Locally-Linear Learning Machines (L3M)

Authors : Joseph Wang and Venkatesh Saligrama

Date : Apr 28th 2017

Analysis by :
Dundi Vinayak Doddipatla, Garlapati B M K Rao, Lalit Mohan (#201350896)

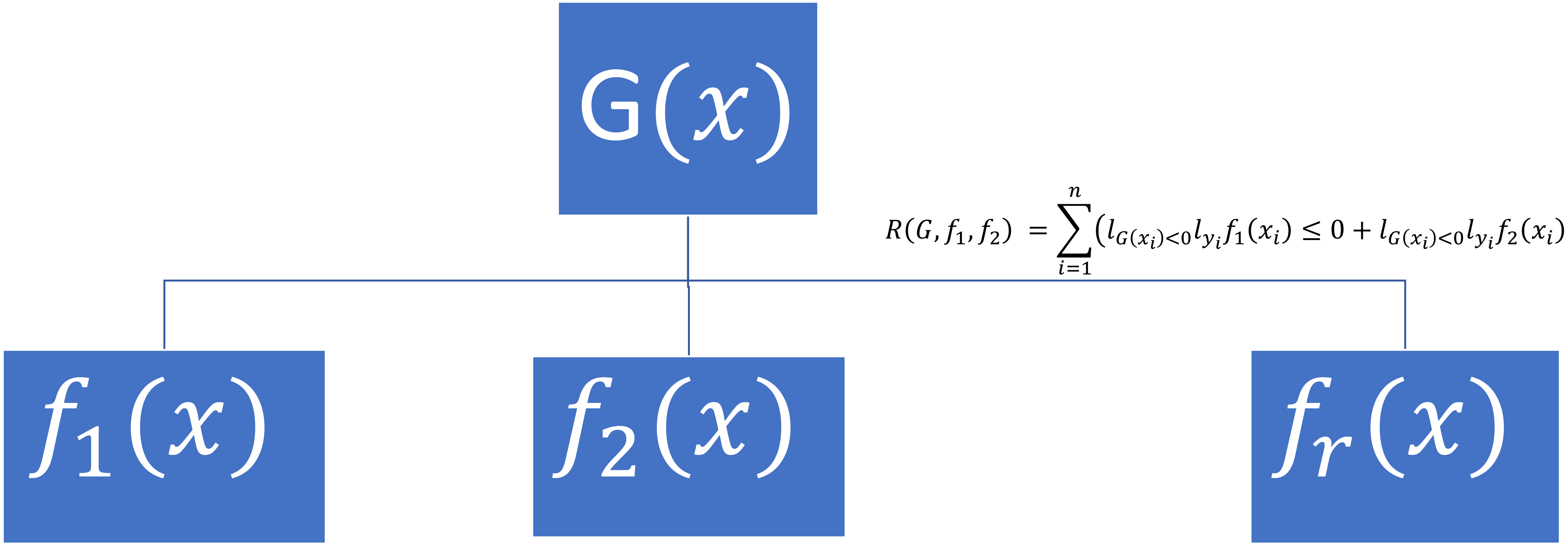
Agenda

- Abstract of the paper – Understanding
- Introduction and Approach
- Formulae
- Algorithm
- Datasets
- Evaluation
- Questions and Threats to Validity
- Plan

Abstract

- Formulate a global Convex risk function to jointly learn linear feature space partitions and region specific linear classifiers.
- Features
  - Discriminant power similar to Kernel SVM and Adaboost
  - Tight control on generalization error
  - Low training time cost due to online training
  - Low test time due to local linearity
  - Low VC dimension and predictable generalization performance
- Tight convex surrogate by embedding empirical risk loss as an OP and then convexifying this resulting problems.

Introduction and approach



$$R(G, f_1, f_2) \ = \sum_{i=1}^{n} \left(l_{G(x_i)<0} l_{y_i} f_1(x_i) \leq 0 + l_{G(x_i)<0} l_{y_i} f_2(x_i) \leq 0 \right)$$

Alternative maximization $l_{G(x_i)<0} l_{y_i} f_1(x_i) \leq \max(1 - G(x) - f(x)y, 0)$

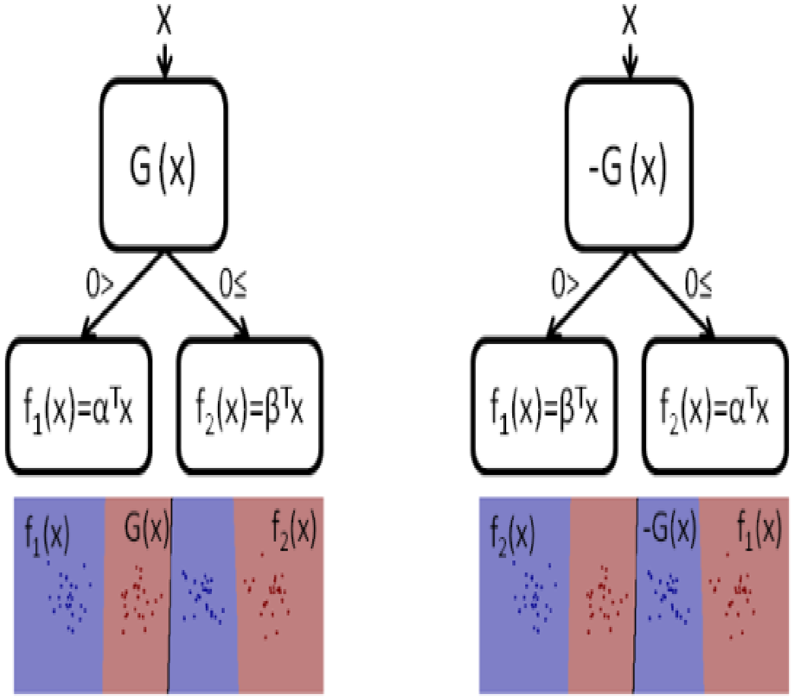
Goal is to learn *G(x)*, *f1(x)* and *f2(x)* jointly that minimizes the empirical loss

Proposition 1.1

- $R(G, f_1, f_2) = \sum_{i=1}^{n}[\ell_{G\,(x_i)<0}\ell_{y_i\,f1(x_i)\leq 0} + \ell_{G\,(x_i)\geq 0}\ell_{y_i\,f2(x_i)\leq 0}]$

$Consider\ f_1 = f_1^*$, $f_2 = f_2^*$ and $G = G*$ minimizes empirical error

$f_1 = f_2^*$, $f_2 = f_1^*$ and $G = -G*$ induces decision boundaries and identical loss



This is not an optimal solution with respect to the indicator loss function. Symmetry of the loss function around the point G = 0 presents the fundamental limitation for all convex relaxations.

Break the symmetry.

Accomplish this choosing a random point, x_k and containing G(x_k) >= β

Convex Parameterization for Binary Partitioning & Binary classification

- Empirical Loss can be expressed as

$$F(x) = \left(l_{G(x)<0}f_1(x) + l_{G(x)\geq 0}f_2(x)\right)$$

G(x) partitions the feature space into 2 regions and in each a local classifier $f_1(x)\ or f_2(x)$ predicts a label for the observation

- Proposition 2.1 :

$$l_{a<0}l_{b<0} = \ min_{\lambda\epsilon|0,1|}\lambda l_{a<0} + (1-\lambda)l_{b<0}$$

Product of indicators to be separated into a linear combination of indicators . Change product of indicators to linear combination (add)

- $R(G, f_1, f_2\ ) = \sum_{i=1}^{n}\ min_{\lambda,\varepsilon[0,1]}\ [\lambda_1\ell_{G\ (x_i)<0} + (1-\lambda_1)\ell_{y_i\ f1(x_i)\leq 0} + \lambda_2\ell_{G\ (x_i)\geq 0} + (1-\lambda_2)\ell_{y_i\ f2(x_i)\leq 0}]$

- Proposition 2.2 :

$$l_{F(x_i)\neq y_i} = \ \left(1 - l_{F(x_i)=y_i}\right)$$

- Proposition 2.3 :

$$R(G, f_1, f_2) = \sum_{i=1}^{n} \max[\left(l_{G(x_i)\geq 0} + l_{y_i f_2(x_i)\leq 0}, l_{y_i f_2(x_i)\leq 0} + l_{G(x_i)<0}\right)]\ - 1$$

$\lambda_1\ and \lambda_2\ \ may\ not\ be\ unique$
$Optimal\ solution\ at\ \lambda_1 = 1 - \ \lambda_2.$
$\lambda\ \ \ = \lambda_1.and\ \ \ \lambda\ \ \ = 1 - \ \lambda_2$

Convex surrogate

- Convexity is preserved in Preposition 2.3 equation unlike equation in Preposition 1.1
- Convex upper-bounding surrogate function by replacing indicator with hinge losses. This results in tightest convex relaxation

$$\hat{R}(G, f_1, f_2) = \sum_{i=1}^{n} \max\left[(1 - y_i f_1(x_i))_+ + (1 - G(x_i))_+, (1 + G(x_i))_+ + (1 - y_i f_2(x_i))_+\right] - 1$$

- Preposition 2.4
  - Upper bound $\max\left[\mathbb{1}_{a\geq 0} + \mathbb{1}_{b\leq 0}, \mathbb{1}_{c\leq 0} + \mathbb{1}_{d\leq 0}\right] - 1$

$$\min_{G, f_1, f_2, G(x_k)\geq\beta} \sum_{i=1}^{n} \max\left[(1-y_i f_1(x_i))_+ + (1-G(x_i))_+, (1+G(x_i))_+ + (1-y_i f_2(x_i))_+\right] + \lambda\left(\|f_1\|_2^2 + \|f_2\|_2^2\right)$$

L3M for Multiple Regions and Multiclass Data

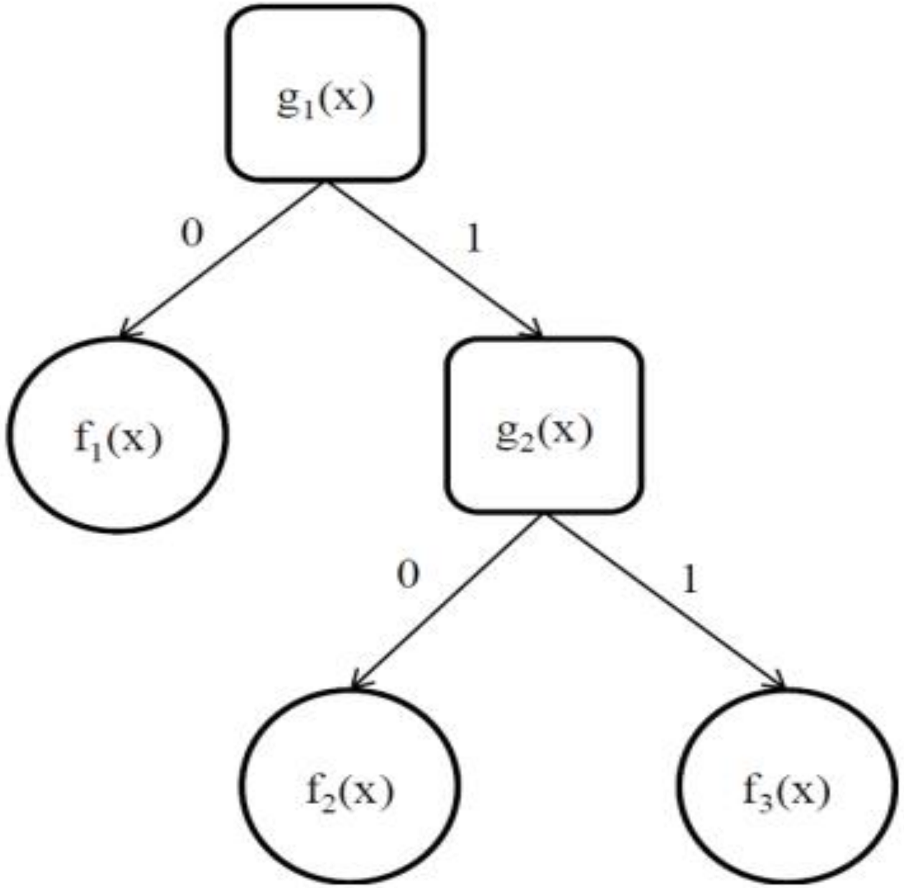
- From Proposition 2.3

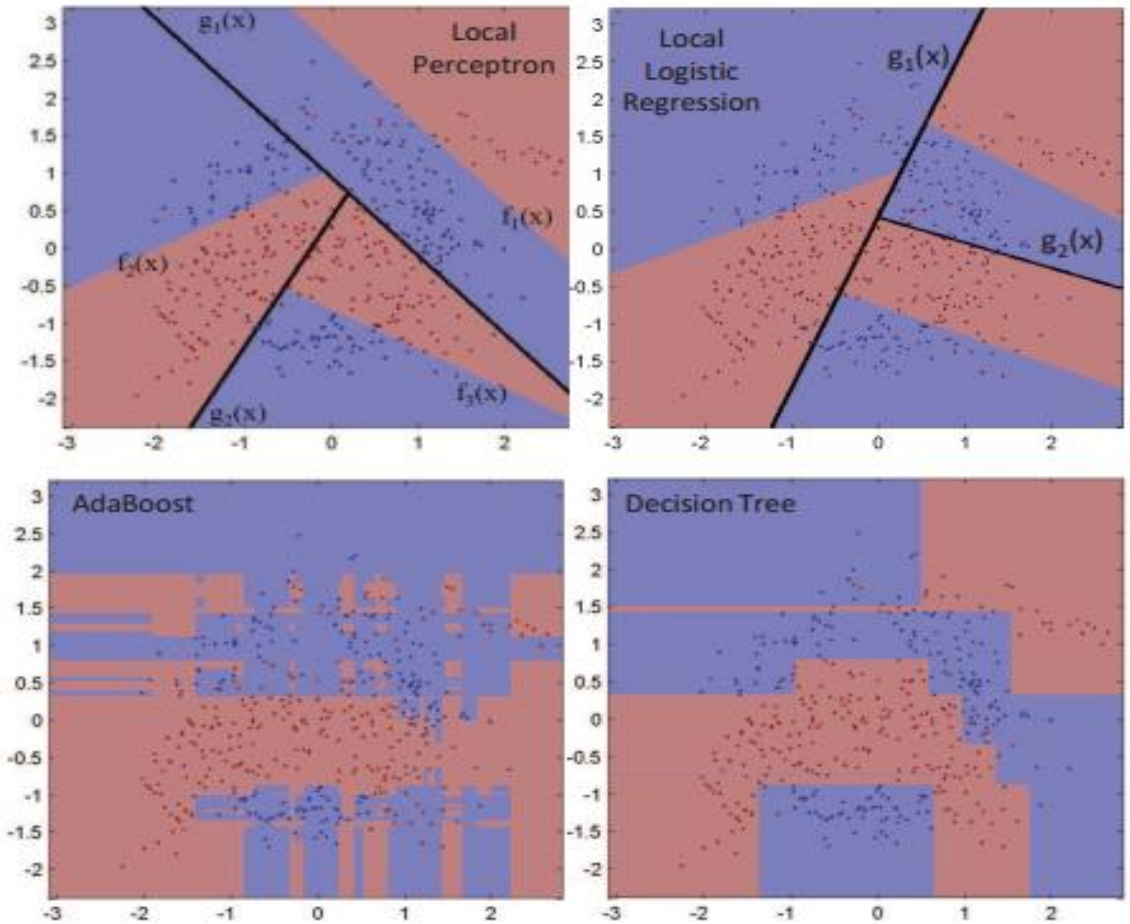
$$R(G, f_1, \ldots, f_r) = \sum_{i=1}^{n} \max_{k \in \{1,\ldots,r\}} \left[\mathbb{1}_{f_k(x_i) \neq y_i} + \mathbb{1}_{G(x_i)=k} - 1\right]$$

$$\phi(G, k, x_i) = \max\left[(1 + g_k(x_i))_+ , \max_{j \neq k} (1 - g_j(x_i))_+\right]$$

Maximum hinge-loss over the one vs all classifiers

Region and local Classification





Theorem 4. 1

- VC-dimension of local linear classifier with 3 regions can be bounded

$$2\left(\frac{(r-1)^2+2}{2}\right)\log\left(e\left(\frac{(r-1)^2+2}{2}\right)\right)(d+1).$$

r to minimize a high-probability bound on the generalization error

Computational Complexity → $O(dr+d)$, multi-class – $O(dr+dc)$

Space Partitioning G(x)

Algorithm 1 Space Partitioning Classifier

Input: Training data, $\{(x_i, y_i)\}_{i=1}^{n}$, number of classification regions, r
Output: Composite function, $F(\cdot)$
Initialize: Assign points randomly to r regions
while F not converged **do**
  for $j = 1, 2, \ldots, r$ **do**
    Train region functions $f_j(x)$ to optimize empirical loss of Eq. (3.7).
  end for
  for $k = r-1, r-2, \ldots, 2, 1$ **do**
    Train reject classifier $g_k(x)$ to optimize empirical loss of Eq. (3.8).
  end for
end while

Algorithm – Online Training of L3Ms

Input: Observation and label, x_t, y_t, current partitioning classifier, α, and local classifiers β_1, β_2

Output: Updated partitioning classifier, α, updated local classifiers β_1, β_2

1. Find active region

$$r_t = \begin{cases} 1 & \text{if } \log(1+e^{\alpha^T x_t}) + \log(1+e^{-y_t \beta_1^T x_t}) > \\ & \quad \log(1+e^{-\alpha^T x_t}) + \log(1+e^{-y_t \beta_2^T x_t}) \\ 2 & \text{otherwise} \end{cases}$$

2. Calculate the subgradient for the partitioning classification functions:

$$\nabla\alpha = \begin{cases} \frac{-x_t}{1+e^{-\alpha^T x_t}} & \text{if } r = 1 \\ \frac{x_t}{1+e^{\alpha^T x_t}} & \text{if } r = 2 \end{cases}, \quad \nabla\beta_1 = \begin{cases} \frac{-y_t x_t}{1+e^{y_t \beta_1^T x_t}} & \text{if } r = 1 \\ 0 & \text{if } r = 2 \end{cases}, \quad \nabla\beta_2 = \begin{cases} 0 & \text{if } r = 1 \\ \frac{-y_t x_t}{1+e^{y_t \beta_2^T x_t}} & \text{if } r = 2 \end{cases}$$

3. Return updated functions:

$$\alpha = \alpha - \frac{\nabla\alpha}{\sqrt{t}}, \quad \beta_1 = \beta_1 - \frac{\nabla\beta_1}{\sqrt{t}}, \quad \beta_2 = \beta_2 - \frac{\nabla\beta_2}{\sqrt{t}}$$

Datasets

Dataset	Dimension	Classes	Training Set	Test Set
Banana	2	2	400	4900
DNA	180	3	2000	1186
Landsat	36	7	4435	2000
Vowel	10	11	528	462
Optdigit	64	10	3823	1797
Pendigit	16	10	7494	3498
Image Seg.	19	7	210	2100

Evaluation

- Performed action for SVM, Adaboost and GDI

Algorithm	Banana	DNA	Landsat	Vowel	Optdigit	Pendigit	Image Segmentation
One vs All Linear SVM	39.55%	7.08%	17.90%	59.09%	7.63%	10.92%	8.24%
One vs All RBF SVM	11.86%	5.48%	9.70%	37.23%	2.34%	1.86%	11.30%
One vs All AdaBoost	32.98%	8.35%	16.10%	69.70%	12.24%	11.29%	10.38%
GDI Tree	14.33%	9.36%	14.45%	56.93%	14.58%	8.78%	9.71%
MDA	20.45%	12.14%	36.45%	67.32%	9.79%	7.75%	15.43%
L3M	11.84%	5.31%	17.50%	40.69%	7.12%	10.52%	10.76%

Our results using EM (for region separation) and Logistic Regression (for classification)

Algorithm	Banana	DNA	Landsat	Vowel	Optdigit	Pendigit	ImageSeg
SVM – Poly	44.9% 0.8 sec	49.15% 50.3 sec	44.9%	80.51%	85.14%	19.01% 35.01 sec	85.85% 0.52 sec
SVM – RBF Gaussian	44.9% 1.145 sec	33.47%	14.9% 18.85 sec	54.76%	2.44%	4.31% 14.91 sec	16.52% 2.08 sec
Adaboost	30.9% 0.615 sec	21.4% 1.19 sec	58.25% 0.63 sec	87.22% 0.36 sec	80.63% 0.39 sec	79.33% 0.52 sec	71.76% 035 sec
Decision Tree	16.34% 0.25 sec	7.5% 1.509 sec	14.65% 1.92 sec	64.71% 0.30 sec	14.2% 0.56 sec	7.94% 1.71 sec	9.23% 0.28 sec
Our Mixture Model	1.02% 4.32 sec	21.75% 48.99 sec	4.7% 17.52 sec	67.97% 2.6 sec	6.51% 25.69 sec	1.17% 10.53 sec	7.23% 4.79 sec

Questions?