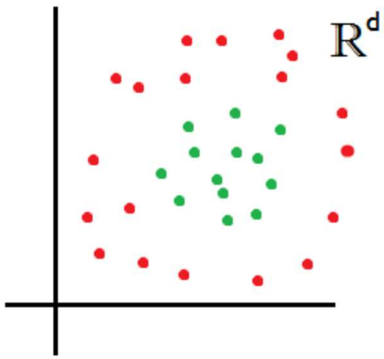
Learning with Similarity Functions

Prateek Jain and Purushottam Kar

Machine Learning and Optimization Group
Microsoft Research Lab India

Target App : Binary Classification

- True (unknown) classifier $f^*: \mathcal{X} \to \{-1,+1\}$
  - $\mathcal{X}$ can be the set of all 50 x 50 images
  - f^* : dichotomy b/w face and non-face images
  - Assume a distribution on the domain $\mathcal{D}$



- Goal : discover another classifier $h : \mathcal{X} \to \{-1,+1\}$
  - h agrees with f^* on "most" points $\Pr_{x\sim\mathcal{D}}[h(x) \neq f^*(x)] \leq \epsilon$
  - h is said to be ϵ-close to the true classifier
- Supervised learning
  - Get a glimpse of f^* in action via a training set
  - $x_1, x_2, \ldots, x_n \sim \mathcal{D}$ i.i.d. samples along with true responses $f(x_i)$
  - Use some interpolation technique to construct a hypothesis *

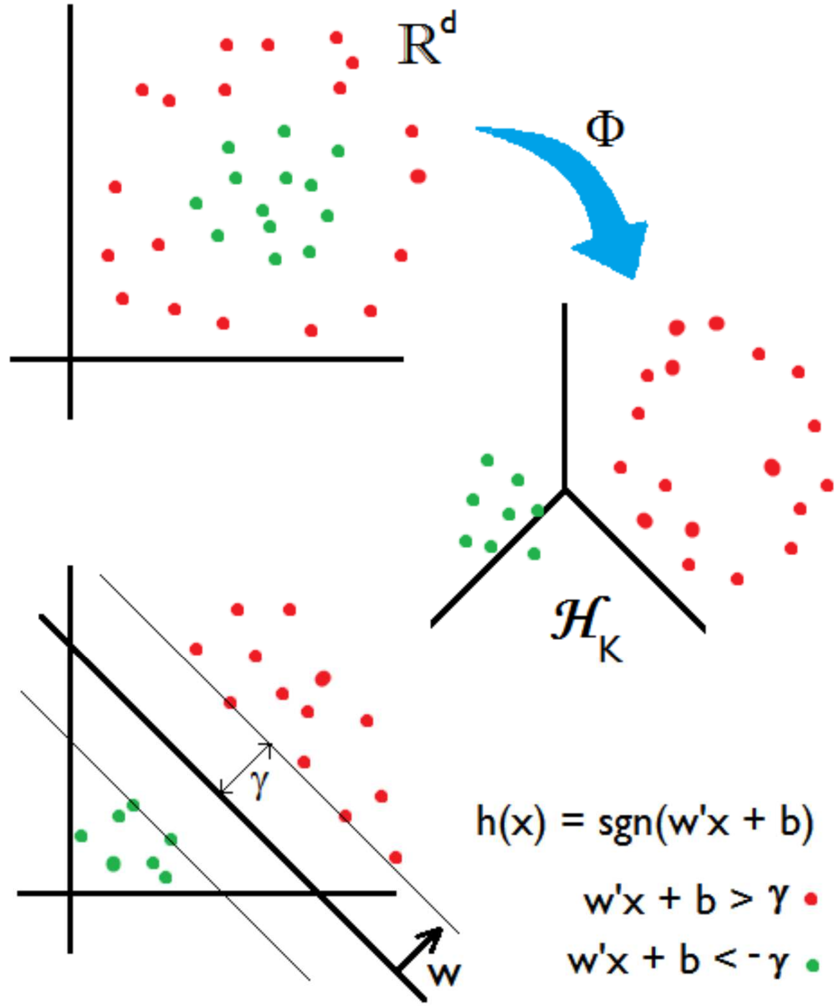
* Ensure this for most samples to claim the PAC-man title !!

Learning from training data

- We have some data for which true labels are known
  - $\{(x_1, f(x_1)), (x_2, f(x_2)), \ldots, (x_n, f(x_n))\}$
  - For simplicity, let $y_i := f(x_i)$
- Set up an interpolation scheme to generalize
  - Nearest neighbor $h(x) = f\left(\underset{x_i}{\operatorname{argmin}}\, d(x, x_i)\right)$
    - Need some distance measure $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$
  - Smoother interpolation $h(x) = \operatorname{sgn}(\sum_{i=1}^{n} s(x, x_i) y_i)$
    - Need some similarity measure $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$
  - Sparse interpolation $h(x) = \operatorname{sgn}(\sum_{i=1}^{n} \alpha_i s(x, x_i) y_i)$
  - Wait … are you trying to sneak in kernel learning ???
  - … well yeah !

Learning with kernels

- Support Vector Machines
  - Learn a hyper plane classifier in a vector space
  - Maximize margin : the larger the better
- Kernel trick
  - Allow SVMs to work in high dimensional spaces
  - Introduce a (large) margin
  - $\Phi : \mathcal{X} \rightarrow \mathcal{H}_K$
  - Learn a linear classifier in $\mathcal{H}_K$



Learning with kernels

Primal view

- Explicit form
- $$\min_{w \in \mathcal{H}_K, b} \left\{ \frac{1}{2} \|w\|^2 + \sum_{i=1}^{n} \ell_i \right\}$$
  $$y_i(\langle w, \Phi(x_i) \rangle + b) \geq 1 - \ell_i$$
  $$\ell_i \geq 0$$

Explicit ⇔ Implicit

- $w = \sum_{i=i}^{n} \alpha_i y_i \Phi(x_i)$
- $h(x) = \text{sgn}(\langle w, \Phi(x) \rangle + b)$
- $\text{sgn}(\sum_{i=1}^{n} \alpha_i y_i K(x, x_i) + b)$

Dual view

- Implicit form *
- $$\max_{\alpha} \{ \alpha^\top \mathbb{1} - \alpha^\top K \alpha \}$$
  $$0 \leq \alpha_i \leq 1$$
  $$\sum y_i \alpha_i = 0$$
  $$K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle$$

- K : $\mathcal{X}$ x $\mathcal{X}$ → $\mathbb{R}$ must be PSD
- Must introduce a margin
- Is all this really necessary ?

* To "b" or not to "b" : that really is a big question ☺

Redefining kernel learning

Geometric view (implicit)

- Embedding $\Phi : \mathcal{X} \to \mathcal{H}_K$
- Classifier $w \in \mathcal{H}_K$
- A kernel K is (ϵ, γ)-Kgood for a problem $(f^*, \mathfrak{D})$ if there exists $w \in \mathcal{H}_K$ such that most points respect a margin
- Suppose $f^*(x) = y$

$$\mathrm{GOOD}_\gamma(x) \coloneqq y\langle w, \Phi(x) + b\rangle \geq \gamma$$

$$\Pr_{x \sim \mathcal{D}}\left[\mathrm{GOOD}_\gamma(x)\right] \geq 1 - \epsilon$$

- Kernel introduces a margin

Functional view (explicit)*

- Embedding $x \mapsto \left(K(x, x_1), \ldots K(x, x_n)\right)$
- Classifier $\alpha \in \mathbb{R}^n$
- A kernel K is (ϵ, γ)-Sgood for a problem if most points are (in weighted sense), closer to points of same label
- Suppose $f^*(x) = y$

$$\mathrm{A}_+(x) \coloneqq \mathop{\mathrm{E}}_{x' \sim \mathcal{D}^+}[w(x')K(x, x')]$$

$$\mathrm{A}_-(x) \coloneqq \mathop{\mathrm{E}}_{x' \sim \mathcal{D}^-}[w(x')K(x, x')]$$

$$\mathrm{GOOD}_\gamma(x) \coloneqq y(\mathrm{A}_+(x) - \mathrm{A}_-(x)) \geq \gamma$$

$$\Pr_{x \sim \mathcal{D}}\left[\mathrm{GOOD}_\gamma(x)\right] \geq 1 - \epsilon$$

- Kernel introduces explicit separation

* Balcan and Blum, "A Theory of Learning with Similarity Functions, ICML 2006"

Learning with kernels

- The functional view makes no reference to any explicit embedding nor does it require the kernel to be PSD
  - Proposed as an alternative "goodness" criterion for kernel learning by [Balcan-Blum, '06]
- Sanity checks for this new "goodness" criterion
  - Utility (anything you call good should be useful as well) [Balcan-Blum, '06]
    - Every (ϵ, γ)-Sgood kernel can be used to learn a classifier that is $(\epsilon + \epsilon_1)$-close to the true classifier for any $\epsilon_1 > 0$
  - Admissibility (everything that was good should continue to remain good) [Srebro, '07]
    - Every (ϵ, γ)-Kgood kernel is $\left(\epsilon + \epsilon_1, \frac{1}{4}\epsilon_1\gamma^2\right)$-Sgood for any $\epsilon_1 > 0$

Learning with Similarity functions

- Several domains have natural notions of (non-PSD) similarities
  - Earth Mover's distance : images, distributions
  - Overlap distance : co-authorship graphs, texts (bag-of-words)
- Using Sgood-ness to extend kernel learning techniques to (non-PSD) similarity functions ?

1. Select random "landmark points" $\mathcal{L} = \{x_1^l, x_2^l, \dots, x_d^l\} \sim \mathcal{D}^d$
2. Construct an embedding $\Psi_{\mathcal{L}}(x) = \left(K(x, x_1^l), \dots, K(x, x_d^l)\right)$
3. Select random training points $\mathcal{T} = \{x_1, x_2, \dots, x_n\} \sim \mathcal{D}^n$
4. Learn a vector $\alpha \in \mathbb{R}^d$ using training points
5. Output classifier $h(x) = \operatorname{sgn}(\langle \alpha, \Psi_{\mathcal{L}}(x) \rangle) = \sum_{i=1}^{d} \alpha_i K(x, x_i^l)$

- Classifier of same form as that in SVM !
  - In fact, one can use the SVM algorithm on $\mathbb{R}^d$ to learn α

Learning with kernels vs. similarities

PSD kernel learning

- Implicit form (with "b")
- $\max_{\alpha}\{\alpha^{\top}\mathbb{1} - \alpha^{\top}K\alpha\}$

$$\sum y_i \alpha_i = 0$$
$$0 \leq \alpha_i \leq 1$$

- $h(x) = \mathrm{sgn}(\sum_{i=1}^{n} \alpha_i y_i K(x, x_i) + b)$
- Data oblivious embedding Φ
- Sparsity inducing regularization on α
- Need just a training set

Similarity learning

- Explicit form
- $\min_{\alpha,b}\left\{\frac{1}{2}\|\alpha\|^2 + \sum_{i=1}^{n} \ell_i\right\}$

$$y_i(\langle \alpha, \Psi_{\mathcal{L}}(x_i)\rangle + b) \geq 1 - \ell_i$$
$$\ell_i \geq 0$$

- $h(x) = \mathrm{sgn}\left(\sum_{i=1}^{n} \alpha_i K\left(x, x_i^l\right) + b\right)$
- Data dependent embedding $\Psi_{\mathcal{L}}$
- Usually get non sparse α
- Need separate landmark and training sets

Theoretical Guarantees

- If a kernel/similarity is (ϵ, γ)-Sgood then most likely the landmarked space has a large margin classifier in it
  - There exists α such that $\Pr_{x \sim \mathcal{D}}\left[y\langle\alpha, \Psi_{\mathcal{L}}(x)\rangle \leq \frac{\gamma}{4}\right] \leq \epsilon + \epsilon_1$*
  - We can learn this large margin classifier using training data
  - We used an Sgood (non)PSD kernel to define a Kgood PSD kernel
- How much data required ?
  - About $\mathcal{O}\left(\frac{1}{\gamma^2 \epsilon_1^2} \log \frac{1}{\delta}\right)$ landmark points and a similar number of training points required to obtain a classifier that is $(\epsilon + \epsilon_1)$-close to the true classifier with a confidence of $(1 - \delta)$

* Note that this means that the kernel $K_{\mathcal{L}}(x, x') = \langle\Psi_{\mathcal{L}}(x), \Psi_{\mathcal{L}}(x')\rangle$ is $\left(\epsilon + \epsilon_1, \frac{\gamma}{4}\right)$-Kgood

A “brief” overview of the guarantees

Task	Suitability (Sgood)	Utility	Sample Complexity	Admissibility Kgood → Sgood
Classification [Balcan-Blum ‘06] [Srebro ‘07]	(ϵ, γ)	$(\epsilon + \epsilon_1)$ Misclassification rate	$\mathcal{O}\left(\frac{1}{\gamma^2 \epsilon_1^2}\right)$(U+L)	$(\epsilon, \gamma) \Rightarrow \left(\epsilon + \epsilon_1, \Theta(\epsilon_1 \gamma^2)\right)$
Regression * [Jain-K.‘12]	(ϵ, B)	$(B\epsilon + \epsilon_1)$ Mean squared error	$\mathcal{O}\left(\frac{B^4}{\epsilon_1^2}\right)$(U+L)	$(\epsilon, \gamma) \Rightarrow \left(\epsilon + \epsilon_1, \Theta\left(\frac{1}{\epsilon_1 \gamma^2}\right)\right)$
Ordinal Regression * [Jain-K.‘12]	(ϵ, B, Δ)	$\psi_\Delta(\epsilon) + \epsilon_1$ Ordinal Regression Error	$\mathcal{O}\left(\frac{B^2}{\Delta^2 \epsilon_1^2}\right)$(U+L)	$(\epsilon, \gamma, \Delta) \Rightarrow \left(\gamma_1 \epsilon + \epsilon_1, \Theta\left(\frac{\gamma_1^2}{\epsilon_1 \gamma^2}\right), \gamma_1 \Delta\right)$
m-Ranking * [Jain-K.‘12]	(ϵ, B)	$\mathcal{O}\left(\sqrt{\frac{m\epsilon}{\log m}} + \epsilon_1\right)$ NDCG loss	$\mathcal{O}\left(\frac{B^6 m^8}{\epsilon_1^4 \log^2 m}\right)$ U + $\mathcal{O}\left(\frac{B^6 m^4}{\epsilon_1^4 \log^2 m}\right)$ L	$(\epsilon, \gamma) \Rightarrow \left(\epsilon + \epsilon_1, \mathcal{O}\left(\sqrt{\frac{m^3}{\epsilon_1^3 \gamma^6}}\right)\right)$

* Notion of suitability (K/S-goodness) a bit different for non classification learning problems

Final words

- Notion of suitability amenable to efficient training algos
  - Suitability criterion with convex surrogate loss functions
  - [Balcan-Blum, '06], [Jain-K., '11]
- Double dipping : can we reuse training set for landmarks ?
  - Yes … via uniform convergence guarantees for data dependent hypothesis spaces * [Srebro et al, '08], [Jain-K., '12]
- Other supervised learning formulations
  - Modified suitability criteria for supervised learning
  - Regression, ordinal regression, ranking
  - Sparse regression (regression with sparse α) [Jain-K. ,'12]
  - Utility, (tight) admissibility results [Jain-K. ,'12]

* Sorry … could not resist using a complicated-looking phrase !