{x}_n + b) \geq 1$$

Support Vector Machines – Soft Margin

$$\min_{\boldsymbol{w},b,\xi_n} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{n=1}^{N}\xi_n$$
$$\text{s. t. } \forall n : y_n(\boldsymbol{w}^\top \boldsymbol{x}_n + b) \geq 1 - \xi_n$$
$$\forall n : \xi_n \geq 0 \,,$$

- **Choose slack variables as:**

$$\xi_n = \max\left(0, 1 - y_n(\boldsymbol{w}^\top \boldsymbol{x}_n + b)\right)$$

- **Simplified optimization problem:**

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{n=0}^{N} \max\left(0, 1 - y_n(\boldsymbol{w}^\top \boldsymbol{x}_n + b)\right)$$

- **Equivalent to hard margin with**

$$\alpha_n = \frac{C}{N}$$

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|w\|^2 + \sum_{n=1}^{N}\alpha_n \max(1 - y_n(\boldsymbol{w}^\top \boldsymbol{x}_n + b), 0)$$
$$\text{s. t. } \forall n : \alpha_n \geq 0 \,.$$

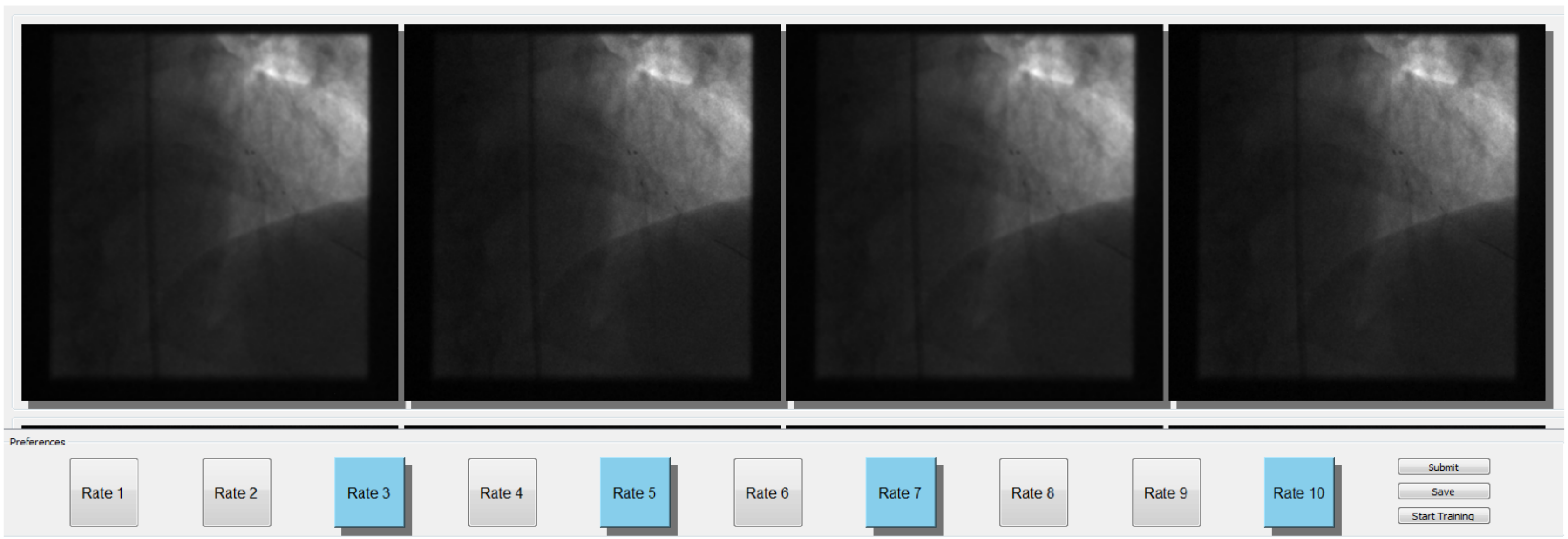
Support Vector Machines – Observations

- Soft margin support vector machines can be formulated with hinge loss
- Duality and notion of support vectors are lost
- Integration into deep networks is easy
- Read details in Vincent Christlein‘s PhD Thesis [1]

Hinge Loss, Support Vector Machines, and the Loss of Users

- Hinge Loss
- Support Vector Machines
- Loss of Users
- Outlook

Loss of Users [2]



$$\operatorname{argmin}_{\varphi} ||\boldsymbol{x}_{\mathrm{ref}} - \boldsymbol{x}_{\mathrm{NN}}||_2^2$$

Loss of Users [2]

- **Experts can pick one choice:**

$$e = ||\boldsymbol{x}_{\text{pref}} - \boldsymbol{x}||_2^2$$

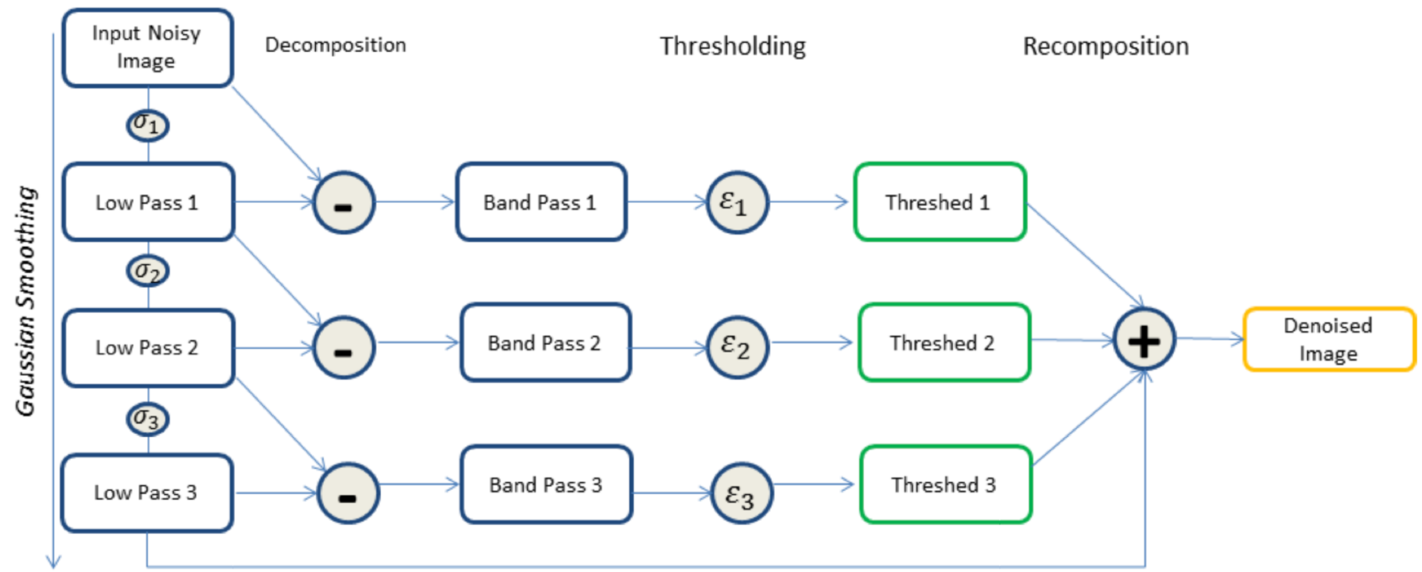
- **The other choices in Frame s are sub-optimal**

$$e_{s,*} \leq e_{s,q} \qquad \forall q \in \{0, \ldots, Q-1\}$$

$$* = 2 \Longrightarrow \begin{array}{l} e_{s,2} < e_{s,0} \\ e_{s,2} < e_{s,1} \\ e_{s,2} < e_{s,3} \end{array}$$

Loss of Users [2]

- **Will only work with few parameters**
- **Here Precision Learning [3] approach to Laplacian Pyramid Filter**



Loss of Users [2]

- **Net needs to mimick the ideal image:**

$$e = ||\boldsymbol{x}_{\mathrm{NN}} - \boldsymbol{x}||_2^2$$

- **Three constraints per click:**

$$* = 2$$

$$\Downarrow$$

$$\begin{array}{lclcl}
e_{s,2} < e_{s,0} & \rightarrow & e_{s,2} - e_{s,0} < 0 & \rightarrow & \max(e_{s,2} - e_{s,0}, 0) \\
e_{s,2} < e_{s,1} & \rightarrow & e_{s,2} - e_{s,1} < 0 & \rightarrow & \max(e_{s,2} - e_{s,1}, 0) \\
e_{s,2} < e_{s,3} & \rightarrow & e_{s,2} - e_{s,3} < 0 & \rightarrow & \max(e_{s,2} - e_{s,3}, 0)
\end{array}$$

Loss of Users

- **Constraint only loss function:**

$$\underset{\sigma_0,\sigma_1,\sigma_2,\tau_0,\tau_1,\tau_2}{\operatorname{argmin}} \sum_{s=0}^{S-1} \sum_{q=0}^{Q-1} \max(e_{s,*} - e_{s,q}, 0)$$

- **Best-case only loss function:**

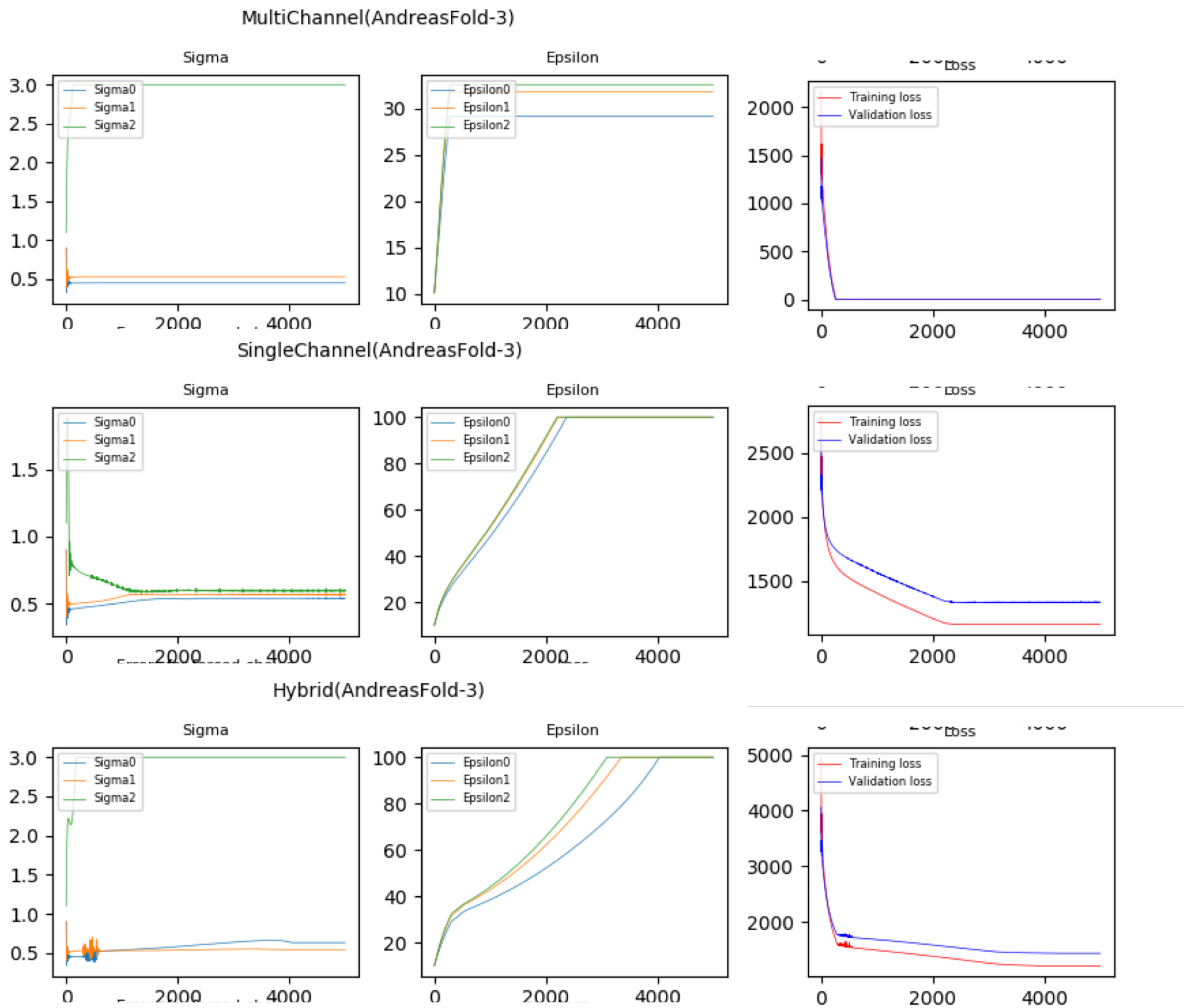
$$\underset{\sigma_0,\sigma_1,\sigma_2,\tau_0,\tau_1,\tau_2}{\operatorname{argmin}} \sum_{s=0}^{S-1} e_{s,*}$$

- **Hybrid loss:**

$$\underset{\sigma_0,\sigma_1,\sigma_2,\tau_0,\tau_1,\tau_2}{\operatorname{argmin}} \sum_{s=0}^{S-1} e_{s,*} + \sum_{q=0}^{Q-1} \max(e_{s,*} - e_{s,q}, 0)$$

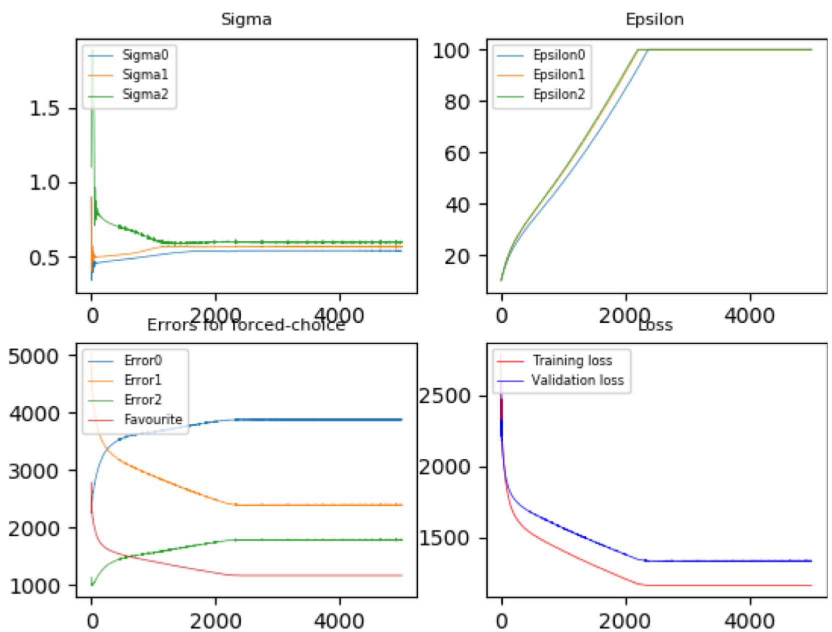
Any similarities with SVM?

Loss of Users – Results – User1

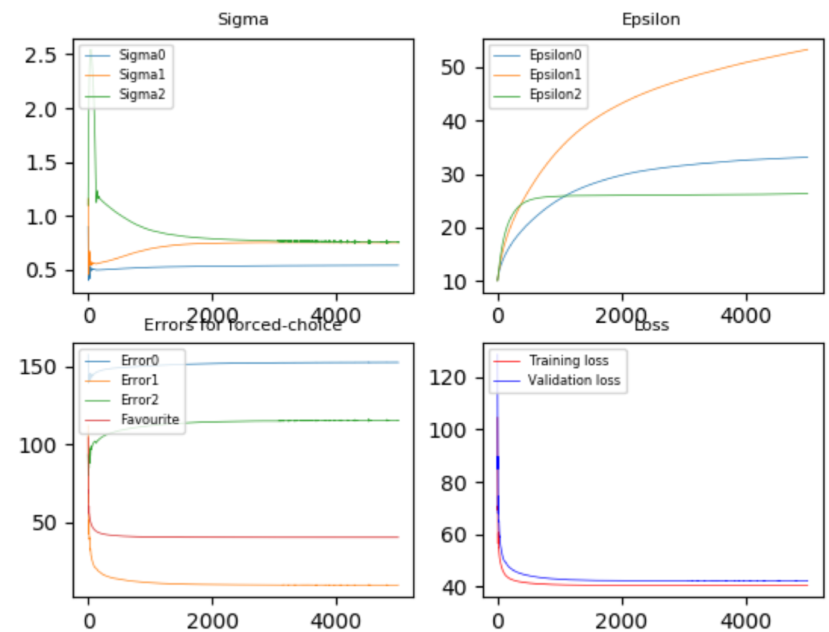


Loss of Users – Results – User1 vs User2

User1



User2



Summary

- Hinge Loss
- Support Vector Machines
- Loss of Users
- More Results in [2]

Thank you for your attention!

[1] Vincent Christlein. Hand-written Document Analysis with Focus on Writer Identification and Writer Retrieval. PhD Thesis. Friedrich-Alexander-University Erlangen-Nuremberg, 2018.

[2] Shahab Zarei, Bernhard Stimpel, Christopher Syben, Andreas Maier. User Loss - A Forced-Choice-Inspired Approach to Train Neural Networks directly by User Interaction. Under Review. https://arxiv.org/abs/1807.09303

[3] Andreas Maier, Frank Schebesch, Christopher Syben, Tobias Würfl, Stefan Steidl, Jang-Hwan Choi, Rebecca Fahrig. Precision Learning: Towards Use of Known Operators in Neural Networks. International Conference on Pattern Recognition ICPR 2018 (to appear). https://arxiv.org/abs/1712.00374