Differential Geometric Regularization for Supervised Learning of Classifiers

Qinxun Bai

Boston University

Visual Recognition

?

Cat

⋮

$\boxed{\text{Dog}}$

⋮

Backpack

⋮

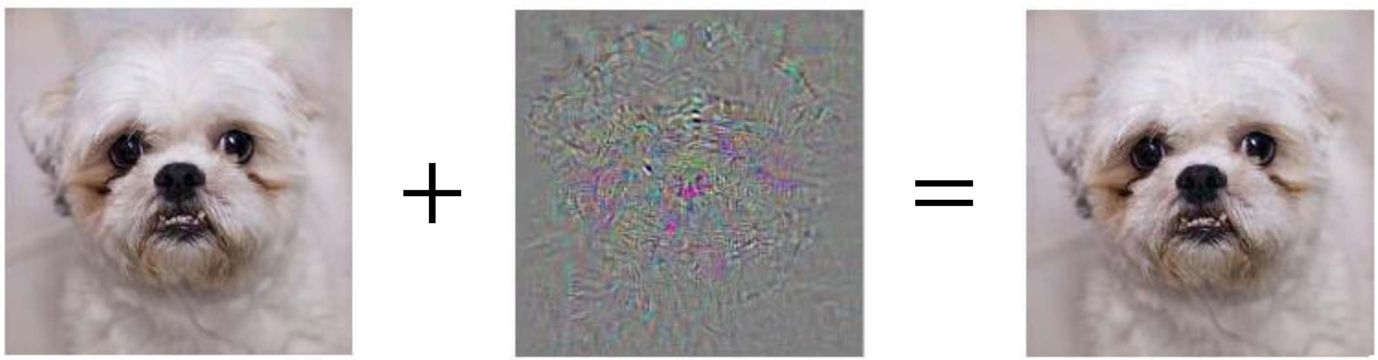
Supervised learning of classifiers

- State-of-the-art on ImageNet Challenge

  Human level: classification error $< 4\%$

Counter-Intuitive Properties

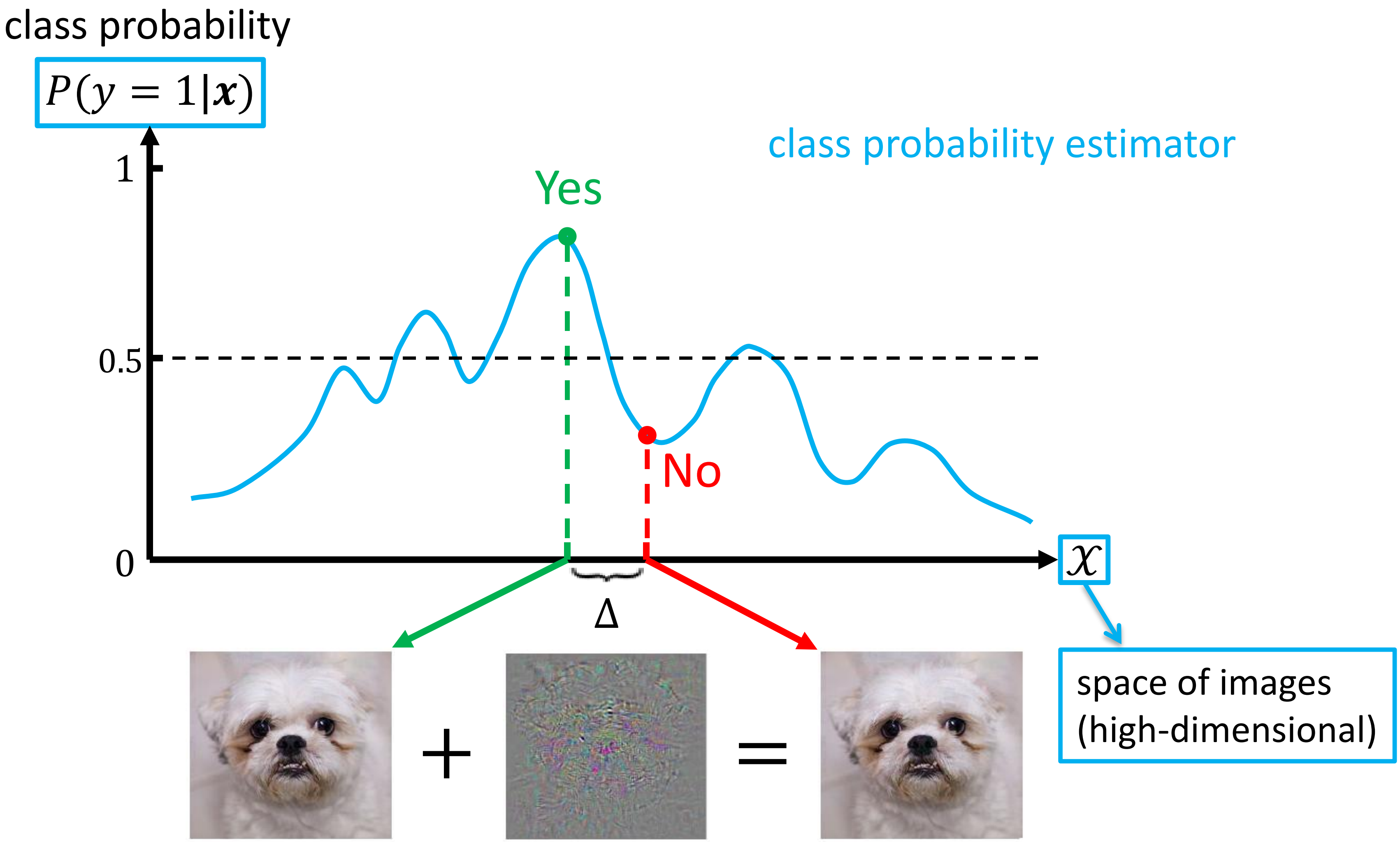
Dog? Yes No



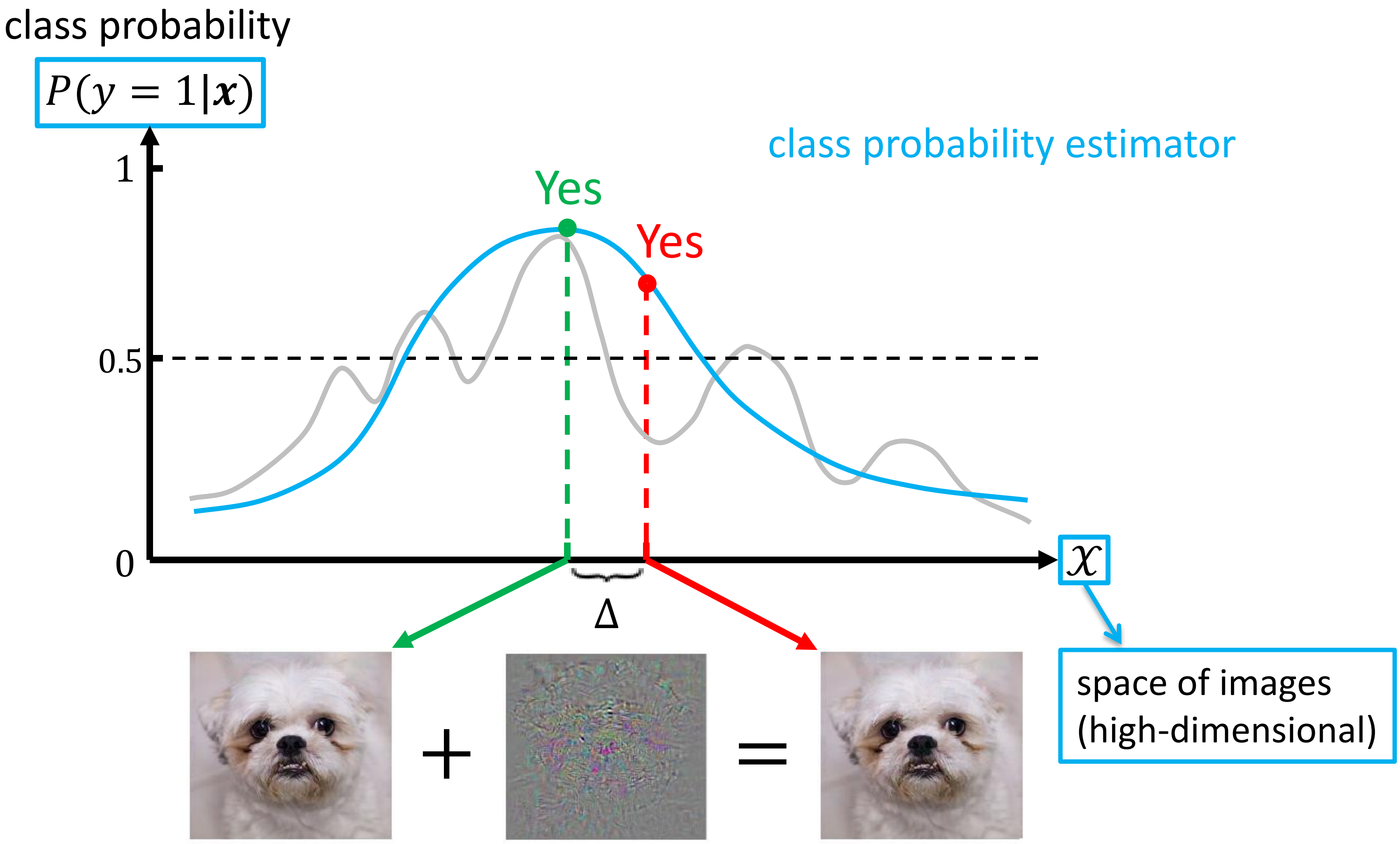
training testing

Fool DNN by hardly perceptible perturbation [*Szegedy et al. 2013*]

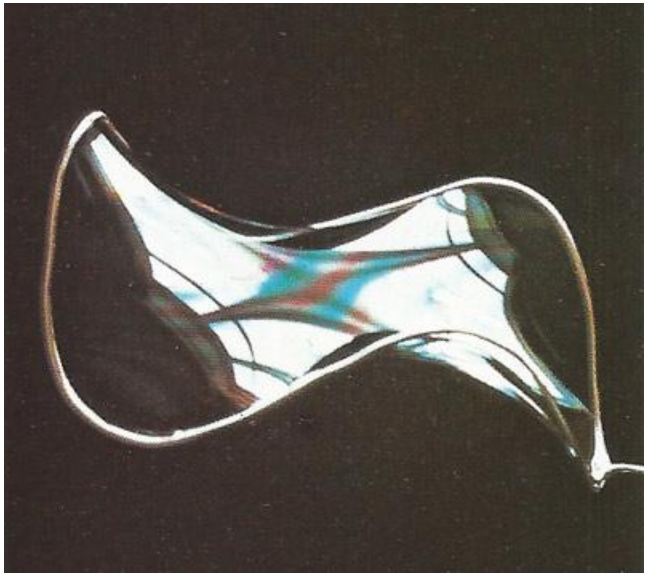
Rapid Local Oscillation



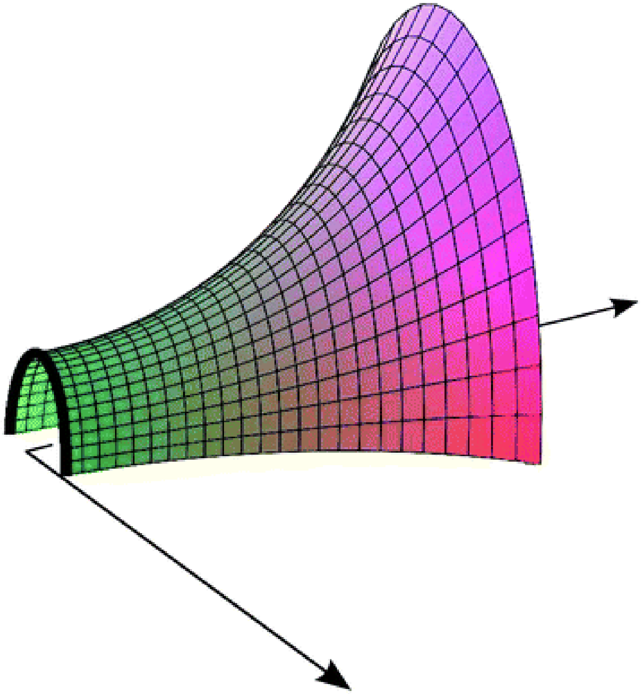
Rapid Local Oscillation



Geometric Idea: Minimal Surfaces



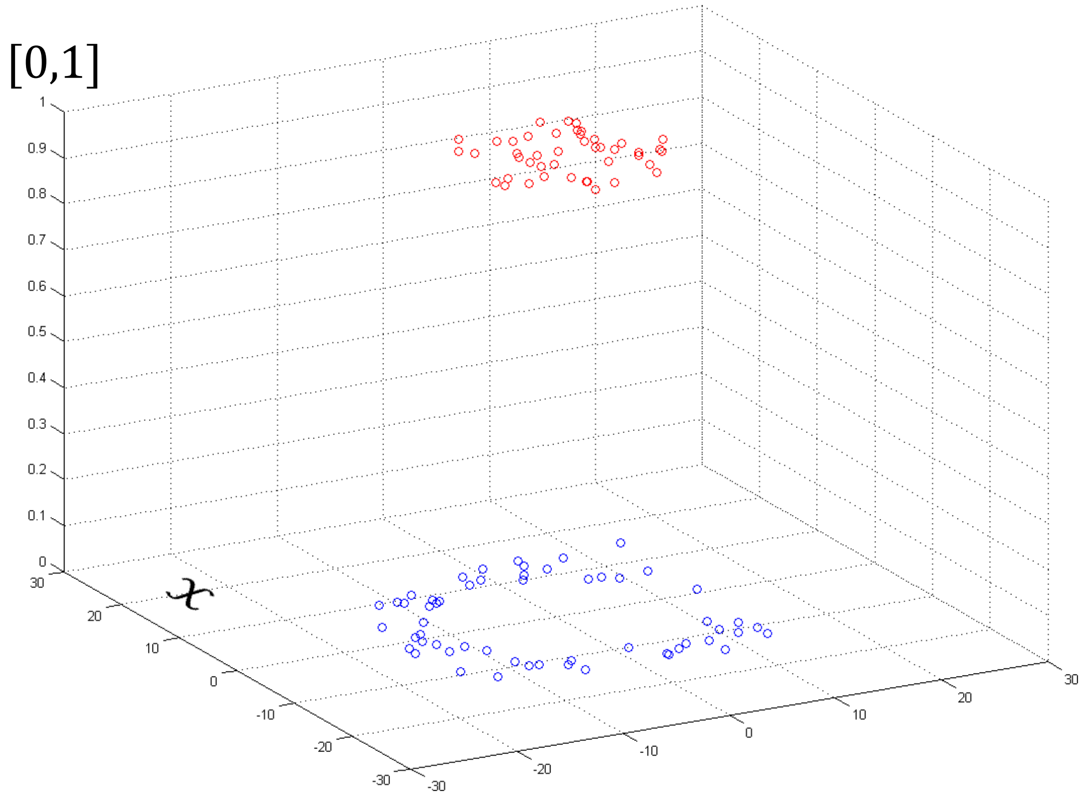
soap film



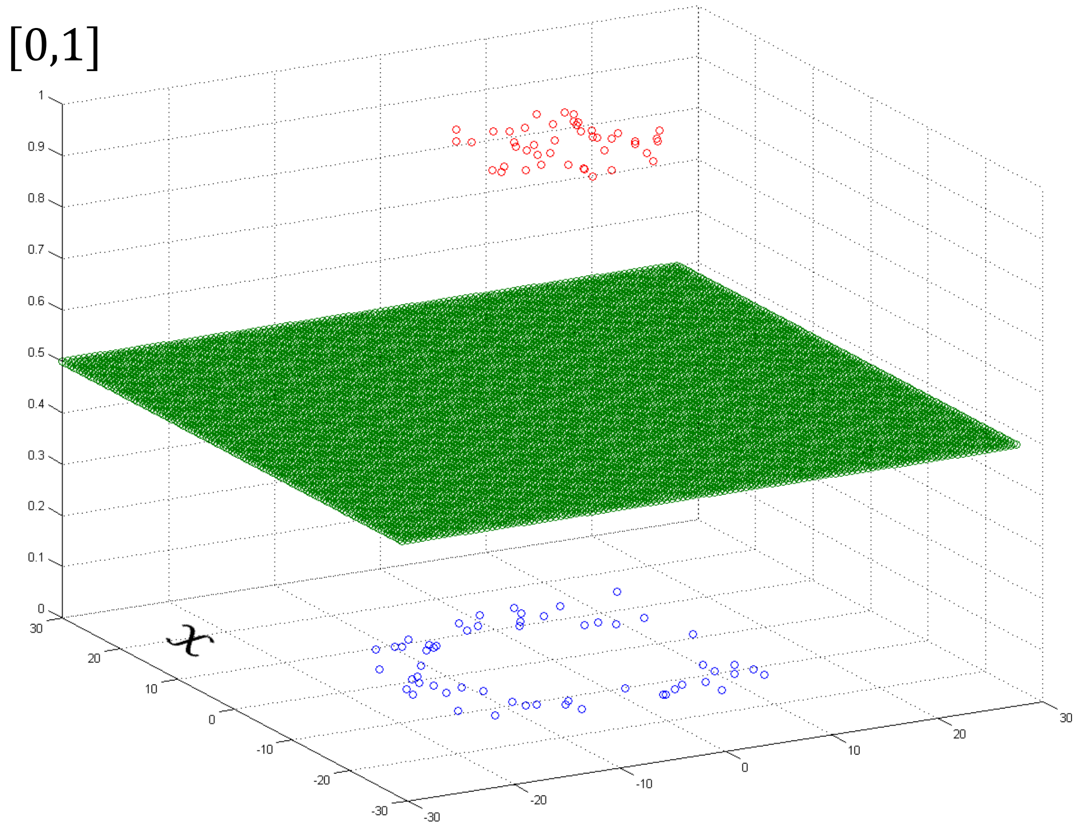
catenoid

image credit: Google image search

Physical Model

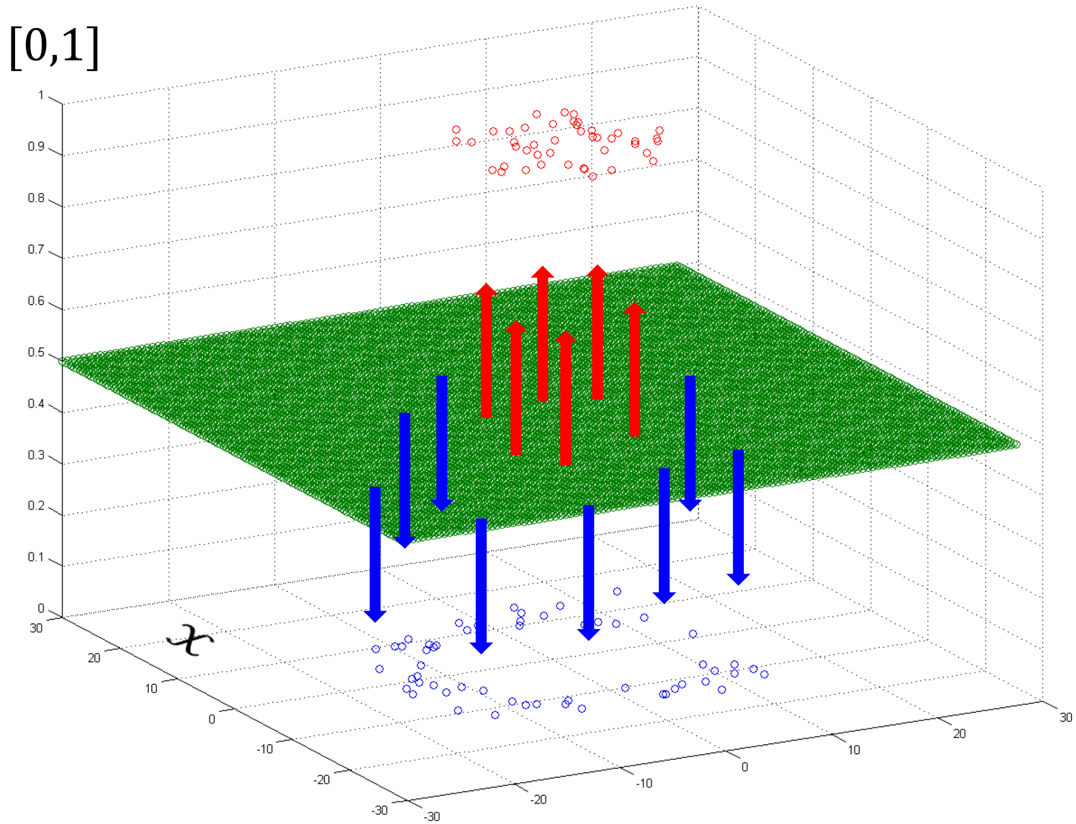


Physical Model



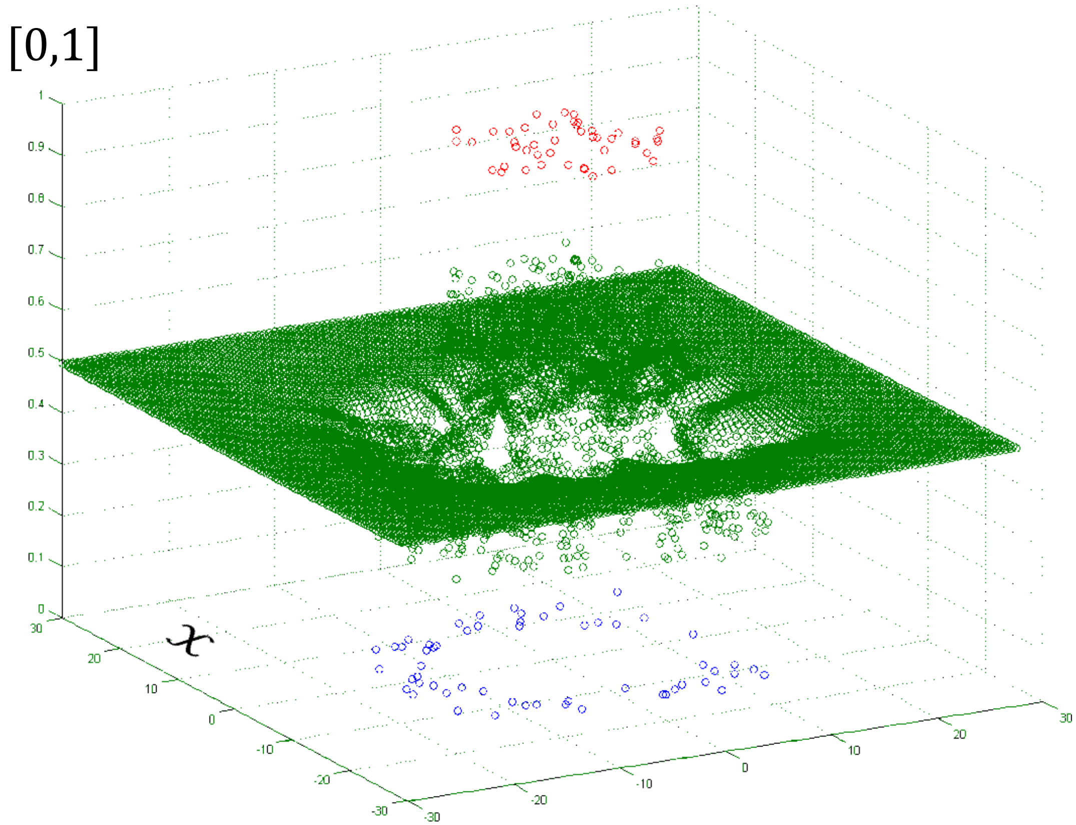
Initial hyper-surface

Physical Model



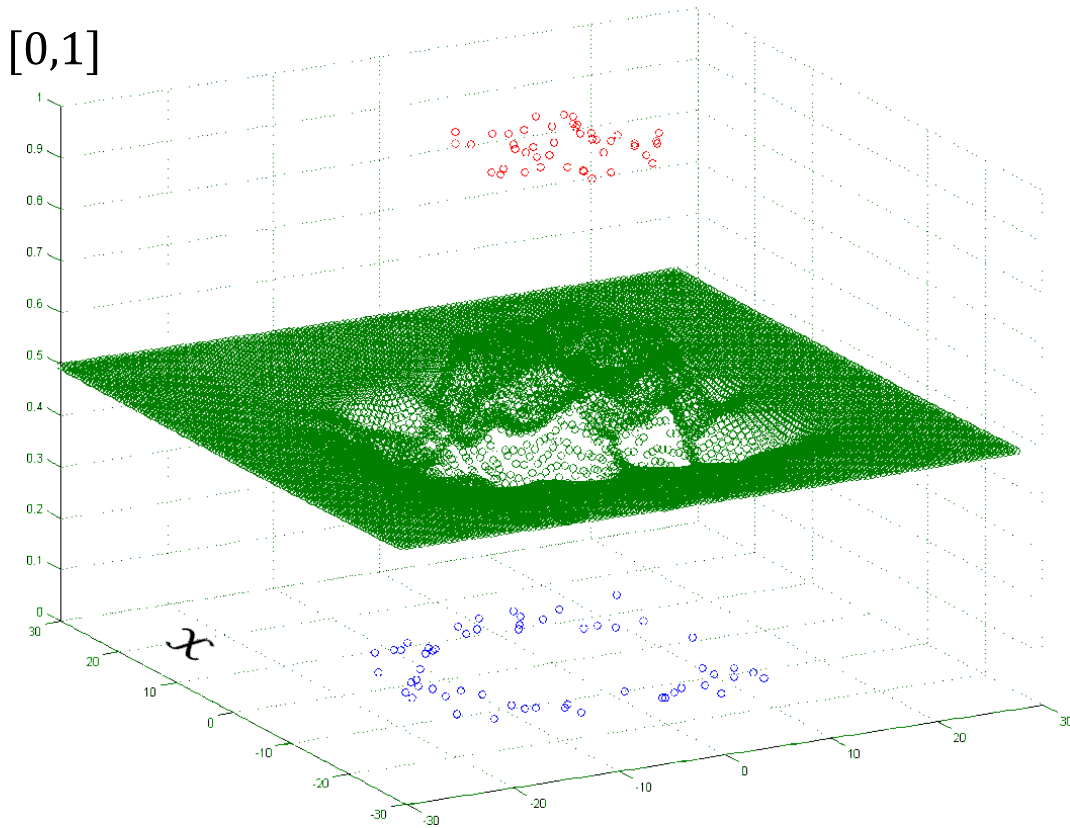
hyper-surface deforms towards training data
as if attracted by gravitational force due to point masses

Physical Model



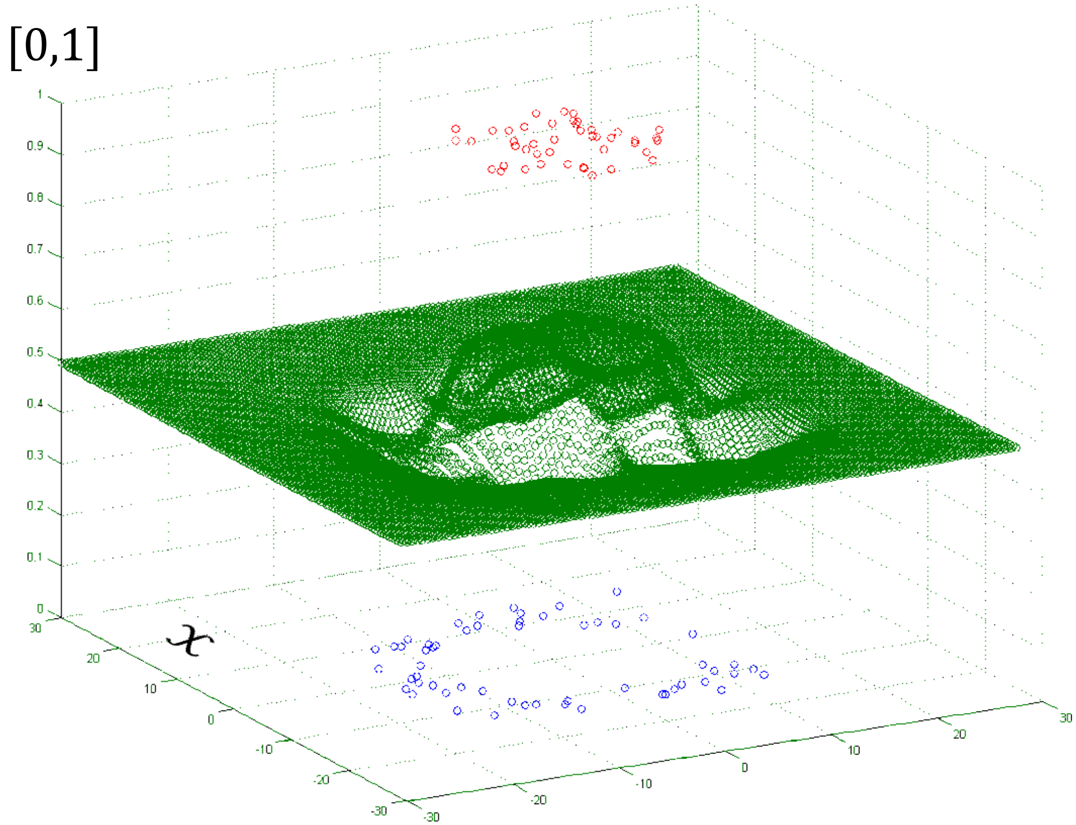
hyper-surface deforms towards training data
as if attracted by gravitational force due to point masses

Physical Model



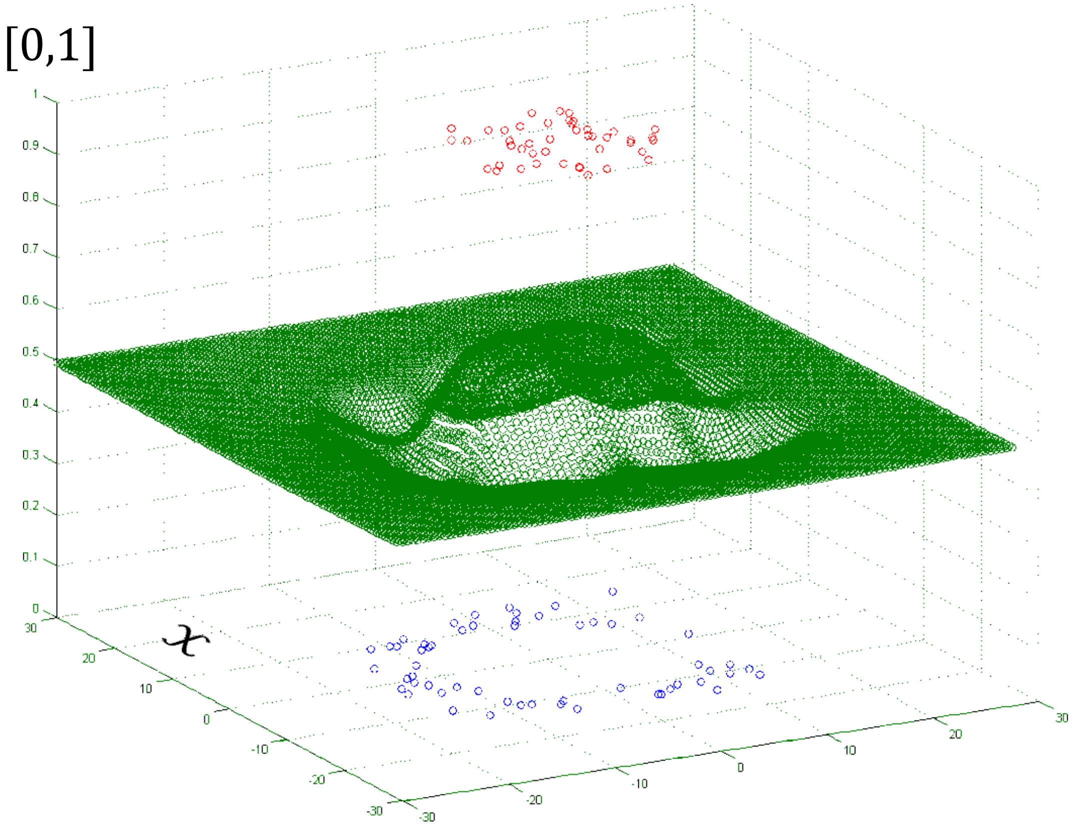
hyper-surface remains as tight as possible
as if in the presence of surface tension

Physical Model



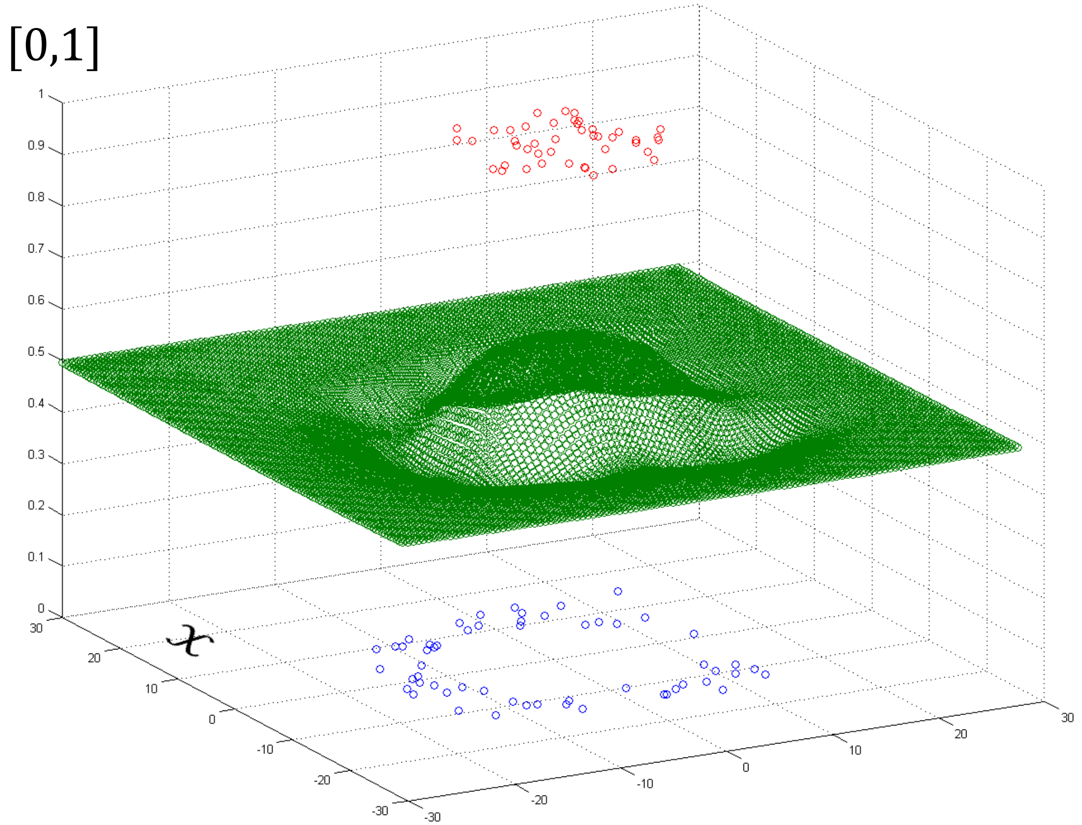
hyper-surface remains as tight as possible
as if in the presence of surface tension

Physical Model



hyper-surface remains as tight as possible
as if in the presence of surface tension

Physical Model



hyper-surface remains as tight as possible
as if in the presence of surface tension

Formal Setup

Learn a function $\boldsymbol{f}: \underbrace{\mathcal{X}}_{\text{input feature space}} \longrightarrow \underbrace{\triangle^{K}}_{\text{output probabilistic simplex for } K \text{ classes}}$ as an estimator of $P(y|\boldsymbol{x})$

input feature space

output probabilistic simplex for K classes

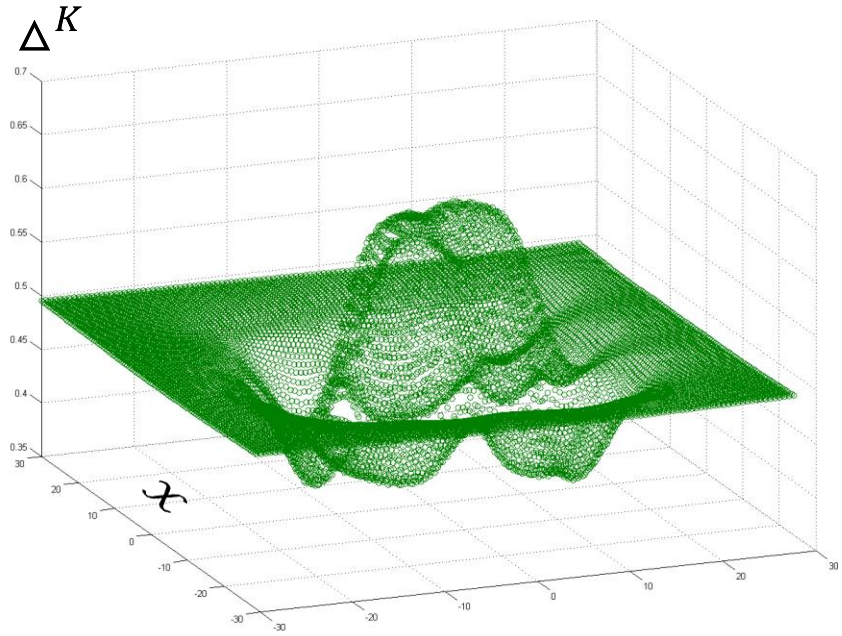
Formal Setup

Learn a function $\boldsymbol{f}: \mathcal{X} \longrightarrow \Delta^K$ as an estimator of $P(y|\boldsymbol{x})$

Hyper-surface associated with $\boldsymbol{f}$:

$$graph(\boldsymbol{f}) = \{\left(\boldsymbol{x}, f^1(\boldsymbol{x}), \cdots, f^K(\boldsymbol{x})\right) | \boldsymbol{x} \in \mathcal{X}\} \in \mathcal{X} \times \Delta^K$$

exploit the geometry of this hyper-surface!



Regularization Scheme

Minimize the regularized loss $\mathcal{P}$ in functional space $\mathcal{H}$

$$\min_{\boldsymbol{f} \in \mathcal{H}} \mathcal{P}(\boldsymbol{f}) = \min_{\boldsymbol{f} \in \mathcal{H}} \{\underline{L(\boldsymbol{f})} + \lambda \boxed{G(\boldsymbol{f})}\}$$

Data term
penalize the error of $\boldsymbol{f}$ in explaining the training data

Regularization term
penalize the volume of $graph(\boldsymbol{f})$

Effect of Regularization

Optimization perspective

$$\min_{f \in \mathcal{H}} \big(L(f) + \lambda G(f)\big) \Leftrightarrow \min_{f \in \mathcal{H}_\lambda} L(f)$$

functional space that the algorithm works in

shrunk functional space $\mathcal{H}_\lambda = \{f \in \mathcal{H}, G(f) \text{ bounded}\}$

- imposing $G(f) \Leftrightarrow$ shrinking $\mathcal{H} \rightarrow \mathcal{H}_\lambda$
- properly-shrink is the key for generalization
- sculpturing: λ is your hand, $G(f)$ is the knife!

Shrink the Search Space $\mathcal{H}$

Decomposition of excess error:

$$\underbrace{R(f)}_{\substack{\text{generalization}\\ \text{risk}}} - \underbrace{R(f^*)}_{\substack{\text{Bayes risk}\\ \text{(optimal)}}} = \big(R(f) - \underbrace{R(\mathcal{H})}_{\substack{\text{optimal risk}\\ \text{achievable in } \mathcal{H}}}\big) + (R(\mathcal{H}) - R(f^*))$$

Shrink the Search Space $\mathcal{H}$

Decomposition of excess error:

$$R(f) - R(f^*) = \underbrace{\big(R(f) - R(\mathcal{H})\big)}_{\text{estimation error}} + \underbrace{(R(\mathcal{H}) - R(f^*))}_{\text{approximation error}}$$

Concerning estimation error:

smaller $\mathcal{H}$, smaller estimation error

Concerning approximation error:

larger $\mathcal{H}$, smaller approximation error

A subtle trade-off: proper shrinking of $\mathcal{H}$, ideally:

$G(f)$ precisely encodes our prior & cross-validation on λ

Functional-norms: Smoothness

Functional perspective

- penalizing functional norm $\longrightarrow$ smoothness

Smoothness of different kinds:

- not specifically tailored to measure the amount of local oscillation
- overkill the hypothesis space
- Sculpturing with an axe? Need a sculptor’s knife!

Our Argument: Mean Curvature

Geometric perspective:

- $\{(\boldsymbol{x}, \boldsymbol{f}(\boldsymbol{x})) | \boldsymbol{x} \in \mathcal{X}\}$: a submanifold in $\mathcal{X} \times \Delta^K$

Mean Curvature of this submanifold:

- in differential geometric sense
- a specific measure of the amount of local oscillation
- generalizes to high dimensional space
- handles binary and multiclass uniformly

Existing Geometric Regularization

On geometry of the marginal distribution $P(\boldsymbol{x})$

- manifold regularization (Belkin *et al*. 2006)

On geometry of the decision boundary in $\mathcal{X}$

- level set based regularization

  Cai & Sowmya 2007; Varshney & Willsky 2010

- Euler's Elastica based regularization

  Lin *et al*. 2012; 2015

The small local oscillation of $\boldsymbol{\eta}(\boldsymbol{x})$ is Not captured

Solve for $\min_{\boldsymbol{f} \in \mathcal{H}} \mathcal{P}(\boldsymbol{f})$

Solving it directly is too difficult!

Solve iteratively by gradient flow: $\frac{d\boldsymbol{f}_t}{dt} = -\nabla\mathcal{P}$

- starting from neutral estimator $\boldsymbol{f}_0 = \left(\frac{1}{K}, \cdots, \frac{1}{K}\right)$
- evolve $\boldsymbol{f}_t$ towards $-\nabla\mathcal{P}$
- $\boldsymbol{f}_t$ will flow to a local minimum of $\mathcal{P}$

Algorithm (binary & multiclass)

Input: training data, trade-off λ, step-size τ

Initialize: $\boldsymbol{f}(\boldsymbol{x}_i; \boldsymbol{w}) = \left(\frac{1}{K}, \cdots, \frac{1}{K}\right), M = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}}$

For $t = 1$ to T

- Evaluate gradient vector $\boxed{\nabla\mathcal{P}}$ at every training point $\boldsymbol{x}_i$
- $\boldsymbol{w} \leftarrow \boldsymbol{w} - \tau M^{-1}[\nabla\mathcal{P}(\boldsymbol{x}_1), \cdots]^T$

Output: class probability $\boldsymbol{f}$

Solid math
Simple algorithm
Parallelizable!

Geometric Foundation on $\mathcal{H}$

$$\mathcal{H} = Maps(\mathcal{X}, \Delta^K), \mathcal{H}' = Maps(\mathcal{X}, \mathbb{R}^K)$$

Topology

- Frechet topology on $\mathcal{H}'$, and the induced topology on $\mathcal{H}$

  i.e. two functions in $\mathcal{H}$ are close if the functions and all their partial derivatives are pointwise close

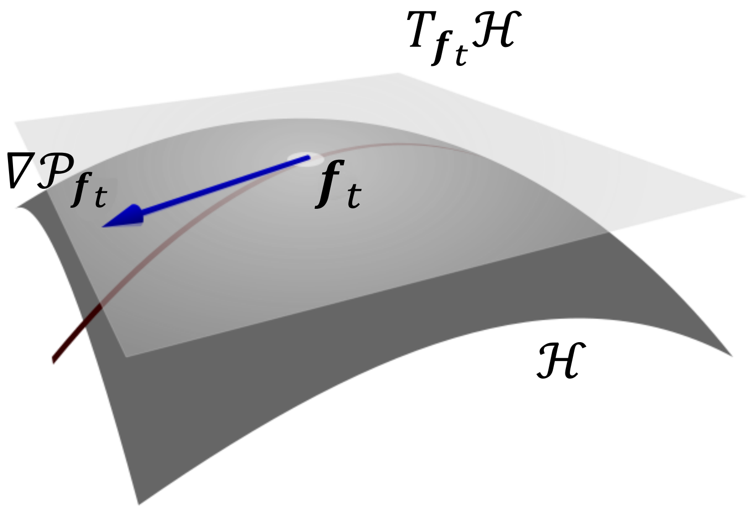
Riemannian metric

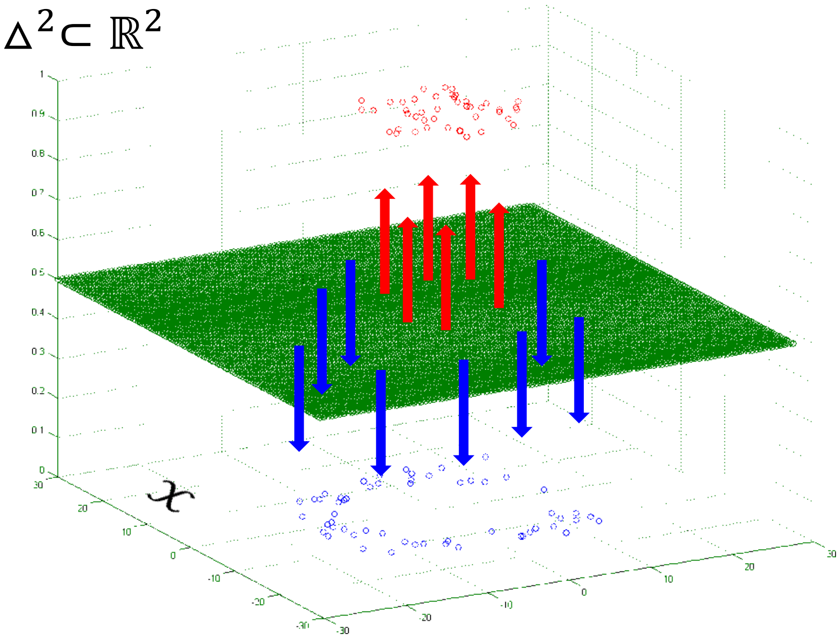
- Restrict the L^2 metric on $\mathcal{H}'$ to each tangent space $T_{\boldsymbol{f}}\mathcal{H}$

$$\langle \phi_1, \phi_2 \rangle = \int_{\mathcal{X}} \phi_1(\boldsymbol{x})\phi_2(\boldsymbol{x})dvol_{\boldsymbol{x}}$$

where $\phi_i \in \mathcal{H}'$ and $dvol_{\boldsymbol{x}}$ is the volume form of the induced Riemannian metric on $graph(\boldsymbol{f})$.

The Gradient $\nabla\mathcal{P}_{\boldsymbol{f}_t}$





$\nabla\mathcal{P}_{\boldsymbol{f}_t}$: tangent vector in $T_{\boldsymbol{f}_t}\mathcal{H}$ ⟷ vector field on $graph(\boldsymbol{f}_t)$

Computation of $\nabla\mathcal{P} = \nabla L + \lambda\nabla G$

Computing ∇L is easy

- *e.g.* back propagation for neural networks

Computing ∇G: mean curvature flow

- **Our Theorem**:

  need only 1st and 2nd partial derivatives of $\boldsymbol{f}$, rest of computation is just matrix manipulations

Empirical Data Term

Quadratic loss

$$L(\boldsymbol{f}) = \sum_{i=1}^{m} \|f(\boldsymbol{x}_i) - \boldsymbol{z}_i\|^2$$

Cross-entropy loss

$$L(\boldsymbol{f}) = -\sum_{i=1}^{m} \sum_{l=1}^{K} z_i^l \log f^l(\boldsymbol{x}_i)$$

computation of $\nabla L(\boldsymbol{x}_i)$ is trivial for both losses

Geometric Regularization Term

Volume penalty

$$G(\boldsymbol{f}) = \int_{graph(\boldsymbol{f})} dvol = \int_{graph(\boldsymbol{f})} \sqrt{\det(\boldsymbol{g})}\, dx^1 \cdots dx^N$$

$\boldsymbol{g}$ is the Riemanian metric on $graph(\boldsymbol{f})$ induced from the standard dot product on $\mathbb{R}^{N+K}$

Geometric Regularization Term

Gradient vector field of $G(\boldsymbol{f})$

$$-\nabla G = \mathrm{TrII}^K$$

$$= (g^{-1})^{ij}(f_{ji}^1 - (g^{-1})^{rs} f_{rs}^l f_i^l f_j^1, \cdots, f_{ji}^K - (g^{-1})^{rs} f_{rs}^l f_i^l f_j^K)$$

where f_i^l, f_{ij}^l denote partial derivatives of f^l

given 1st and 2nd partial derivatives
the computation involves only matrix manipulations

Example Formulation: RBFs

Represent $\boldsymbol{f}$ as "softmax" output of RBFs

$$f^j = \frac{\exp(h^j)}{\sum_{l=1}^{K} \exp(h^l)}, \quad h^j = \sum_{i=1}^{m} a_i^j \varphi_i(\boldsymbol{x}), \text{ for } j = 1, \cdots, K$$

where $\varphi_i(\boldsymbol{x}) = e^{-\frac{1}{c}\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}$ is the RBF centered at $\boldsymbol{x}_i$

Gradient update for $\mathrm{A} = (a_i^l)$

$$A \leftarrow A - \tau M^{-1}[\nabla\mathcal{P}_{\boldsymbol{h}}(\boldsymbol{x}_1), \cdots, \nabla\mathcal{P}_{\boldsymbol{h}}(\boldsymbol{x}_m)]^T,$$

where $\nabla\mathcal{P}_{\boldsymbol{h}}(\boldsymbol{x}_i) = \left[\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{h}}\right]_{\boldsymbol{x}_i}^T \nabla\mathcal{P}_{\boldsymbol{f}}(\boldsymbol{x}_i), \quad M_{ij} = \varphi_j(\boldsymbol{x}_i)$

Experiments – RBF Representation

Datasets from UCI Repository

- Four binary and four multiclass datasets
- Following the choice/setup of previous papers

Comparing with two groups of classifiers

- RBF + functional norm regularization: RBN, SVM, KLR
- RBF + existing geometric regularization: LLS, GLS, EE

UCI Datasets – Interesting pairs

KLR vs. Ours-CE

- same: RBF-based, cross-entropy loss
- diff regularizer: RKHS norm vs geometry on class probability

GLS vs. Ours-CE/Ours-Q

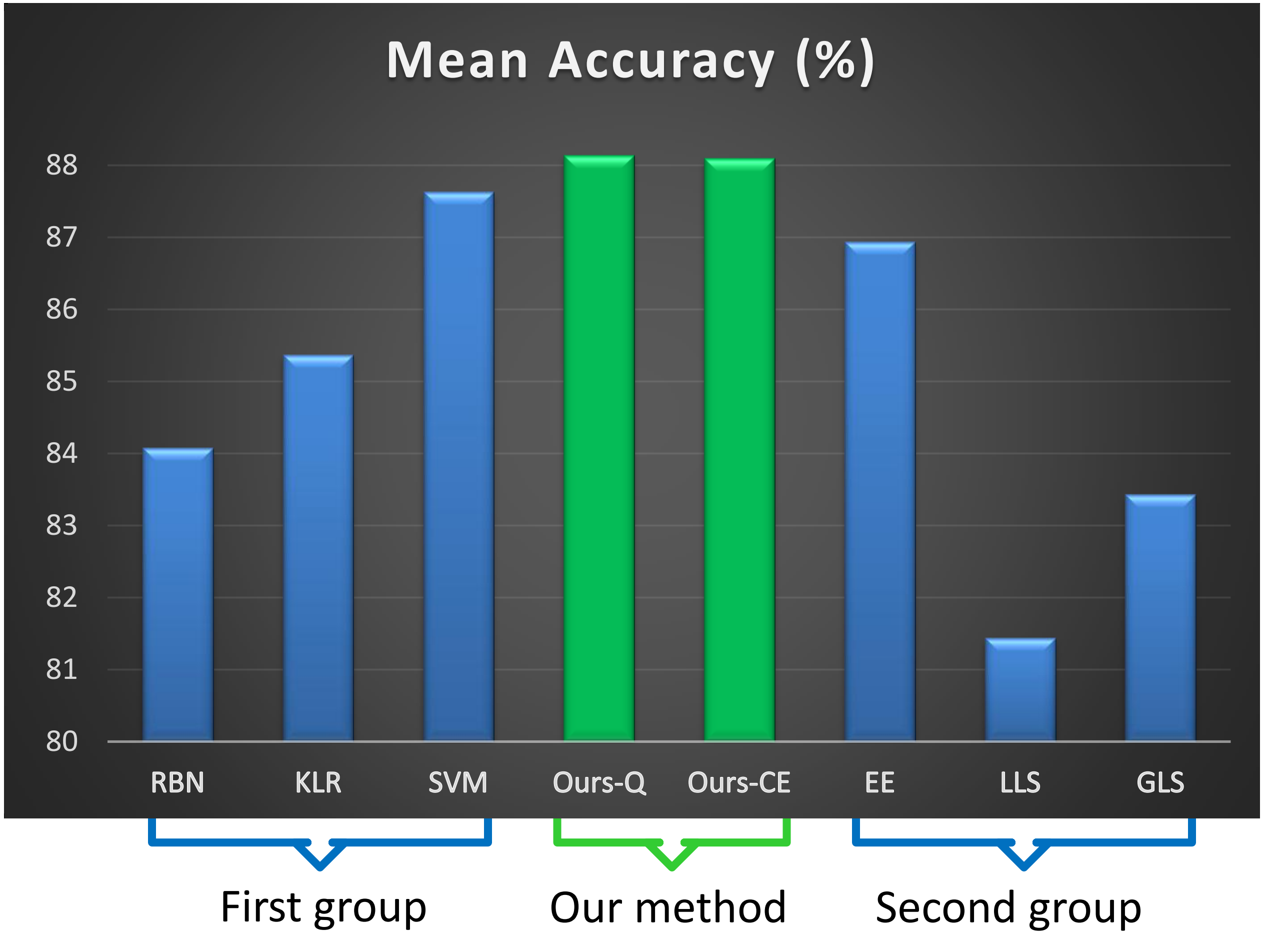
- same: RBF-based, volume based geometric regularizer
- diff geometry: on decision boundary vs on class probability

EE vs. Ours-Q

- same: RBF-based, quadratic loss
- diff geometric regularizer:

  sophisticated on decision boundary vs on class probability

Results on UCI Datasets

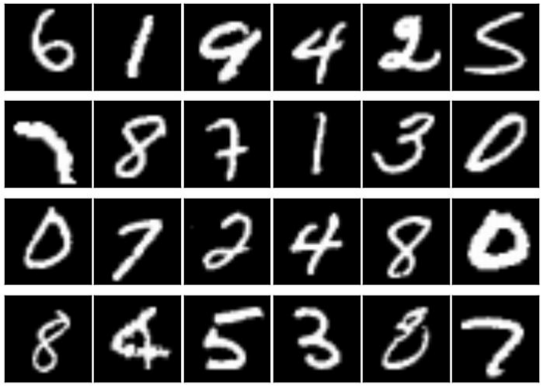


Experiments – RBF Representation

Real-world datasets – comparing with baseline

- Flickr Material Database (4096 dimensional feature)
- MNIST handwritten digits (60,000 samples)

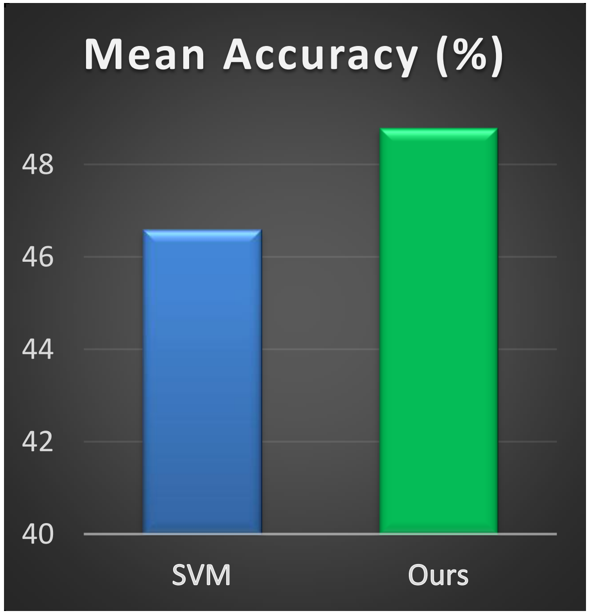
Flickr Material Database



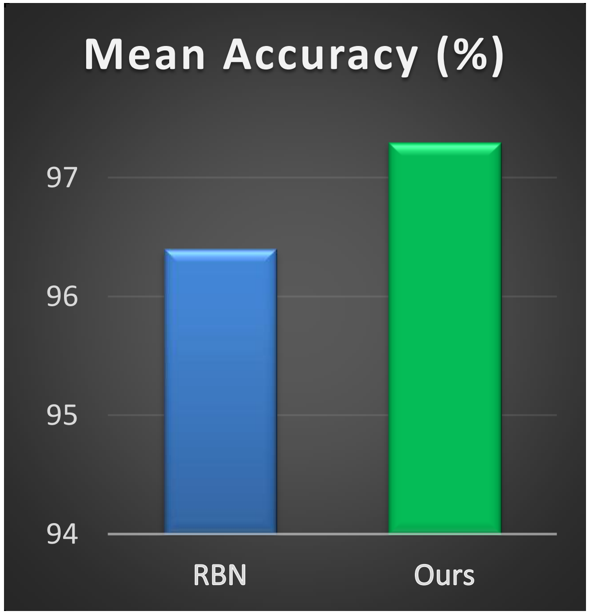
MNIST handwritten digits

Results on Real-world Datasets

Flickr Material Database



MNIST handwritten digits



Summary

- New geometric perspective on overfitting
- First regularization approach that exploits the geometry of a class probability estimator for classification
- Unified framework for both binary and multiclass cases
- Compares favorably to existing regularization methods

Collaborators

Steven Rosenberg
Mathematics, BU

Stan Sclaroff
Computer Science, BU

Zheng Wu
Mathworks Inc.