Robust Artificial Intelligence

Reading Seminar; Tsinghua University

Thomas G. Dietterich, Oregon State University

tgd@cs.orst.edu

Lecture 2: Rejection

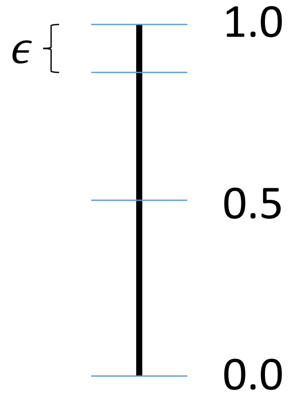
- Given:
  - Training data $(x_1, y_1), \dots, (x_N, y_N)$
  - Target accuracy level $1 - \epsilon$
  - Learn a classifier f and a rejection rule r
- At run time
  - Given query x_q
  - If $r(x_q) < 0$, REJECT
  - Else classify $f(x_q)$

Papers for Today

- Cortes, C., DeSalvo, G., & Mohri, M. (2016). Learning with rejection. *Lecture Notes in Artificial Intelligence*, *9925 LNAI*, 67–82. http://doi.org/10.1007/978-3-319-46379-7_5
- Shafer, G., & Vovk, V. (2008). A tutorial on conformal prediction. Journal of Machine Learning Research, 9, 371–421. Retrieved from http://arxiv.org/abs/0706.3188
- Papadopoulos, H. (2008). Inductive Conformal Prediction: Theory and Application to Neural Networks. Book chapter. https://www.researchgate.net/publication/221787122_Inductive_Conformal_Prediction_Theory_and_Application_to_Neural_Networks

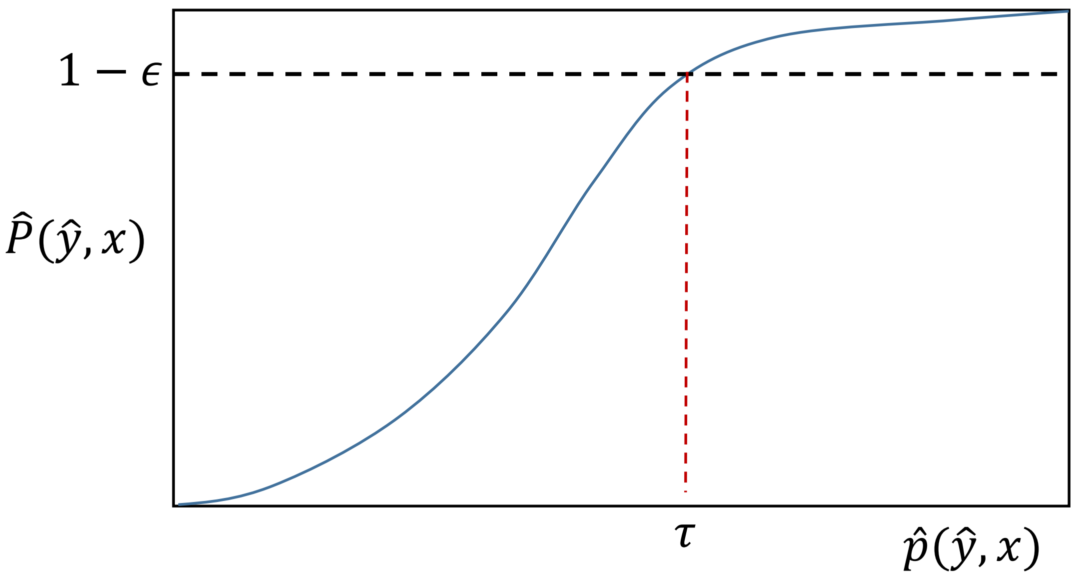
Basic Theory

- Suppose $f^*(x, y) = P(y|x)$ is the optimal probabilistic classifier
- Best prediction is $\hat{y} = \arg\max_y f^*(x, y)$
- Then the optimal rejection rule is to REJECT if $f^*(x, \hat{y}) < 1 - \epsilon$
- (Chow 1970)



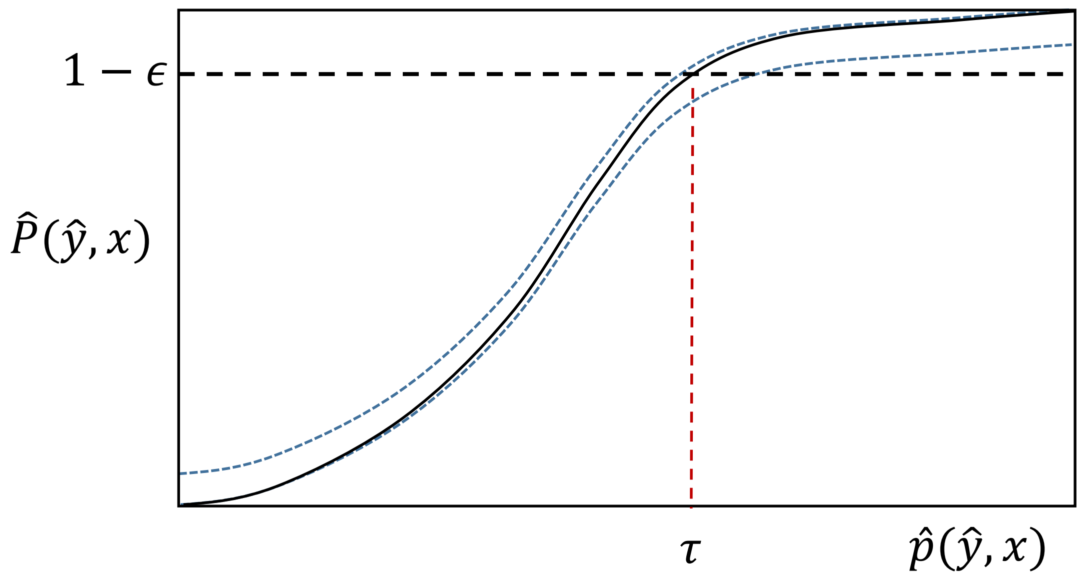
Non-Optimal Case

- If f is not optimal, we can still determine a threshold with performance guarantees
- Let $(f(x_i, \hat{y}_i), I[\hat{y}_i = y_i])$ be a set of calibration data points $i = 1, \dots, N$
- Sort them by $\hat{p}(\hat{y}_i | x_i) = f(x_i, \hat{y}_i)$
- Choose the smallest threshold τ such that if $f(x_i, \hat{y}_i) > \tau$ then the fraction of correct predictions is $1 - \epsilon$



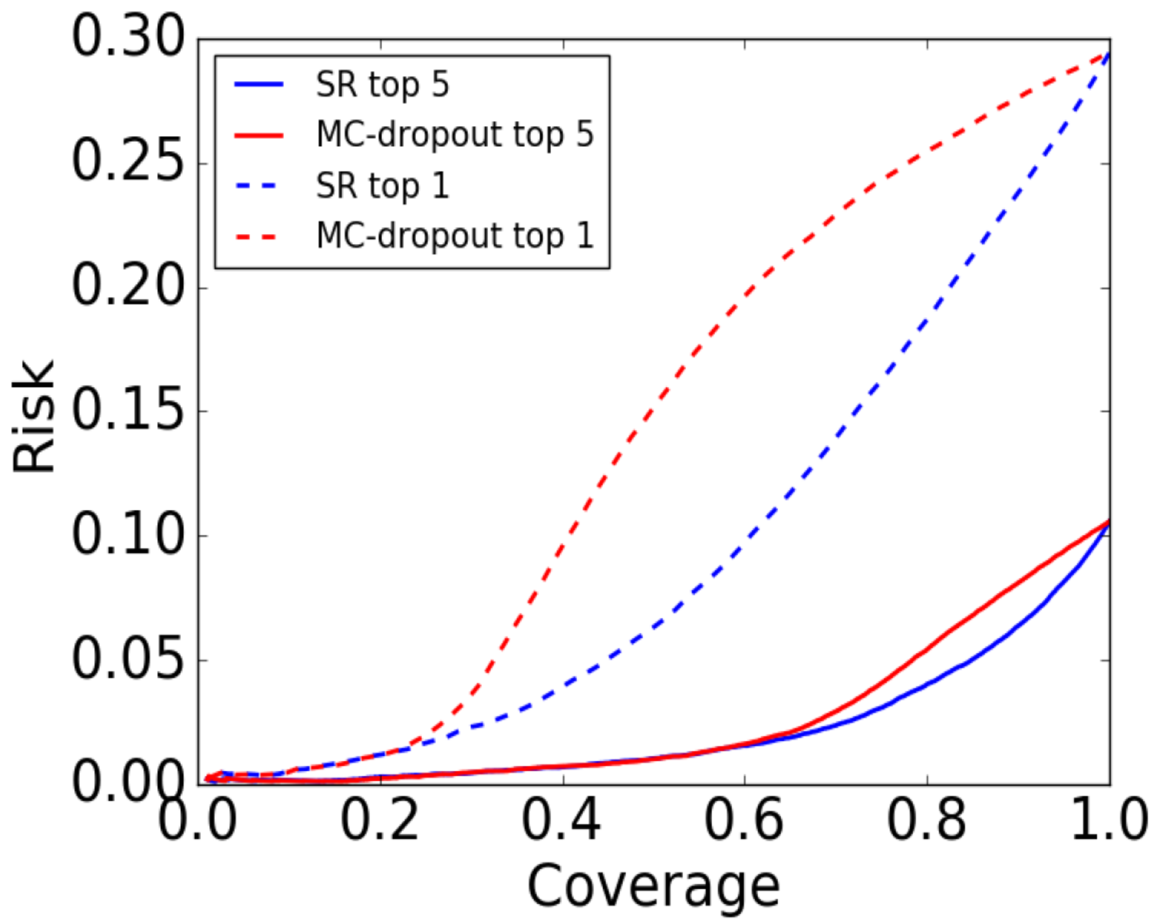
Finite Sample (PAC) Guarantee

- $P\left(\sqrt{n}\sup_x\left|\hat{F}_n(x) - F(x)\right| > \lambda\right) \leq 2\exp\left(-2\lambda^2\right)$ Massart (1990)
- Set $x \coloneqq \tau$
- $P\left(\eta > \frac{\lambda}{\sqrt{n}}\right) = 2\exp\left(-2\lambda^2\right)$
- Set $\frac{\lambda}{\sqrt{n}} = \eta$ and $\delta = 2\exp\left(-2\lambda^2\right)$; solve for n
- $\lambda = \eta\sqrt{n}$
- $\delta = 2\exp\left(-2\eta^2 n\right)$
- $\log\frac{\delta}{2} = -\eta^2 n$
- $n = \frac{1}{\eta^2}\log\frac{2}{\delta}$
- If $n > \frac{1}{\eta^2}\log\frac{2}{\delta}$ then w.p. $1-\delta$, the true error rate will be bounded by $1-(\epsilon+\eta)$



Related Work

- Geifman & El Yaniv (2017)
  - Develop confidence scores based on either the softmax ("SR") or Monte Carlo dropout ("MC-dropout")
  - Binary search for the threshold
  - Use an exact Binomial confidence interval instead of Massart's bound
  - Union bound over the binary search queries



(c) Image-Net

Cost-Sensitive Rejection

- Cost Matrix
- Optimal Classifier
  - For $\hat{p}(y = 1|x) \geq \tau_1$, predict 1
  - For $\hat{p}(y = 2|x) \geq \tau_2$, predict 2
  - Else REJECT
- Search all pairs (τ_1, τ_2) to minimize expected cost
- Pietraszek (2005) provides a fast algorithm based on (a) isotonic regression and (b) computing the slopes on the ROC curve corresponding to τ_1 and τ_2

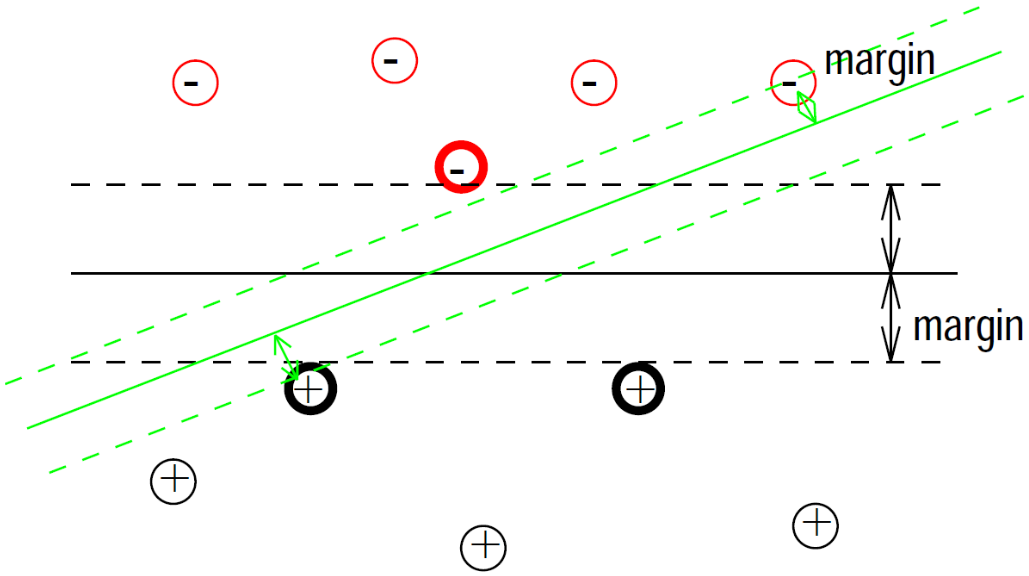
	Actions		
Probabilities	Predict 1	Predict 2	Reject
$P(y = 1\|x)$	0	c_{12}	c_{1r}
$P(y = 2\|x)$	c_{21}	0	c_{2r}

Support Vector Machines

- Key insight: Maximize the Margin around the Decision Boundary
- Three strategies:
  - Fit standard SVM, then calibrate or threshold
  - Fit a double-hinge loss (DHL) SVM that maximizes margin around the rejection thresholds
  - Fit two separate functions (classifier and rejection function) that maximize margins around the rejection thresholds

Reminder: Standard SVM

- Linear classifier that maximizes the margin between positive and negative examples
- $y \in \{+1, -1\}$ so $y_i f(x_i) > 0$ means x_i is classified correctly



$$\min_{w,b,\xi} C\|w\|^2 + \sum_i \xi_i \text{ subject to}$$

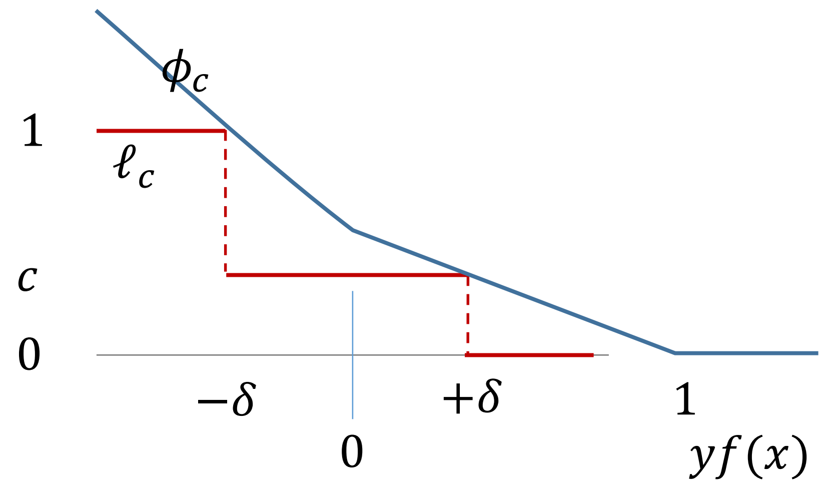
$$y_i(w^\top x_i + b) + \xi_i \geq 1 \ \ \forall i$$

- The ξ_i are "slack variables" the measure how "wrong" we are classifying x_i
- C is the regularization parameter

Double Hinge Loss

(Herbei & Wegkamp, 2006; Bartlett & Wegkamp, 2008)

- Assume cost of rejection is c
- Reject if $|f(x)| < \delta$
- Loss function $\ell_c\big(yf(x)\big)$
  - if $yf(x) < -\delta$ $\quad \ell_c = 1$
  - if $yf(x) \in [-\delta, +\delta]$ $\ell_c = c$
  - if $yf(x) > +\delta$ $\quad \ell_c = 0$
- Convex upper bound ϕ_c
  - if $yf(x) < 0$ $\quad \phi_c = 1 - ayf(x)$
  - if $yf(x) \in [0,1)$ $\quad \phi_c = 1 - yf(x)$
  - if $yf(x) > 1$ $\quad \phi_c = 0$



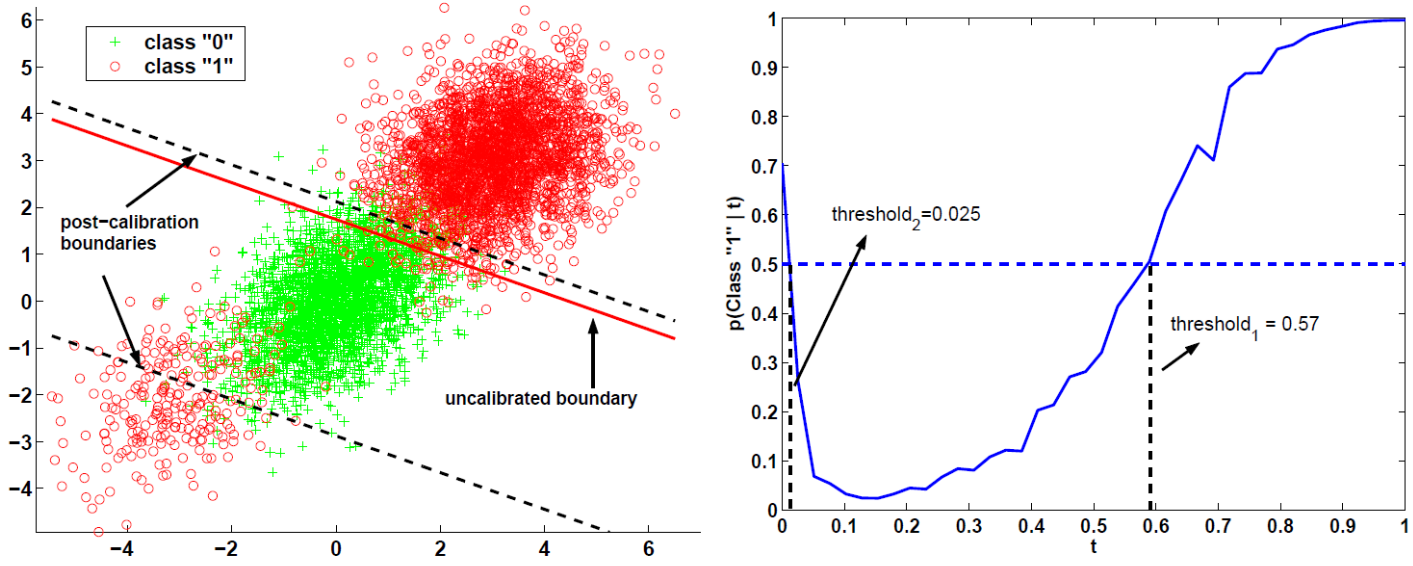
DHL Optimization Problem

- $\min\limits_{w,b,\xi,\gamma} \sum_i \xi_i + \frac{1-2c}{c}\gamma_i$ subject to
- $y_i(w^\top x_i + b) + \xi_i \geq 1$
- $y_i(w^\top x_i + b) + \gamma_i \geq 0$
- $\xi_i \geq 0; \gamma_i \geq 0$
- $\sum_{i,j} \alpha_i \alpha_j (x_i \cdot x_j) \leq r^2$ (regularization constraint)

- This is a quadratically-constrained quadratic program, so it can be solved, but it is not easy

Non-Optimal Case (2)

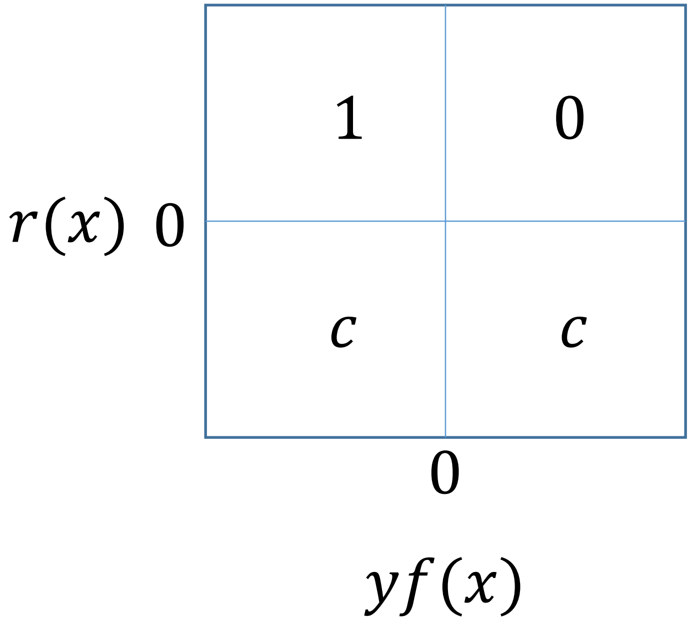
- Defining the rejection function in terms of h assumes that the probability of error is monotonically related to $\hat{p}(y|x)$.
- We saw last lecture that this is not necessarily true
- We can try to fix h or we can learn a more complex r function
- Unlikely to be a problem for flexible models, but could be a problem for linear and SVM methods



(a) (b)

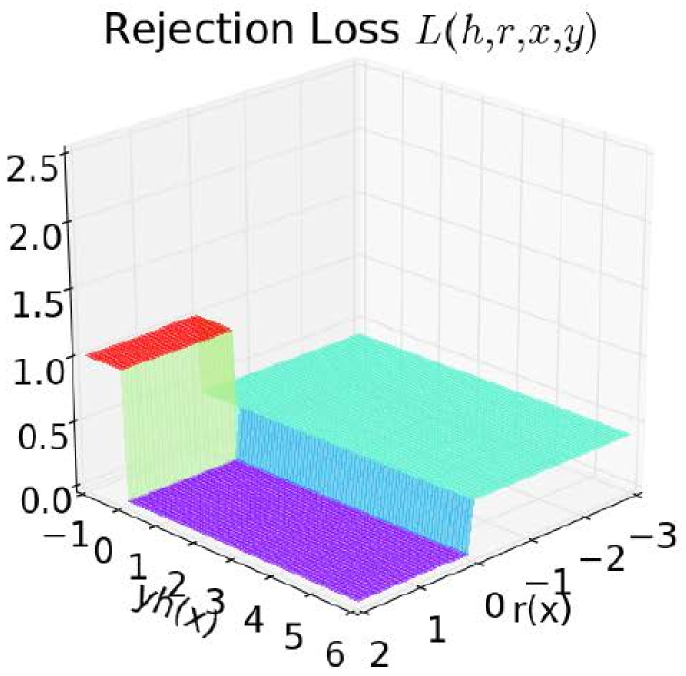
Method 3: Learn (f, r) pair
(Cortes, DeSalvo & Mohri, 2016)

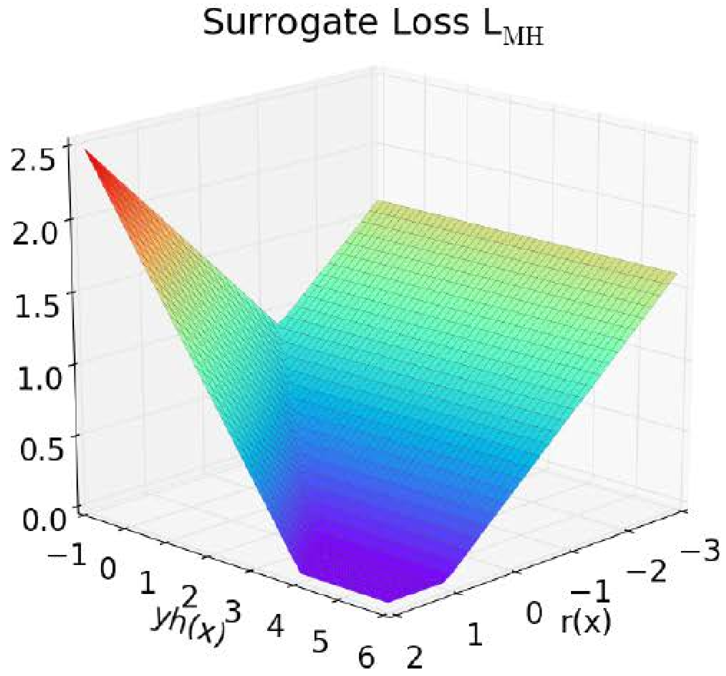
- Two-dimensional loss function
- If $r(x) \geq 0$ and $yf(x) \geq 0$ loss = 0
- If $r(x) \geq 0$ and $yf(x) < 0$ loss = 1
- If $r(x) < 0$ loss = c



Convex Upper Bound

- $L_{MH}(r, f, x, y) = \max\left(1 + \frac{1}{2}\left(r(x) - yf(x)\right), c\left(1 - \frac{1}{1-2c} r(x)\right), 0\right)$



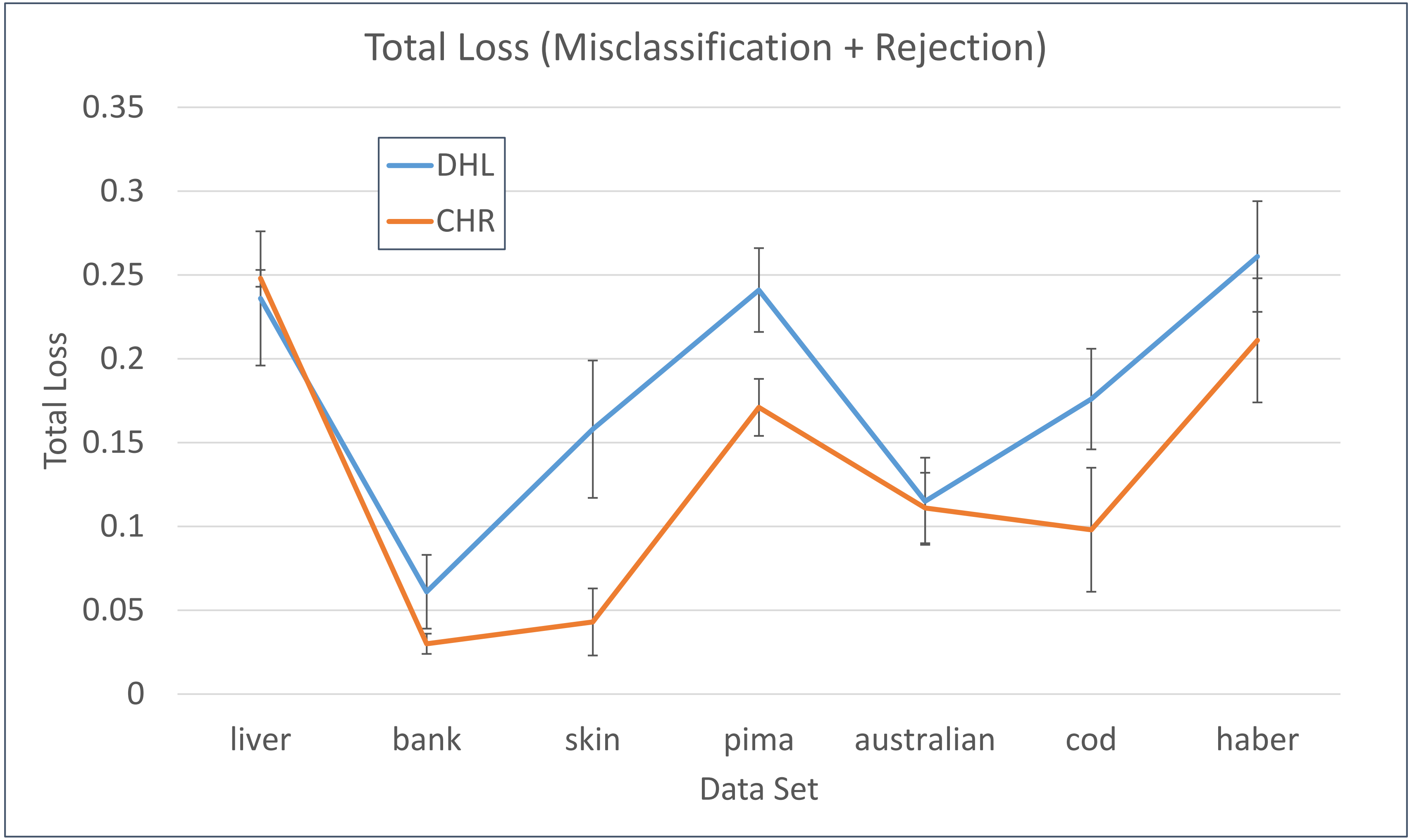


CHR Optimization Problem

- $f(x) = w^\top x + b$
- $r(x) = u^\top x + b'$

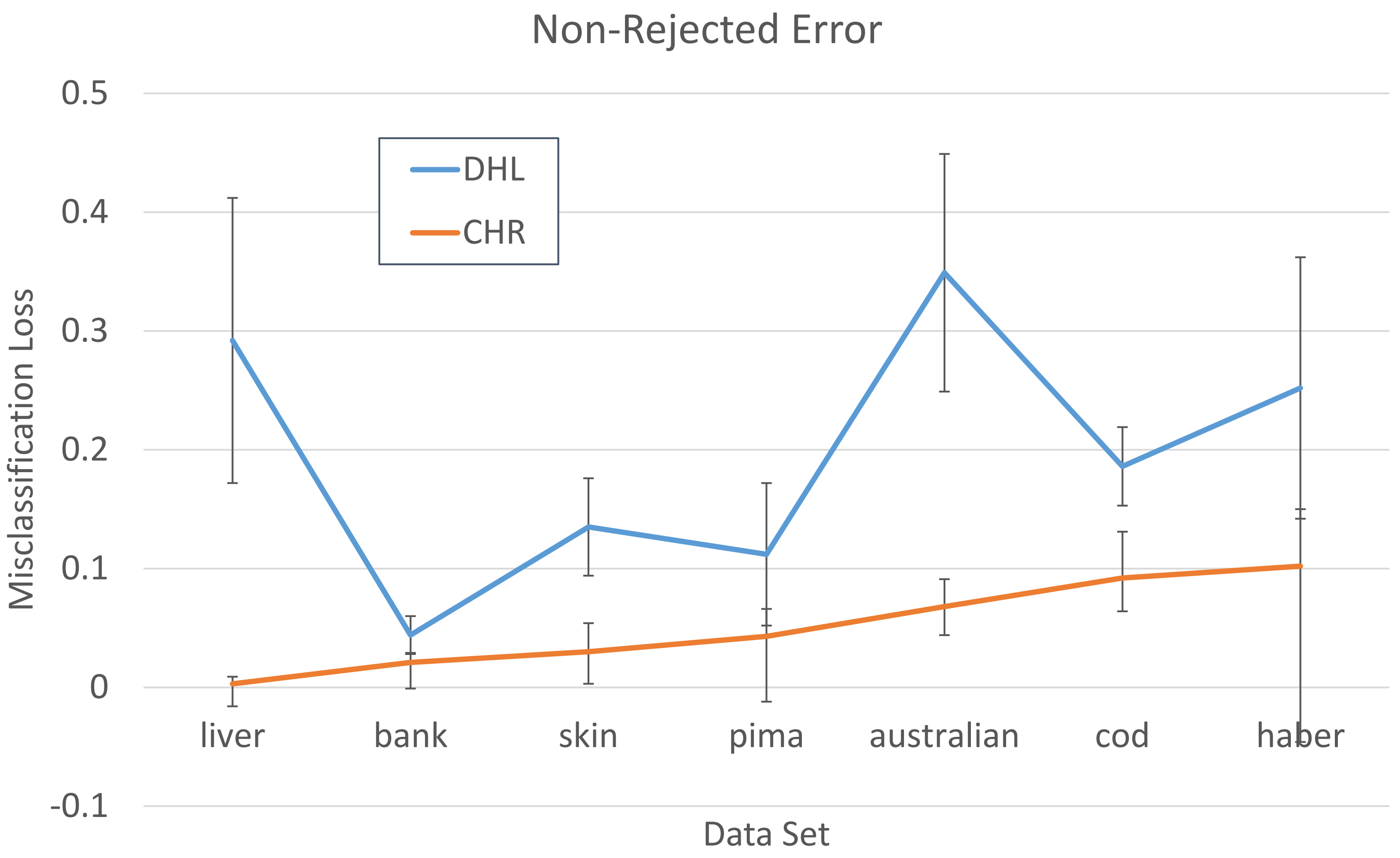
- $\min_{w,u,\xi} \frac{\lambda}{2} \|w\|^2 + \frac{\lambda'}{2} \|u\|^2 + \sum_i \xi_i$ subject to
  - $c\left(1 - \frac{1}{1-2c}(u^\top x_i + b')\right) \leq \xi_i$
  - $\frac{1}{2}(u^\top x_i + b' - y_i w^\top x_i - b) \leq 1 + \xi_i$
  - $0 \leq \xi_i$
- By minimizing ξ_i we are minimize the max of these three terms

Experimental Tests



Error on Non-Rejected Points

Note: DHL modified to reject the same number of points as CHR



Reject Option Conclusions

- Basic thresholding is easy and gives PAC guarantees
- 2-class thresholding with differential costs is easy
- K-class thresholding?
- Thresholding SVMs is interesting
  - Focus the “margin” on the reject boundaries
  - Learning a (f, r) pair is better than optimizing the double hinge loss
- Open question: How to jointly train DNNs and a rejection function

Conformal Prediction (online version)

- Given:
  - Training data $\llbracket z_1, \ldots, z_{n-1} \rrbracket$ where $z_i = (x_i, y_i)$
  - Classifier f trained on the training data
  - Nonconformity measure $A_n: \mathcal{Z}^{n-1} \times \mathcal{Z} \mapsto \mathbb{R}$
  - Query x_n
  - Accuracy level δ
- Find:
  - A set $C(x_q) \subseteq \{1, \ldots, K\}$ such that $y_q \in C(x_q)$ with probability $1 - \delta$
- Method:
  - For each k, let $z_n^k = (x_q, k)$
    - $\forall i\ \alpha_i^k \coloneqq A(\llbracket z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_n \rrbracket, z_i)$ “how different is z_i from the rest of the z values?
    - Let p^k = fraction of $\llbracket \alpha_1^k, \ldots, \alpha_n^k \rrbracket$ that are $\geq \alpha_n^k$
  - $C(x_q) = \{k | p^k \geq \delta\}$
  - Output $C(x_q)$

Examples of Nonconformity Measures

- Conditional probability method:
  - Train a probabilistic classifier f on $\llbracket z_1, \dots, z_{i-1}, z_{i+1}, \dots, z_n \rrbracket$
  - Then compute $A(\llbracket z_1, \dots, z_{i-1}, z_{i+1}, \dots, z_n \rrbracket, z_i) = -\log f(z_i)$
- Nearest neighbor nonconformity
  - $A(B, z) = \dfrac{\text{distance to nearest } z' \in B \text{ in same class}}{\text{distance to nearest } z' \in B \text{ in different class}}$

Additional Information

- In addition to outputting $C(x_q)$, we can output
  - $\hat{y}_q = \arg\max_k p^k$ (the best prediction)
  - $p_q = \max_k p^k$ (the p-value of the best prediction)
  - $1 - \max_{k; k \neq \hat{y}_q} p^k$ (the "confidence". We have more confidence if the second-best p-value is small)

Batch (“inductive”) Conformal Prediction

- Divide data into training and calibration
- Train f on the training data
- Let $[\![z_1, \ldots, z_n]\!]$ be the validation data
- Let $\alpha_1, \ldots, \alpha_n$ be the non-conformity scores of the validation data
  - $\alpha_i \coloneqq A([\![z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_n]\!], z_i)$
- Given query x_q
  - For $k = 1, \ldots, K$
    - Let $z_q^k = (x_q, k)$
    - let $\alpha_q^k = A\left([\![z_1, \ldots, z_n]\!], z_q^k\right)$
    - Let p^k = fraction of $\left[\!\!\left[\alpha_1, \ldots, \alpha_n, \alpha_q^k\right]\!\!\right]$ that are $\geq \alpha_q^k$
  - $C\left(x_q\right) = \left\{k \middle| p^k \geq \delta\right\}$

- Key difference: z_q^k does not affect the other non-conformity scores

Almost Equivalent to Learning a Threshold

- Let $\tau =$ the δ quantile of $[\![\alpha_1, \dots, \alpha_n]\!]$
- Given query x_q
  - For $k = 1, \dots, K$
    - Let $z_q^k = (x_q, k)$
    - let $\alpha_q^k = A\left([\![z_1, \dots, z_n]\!], z_q^k\right)$
  - $C\left(x_q\right) = \left\{k \middle| \alpha_q^k \geq \tau\right\}$

- Additional difference: τ is computed without considering α_q^k
- IF n is large enough, this does not matter

Experimental Results

	Satellite	Shuttle	Segment
Hidden Units	23	12	11
Hidden Learning Rate	0.002	0.002	0.002
Output Learning Rate	0.001	0.001	0.001
Momentum Rate	0.1	0	0.1

- Non-conformity Measures
  - Resubstitution:
    - Train f on all data
    - Let $\hat{y}_i = f(x_i)$
    - $A\left(\llbracket z_1, \ldots, z_{i-1}, z_i, \ldots, z_N \rrbracket, (x_i, k)\right) = I[\hat{y}_i = k]$
  - Leave One Out:
    - Train f on $\llbracket z_1, \ldots, z_{i-1}, z_i, \ldots, z_N \rrbracket$
    - Let $\hat{y}_i = f(x_i)$
    - $A\left(\llbracket z_1, \ldots, z_{i-1}, z_i, \ldots, z_N \rrbracket, (x_i, k)\right) = I[\hat{y}_i = k]$

Satellite

Error:
$y_i \notin C(x_i)$

Nonconformity Measure	Confidence Level	Only one Label (%)	More than one label (%)	No Label (%)	Errors (%)
Resubstitution	99%	60.72	39.28	0.00	1.11
	95%	84.42	15.58	0.00	4.67
	90%	96.16	3.02	0.82	9.59
Leave one out	99%	61.69	38.31	0.00	1.10
	95%	85.70	14.30	0.00	4.86
	90%	96.11	3.10	0.79	9.43

Table 3. Results of the second mode of the Neural Networks ICP for the Satellite data set.

Shuttle

Nonconformity Measure	Confidence Level	Only one Label (%)	More than one label (%)	No Label (%)	Errors (%)
Resubstitution	99%	99.23	0.00	0.77	0.77
	95%	93.52	0.00	6.48	6.48
	90%	89.08	0.00	10.92	10.92
Leave one out	99%	99.30	0.00	0.70	0.70
	95%	93.86	0.00	6.14	6.14
	90%	88.72	0.00	11.28	11.28

Table 4. Results of the second mode of the Neural Networks ICP for the Shuttle data set.

Segmentation

Nonconformity Measure	Confidence Level	Only one Label (%)	More than one label (%)	No Label (%)	Errors (%)
Resubstitution	99%	90.69	9.31	0.00	0.95
	95%	97.71	1.25	1.04	3.68
	90%	94.68	0.00	5.32	6.71
Leave one out	99%	91.73	8.27	0.00	1.04
	95%	97.79	1.21	1.00	3.55
	90%	94.76	0.00	5.24	6.67

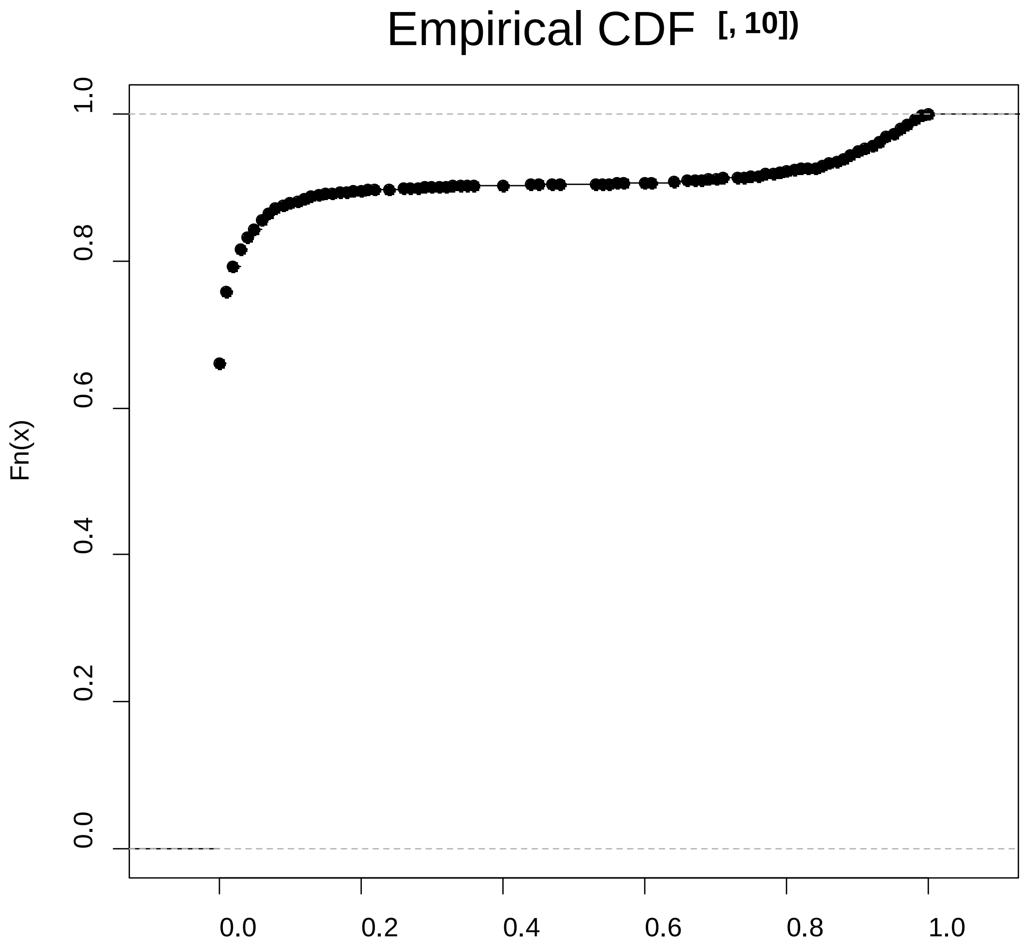
Table 5. Results of the second mode of the Neural Networks ICP for the Segment data set.

Pendigits + Random Forest

(Dietterich, unpublished)

- Train a random forest on half of UCI Training Set
- Use the predicted class probability $P(y = k|x)$ as the (non)conformity score
- Compute τ values using other half of Training Set
- Compute C on the Test Set

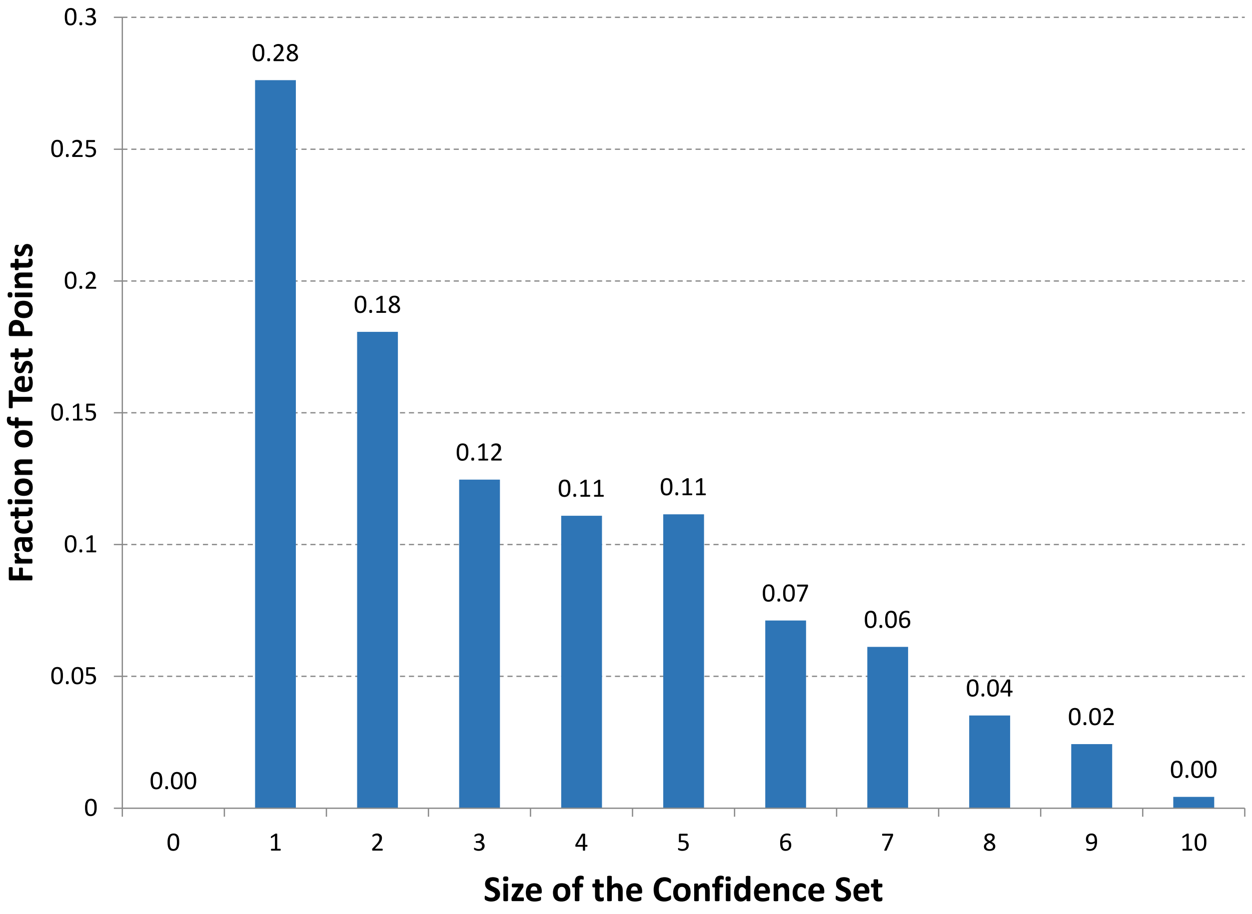
Cumulative Distribution Function for Class “9”



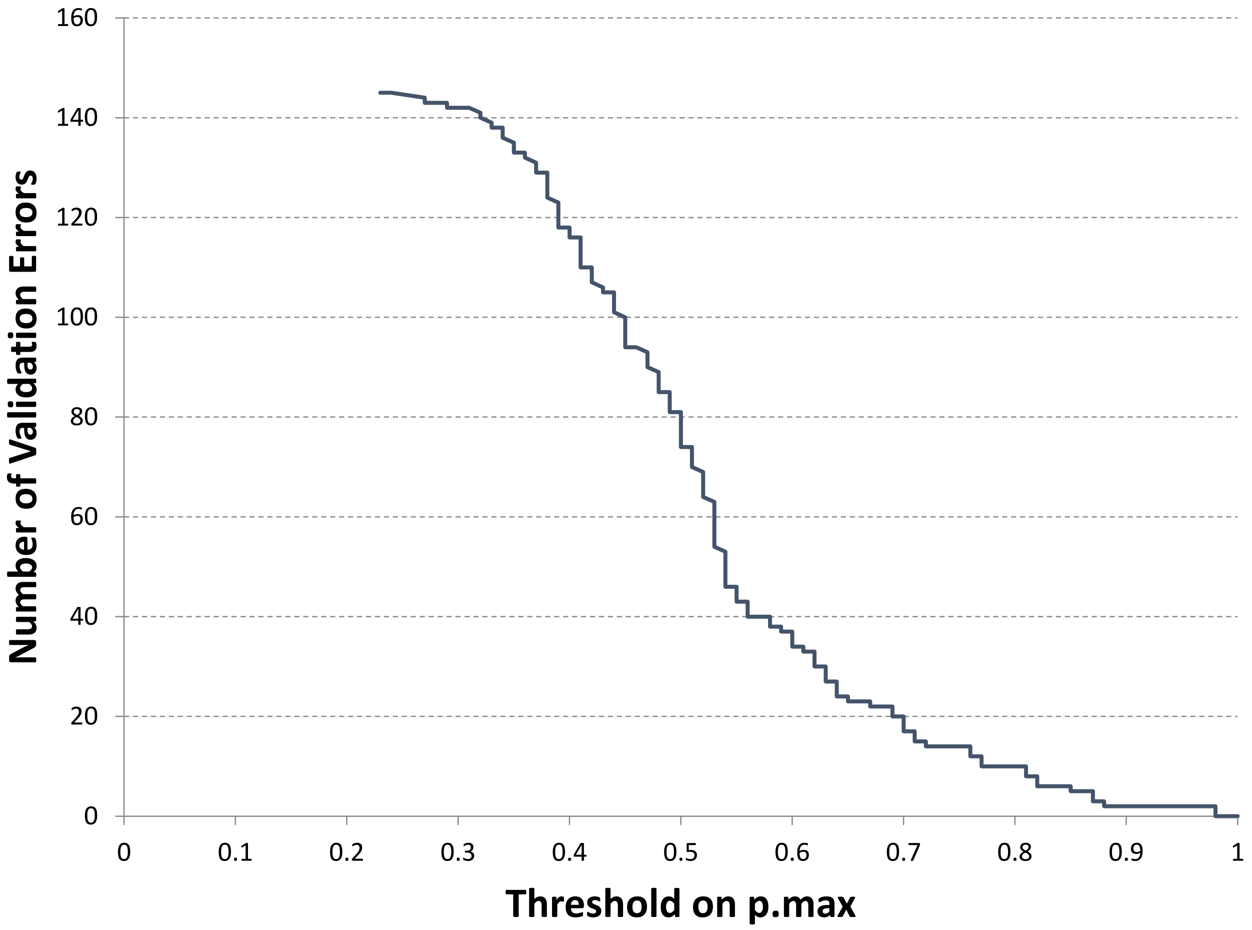
$C(x, "9")$

Pendigits Results

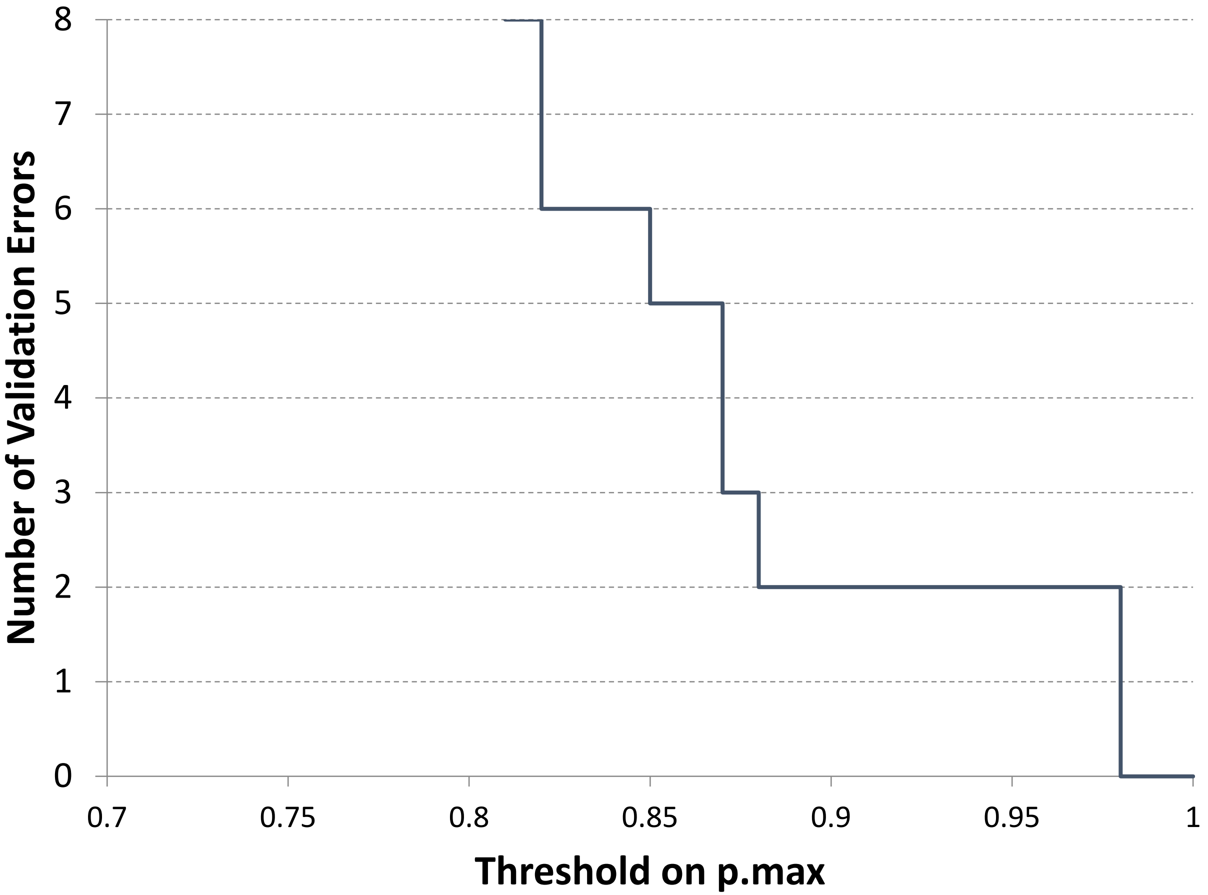
- All τ values were 0 (for $\epsilon = 0.001$)
- Probability $y \in \Gamma(x) = 0.9997$
- Abstention rate = 0.72
- Sizes of prediction sets Γ:



Simple Thresholding of $\max_k \hat{p}(y = k|x)$



Zoomed In: $\tau = 0.87$ for $\delta = 0.05$



Test Set Results

- Probability of correct classification: 0.9987
- Rejection rate: 33.4%
  - [Conformal prediction was 72%]

Another Use Case: Lexicon Reduction

- US Postal Service Address Reading Task
  - (Madhvanath, Kleinberg, Govindaraju, 1997)
- Two classifiers
  - Method 1: Fast but not always accurate
  - Method 2: Slower but more accurate
    - Can only afford to run on 1/3 of envelopes
    - Faster if it can be focused on a subset of the classes
- Apply conformal prediction using Method 1
  - Eliminate as many classes as possible
  - Apply Method 2 if $|C(x)| > 1$

Summary

- Lecture 1: Calibration
- Lecture 2: Rejection
  - Method 1: Threshold f with single or multiple thresholds
    - Multiple thresholds requires a change in the SVM methodology
  - Method 2: Learn a separate rejection function and threshold it
  - Method 3: Conformal: Use thresholding to construct a *confidence set*
    - Reject if $\left|C\left(x_q\right)\right| \neq 1$
    - Can perform “lexicon reduction”
  - In my experience, Conformal Prediction is not good for Rejection, but more experiments are needed

Next Lecture

- All of these methods assume a closed world
- What happens when queries may belong to “alien” classes not observed during training?
- Papers:
  - Bendale, A., & Boult, T. (2016). Towards Open Set Deep Networks. In CVPR 2016 (pp. 1563–1572). http://doi.org/10.1109/CVPR.2016.173
  - Liu, S., Garrepalli, R., Dietterich, T. G., Fern, A., & Hendrycks, D. (2018). Open Category Detection with PAC Guarantees. *Proceedings of the 35th International Conference on Machine Learning, PMLR*, *80*, 3169–3178. http://proceedings.mlr.press/v80/liu18e.html

Citations

- Bartlett, P., Wegkamp, M. (2008). Classification with a reject option using a hinge loss. JMLR, 2008.
- Chow (1970). On optimum recognition error and reject trade-off. IEEE Transactions on Computing.
- Cortes, C., DeSalvo, G., & Mohri, M. (2016). Learning with rejection. *Lecture Notes in Artificial Intelligence*, *9925 LNAI*, 67–82. http://doi.org/10.1007/978-3-319-46379-7_5
- Geifman, Y., El Yaniv, R. (2017) Selective Classification for Deep Neural Networks. NIPS 2017. arXiv: 1705.08500
- Herbei, R., Wegkamp, M. (2005). Classification with reject option. Canadian Journal of Statistics.
- Madhvanath, S., Kleinberg, E., Govindaraju, V. (1997). Empirical Design of a Multi-Classifier Thresholding/Control Strategy for Recognition of Handwritten Street Names. *International Journal of Pattern Recognition and Artificial Intelligence, 11(6):*933-946. https://doi.org/10.1142/S0218001497000421
- Papadopoulos, H. (2008). Inductive Conformal Prediction: Theory and Application to Neural Networks. Book chapter. https://www.researchgate.net/publication/221787122_Inductive_Conformal_Prediction_Theory_and_Application_to_Neural_Networks
- Pietraszek, T. (2005). Optimizing abstaining classifiers using ROC analysis. In ICML, 2005
- Shafer, G., & Vovk, V. (2008). A tutorial on conformal prediction. Journal of Machine Learning Research, 9, 371–421. Retrieved from http://arxiv.org/abs/0706.3188