Multiresolution Methods for Large-scale Learning @ NIPS 2015

Ilya Safro

Clemson University

Multigrid-inspired Methods for Large-scale Networks
- Fast Response to Infection Spread (with S. Leyffer)
- Support Vector Machines (with T. Razzaghi)
- Network Generation (with A. Gutfraind and L.A. Meyers)
Algebraic Multigrid in 3 Slides: Relaxation, Smoothness

**Observation**

A suitable relaxation can reduce the information content of the error (by smoothing it), and quickly make it approximable by far fewer variables (which are related to the smooth error modes).

Example: Solve $Ax=b$ with initial random guess $x^{(0)}$ ($A$ is s.p.d.) by stationary iterative relaxation (such as Gauss-Seidel)

$$x^{(k+1)} = T x^{(k)} + v$$

---

<table>
<thead>
<tr>
<th>Initial error</th>
<th>After 5 iterations</th>
<th>After 10 iterations</th>
<th>After 500 iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error = $x^* - x^{(k)}$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Brandt, McCormick, Rudge, “Algebraic Multigrid (AMG) for automatic multi-grid solution with application to geodetic computations”, 1982

- Given: $A \in \mathbb{R}^{n \times n}$ positive definite, symmetric.
- Goal: solve $Ax = b$.
- Claim: If $A$ is positive definite, then

$$x^* \text{ minimizes } P(x) = \frac{1}{2} x^T Ax - x^T b \iff Ax^* = b.$$ 

- $\tilde{x}$ - current approximation
- $e(rror) = x^* - \tilde{x}$
- $b - A\tilde{x} = r(esidual) = A(x^* - \tilde{x}) = Ae$
Algebraic Multigrid in 3 Slides: Coarsening, Correction Scheme

Solve $Ae = r$, where $e(error) = x^* - \tilde{x}$ and $r(residual) = b - A\tilde{x}$

$$\min \frac{1}{2} e^T Ae - e^T r \iff$$

$$\min \frac{1}{2} (\tilde{e} + \uparrow^c c^e)^T A (\tilde{e} + \uparrow^c c^e) - (\tilde{e} + \uparrow^c c^e)^T r \iff \ldots \iff$$

$$\min \frac{1}{2} (c^e)^T \left[ (\uparrow^c c^e)^T A \uparrow^c c \right] c^e - (c^e)^T (\uparrow^c c^e)^T (r - A\tilde{e}) \iff$$

$$\min \frac{1}{2} (c^e)^T A^e e^e - (c^e)^T r^e$$

- $\tilde{e}$ - initial fine level error
- $c^e$ - coarse level error
- $\uparrow^f c$ - coarse-to-fine interpolation operator
Multigrid Framework

Examples:
• VLSI placement
• Graph partitioning
• Eigensolvers
• Clustering
• Linear arrangement
• Community detection
• Modularity
• Traveling salesman
• Visualization
• Compression-friendly ordering
• Coloring
• Spectral problems
Algebraic Distance

The iterators $H_*$ for $x^{(k+1)} = H_*x^{(k)}$ are defined as

$$
H_{Gauss} = (D - L)^{-1}U \quad H_{SOR} = (D/\omega - L)^{-1}((1/\omega - 1)D + U)
$$

$$
H_{Jacobi} = D^{-1}(L + U) \quad H_{Jacobi} = (D/\omega)^{-1}((1/\omega - 1)D + L + U)
$$

Extended $p$-normed algebraic distance between nodes $i$ and $j$ after $k$ iterations $x^{(k+1)} = H_*x^{(k)}$ on $R$ random initializations $x^{(0,r)}$

$$
\rho_{ij}^{(k)} := \left( \sum_{r=1}^{R} \left| x_i^{(k,r)} - x_j^{(k,r)} \right|^p \right)^{1/p}
$$

Slow convergence but very fast stabilization. Extendible to hypergraphs. See [1]

Weighted Aggregation of Graphs (inspired by Algebraic Multigrid)

Examples

• S, Ron, Brandt “Graph minimum linear arrangement by multilevel weighted edge contractions”, 2006
• Ron, S, Brandt “Relaxation-based coarsening and multiscale graph organization”, 2011
• S, Sanders, Schultz “Advanced coarsening schemes for graph partitioning”, 2013
Coarse nodes

- Choose a dominating set \( C \subset V \) s.t. all other nodes in \( F = V \setminus C \) are strongly coupled to \( C \)

- Strongly coupled =
  - (normalized) algebraic distance
    [Chen, Safro 2012], [Livne, Brandt 2013], ...
  - random walk approaches (commute, hitting times)
    [Fouss et al. 2007], [Zhao et al. 2013], ...
  - interpretations of the diffusion, wavelets and Brownian motion
    [Kondor, Lafferty 2002], [Coifman et al. 2005], [Lee at al. 2006],
    [Meyerhenke, Monien 2008], [Hashimoto et al. 2015], ...
  - effective resistance
    [Spielman, Srivastava 2011], [Ghosh, Boyd 2008], ...
Interpolation weights

\[(\uparrow^F_C)_{ij} = \begin{cases} 
\frac{\rho_{ij}}{\left( \sum_{k \in N(i)} \rho_{ik} \right)^{-1}} & i \in F, j \in N(i) \\
1 & i \in C, j = i \\
0 & \text{otherwise}
\end{cases}\]

- Define the interpolation weights for all F-nodes
- Intuitively, these weights are the probabilities for a vertex to share a common property (such as the partition in the partitioning problem) with the aggregates it belongs to
Coarse Graph by AMG weighted aggregation

$\leftarrow (\uparrow^f_c)^T L_f \uparrow^f_c$

$w_{IJ} = \sum_{l,k} (\uparrow^f_c)_{Il} \cdot w_{lk} \cdot (\uparrow^f_c)_{kJ}$

Note that well known matching-based multilevel solvers such as Metis, Scotch, KAHIP, Jostle, etc. can be formulated as restricted cases of AMG.
Multiscale Methods for Networks

Computational Optimization Problems

- Compression, Linear Arrangement, Bandwidth, 2-sum, Wavefront
- Partitioning, Clustering, Vertex Separator
- Nonlinear Dimensionality Reduction
- Response to Epidemics and Cyber Attacks
- Visualization

Network Modeling

- Network Generation
- Graph Sparsification

Machine Learning

- Support Vector Machines
- Text Analysis and Hypothesis Modeling
- Segmentation

More examples of non-matching coarsening: Eigensolvers (Livne, Brandt, Sanders, Henson,...), Random-walk ranking (Sanders, Henson, Sterck,...), Segmentation (Basri, Galun,...), Wavefront (Hu, ...) and more
Open Science Grid: collaboration network example

Goldberg, Leyffer, S “Optimal Response to Epidemics and Cyber Attacks on Networks”, 2015
connections between open sites

\[ x_i \in \{0, 1\} \]

infection probability at \( i \)

\[ \phi_i \]

number of shared users

\[ w_{ij} \]

probability of \( j \rightarrow i \) spread

\[ p_{ij} \]

Model

\[
\text{maximize } \sum_{ij \in E} w_{ij} x_i x_j
\]

subject to

\[
x_i - \prod_{j \in N(i)} (1 - p_{ij} \phi_j x_j) \leq t_i \quad \forall i \in V
\]

infection at node \( i \) is less than some constant

\[ x \in \{0, 1\}^n \]
function MSSolve(G)
if G is small then
    $S_f \leftarrow$ solve the problem exactly
else
    order infected nodes
    find coarse variables
    $G_c \leftarrow$ create coarse graph
    $S_c \leftarrow$ MSSolve($G_c$)
    $S_f \leftarrow$ Interpolate($S_c$)
    $S_f \leftarrow$ LocalRefinement($S_f$)
end if
return $S_f$

Leyffer, S “Fast Response to Infection Spread and Cyber Attacks on Large-scale Networks”, 2013
Coarsening

Jacobi over-relaxation

\[ H = (1 - \omega)I + \omega D^{-1}W \]

Algebraic distance is a strength of connection

\[
q_{ij}^{(k)} = \left( \sum_{r=1}^{R} |\chi_i^{(k,r)} - \chi_j^{(k,r)}|^p \right)^{1/p}, \text{ where } \chi^{(k,r)} = H^k \chi^{(0,r)}
\]

Used as sparsification for Galerkin and in detection of coarse variables

Ron, S, Brandt “Relaxation-based coarsening and multiscale graph organization”, 2011
Chen, S “Algebraic distance on graphs”, 2012

Coarse model

maximize

\[
\sum_{ij \in E_c} W_{ij} X_i X_j + \sum_{i \in V_c} A_i X_i
\]

subject to

\[
X_i - \prod_{j \in N(i)} (1 - P_{ij} \Phi_j X_j) \leq T_i \quad \forall i \in V_c
\]

\[
X \in \{0, 1\}^n
\]

\[
P_{ij}, W_{ij}, \Phi_i, X_i, T_i \leftarrow \text{AMG coarsening.}
\]

Galerkin reinforced by algebraic distance
Uncoarsening

\[
\begin{align*}
\text{maximize} & \sum_{i,j \in S} w_{ij} x_i x_j + \sum_{i \in S, j \notin S} w_{ij} x_i \tilde{x}_j + \sum_{i \in S} a_i x_i \\
\text{subject to} & \quad x_i - k_i \prod_{j \in N(i), j \notin S} (1 - p_{ij} \phi_{j,t-1} x_j) \leq b_i \quad \forall i \in V \\
& \quad x_i \in \{0, 1\} \quad \forall i \in V \\
& \quad k_i = \prod_{j \in N(i), j \notin S} (1 - p_{ij} \phi_{j,t-1} \tilde{x}_j)
\end{align*}
\]
Small random graphs, $|V|<100$ nodes
Erdos-Renyi, Barabasi-Albert, and R-MAT models
Large-scale networks, $10^4 < |V| < 10^6$

Multilevel algorithm is approximately 200-300 times faster than iterative combination of several solvers.

**Quality of the objective:**
Ratios between Multilevel Alg and best combination of several solvers

Sources: SNAP and UFL collections

Multiscale Methods for Networks
Ilya Safro, Clemson University
Classification problems: Weighted SVM

Imbalanced classification problems emerge in applications where the examples of one class (majority) greatly outnumber the examples of the other class (minority). The detection of the minority class is more important than the majority class. Penalize misclassification of each class with coefficients $C^+/C^-$

$$\min_{w,b,\xi} \frac{1}{2} \|w\|^2 + C^+ \sum_{i=1}^{n} \xi_i^2 + C^- \sum_{i=1}^{n} \xi_i^2$$

s.t. $y_i(w^T x_i - b) \geq 1 - \xi_i$,

More parameters are in kernels

- RBF $\exp(-\gamma \|x_i - x_j\|^2)$
- Cauchy $(1 + \frac{1}{\alpha \|x_i - x_j\|^2})^{-1}$
Multilevel SVM and Weighted SVM

Razzaghi, S “Scalable Multilevel Support Vector Machines”, ICCS 2015

Minority

Initial training data

Majority

Finest level SVs

Create approximate k-NN graph for each class or for their mixture

Main ideas:

- **Inherit** support vectors from the coarse level as training set
- Add some of their neighborhood
- **Inherit** parameters for model selection from the coarse level

Multiscale Methods for Networks
Ilya Safro, Clemson University
Coarse Variables

Types of separate coarsening for two classes

1. Iterative selection of (several) independent set(s) of nodes (similar to [Sakellaridi et al 2008])
2. Strict coarsening (merging pairs of variables based on some distance function)
3. AMG coarsening (Galerkin for separate classes)

Mutual coarsening for two classes?

1. Yes. Formulate as fuzzy SVM
2. No. Formulate as regular (W)SVM
Merging two classes: 
Probabilistic Support Vector Machines

- Probabilistic SVM

\[
\begin{align*}
\min_{w,b,\xi^+,\xi^-} & \quad \frac{1}{2}\|w\|^2 + C \sum_{i \in V_f} (\xi_i^+ m_i^+ + \xi_i^- m_i^-) \\
\text{s.t.} & \quad w^T x_i + b \geq 1 - \xi_i^+, \quad i \in V_f \\
& \quad w^T x_i + b \geq -1 + \xi_i^-, \quad i \in V_f
\end{align*}
\] (1a)

- \(m_i^+\) and \(m_i^-\) are the memberships for observation \(x_i\) with respect to the positive and negative classes

- How to coarsen probabilistic data points and their labels?
Merging two classes: Probabilistic Support Vector Machine

- Each coarse data point is chosen to be linear combination of a small number of fine data points, such that $X^{(C)}$ and $X^{(F)}$ are the data set of coarse and fine level,

$$X^{(C)} = P' \star X^{(F)}$$

- Fuzzy label ($y_i$) is defined as, $x_i$ has label $+1$ with fuzzy value ($m_i^+$) and label $-1$ with fuzzy value ($m_i^-$),

$$m_i^+ = \frac{\sum_{j \in F} P_{ij} \star m_j^+}{\sum_{j \in F} P_{ij}} \quad m_i^- = \frac{\sum_{j \in F} P_{ij} \star m_j^-}{\sum_{j \in F} P_{ij}}$$
Uncoarsening: Strict, AMG (separate classes) and Probabilistic WSVM (merged classes)

- Regular (W)SVM is often extremely expensive if model selection (finding parameters $C^+/C^-/\Gamma$) is applied.
- Apply grid search of $C^+/C^-/\Gamma/...$ at 2-3 coarsest levels only
- Inherit parameters from the coarse level. Update using bilinear interpolation.

Set of support vectors is relatively small

Refinement: Training is performed by pairs of clusters of support vectors using libSVM

Separating hyperplane
| Dataset          | $r_{imb}$ | $F$ | $|C^+ \cup C^-|$ | $|C^-|$ | $|C^+|$ | AMG WSVM | Single Level |
|------------------|----------|-----|-----------------|-------|--------|---------|------------|
| Letter26         | 0.96     | 16  | 20000          | 734   | 19266  | 18      | 333        |
| Ringnorm         | 0.50     | 20  | 7400           | 3664  | 3736   | 5       | 42         |
| Buzz             | 0.80     | 77  | 140707         | 27775 | 112932 | 957     | 70452      |
| Clean            | 0.85     | 166 | 6598           | 1017  | 5581   | 6       | 167        |
| Advertisement    | 0.86     | 1558| 3279           | 459   | 2820   | 91      | 412        |
| ISOLET           | 0.96     | 617 | 6238           | 240   | 5998   | 64      | 1367       |
| cod-rna          | 0.67     | 8   | 59535          | 19845 | 39690  | 92      | 1611       |
| Twonorm          | 0.50     | 20  | 7400           | 3703  | 3697   | 4       | 45         |
| Nursery          | 0.67     | 8   | 12960          | 4320  | 8640   | 33      | 519        |
| EEG Eye State    | 0.55     | 14  | 14980          | 6723  | 8257   | 45      | 447        |
| Protein          | 0.99     | 74  | 145751         | 1296  | 144455 | 1597    | 73311      |
| Forest           | 0.98     | 54  | 581012         | 9493  | 571519 | 13328   | 352500     |

- Model selection is applied
- Comparable quality except Advertisement and Forest in which AMG WSVM is better by 20% in G-mean

Time in seconds
Without Algebraic Distance

Sensitivity analysis of AMG-(W)SVM in terms of the interpolation order \((r)\) and performance measures for Buzz data set

<table>
<thead>
<tr>
<th>Metric</th>
<th>(r = 1)</th>
<th>(r = 2)</th>
<th>(r = 3)</th>
<th>(r = 4)</th>
<th>(r = 5)</th>
<th>(r = 6)</th>
<th>(r = 10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>G-mean</td>
<td>0.26</td>
<td>0.33</td>
<td>0.56</td>
<td>0.83</td>
<td>0.90</td>
<td>0.91</td>
<td>0.89</td>
</tr>
<tr>
<td>Sens</td>
<td>0.14</td>
<td>0.33</td>
<td>0.68</td>
<td>0.89</td>
<td>0.98</td>
<td>0.97</td>
<td>0.95</td>
</tr>
<tr>
<td>Spec</td>
<td>0.47</td>
<td>0.34</td>
<td>0.47</td>
<td>0.79</td>
<td>0.82</td>
<td>0.86</td>
<td>0.82</td>
</tr>
<tr>
<td>Acc</td>
<td>0.21</td>
<td>0.33</td>
<td>0.64</td>
<td>0.87</td>
<td>0.95</td>
<td>0.95</td>
<td>0.94</td>
</tr>
<tr>
<td>Prec</td>
<td>0.52</td>
<td>0.94</td>
<td>0.85</td>
<td>0.94</td>
<td>0.96</td>
<td>0.97</td>
<td>0.95</td>
</tr>
<tr>
<td>F-score</td>
<td>0.23</td>
<td>0.49</td>
<td>0.69</td>
<td>0.91</td>
<td>0.97</td>
<td>0.94</td>
<td>0.93</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Metric</th>
<th>(r = 1)</th>
<th>(r = 2)</th>
<th>(r = 3)</th>
<th>(r = 4)</th>
<th>(r = 5)</th>
<th>(r = 6)</th>
<th>(r = 10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>G-mean</td>
<td>0.26</td>
<td>0.40</td>
<td>0.60</td>
<td>0.90</td>
<td>0.93</td>
<td>0.93</td>
<td>0.94</td>
</tr>
<tr>
<td>Sens</td>
<td>0.14</td>
<td>0.32</td>
<td>0.74</td>
<td>0.95</td>
<td>0.98</td>
<td>0.98</td>
<td>0.98</td>
</tr>
<tr>
<td>Spec</td>
<td>0.47</td>
<td>0.5</td>
<td>0.48</td>
<td>0.85</td>
<td>0.88</td>
<td>0.89</td>
<td>0.89</td>
</tr>
<tr>
<td>Acc</td>
<td>0.21</td>
<td>0.35</td>
<td>0.69</td>
<td>0.93</td>
<td>0.96</td>
<td>0.97</td>
<td>0.97</td>
</tr>
<tr>
<td>Prec</td>
<td>0.52</td>
<td>0.93</td>
<td>0.86</td>
<td>0.96</td>
<td>0.97</td>
<td>0.97</td>
<td>0.97</td>
</tr>
<tr>
<td>F-score</td>
<td><strong>0.23</strong></td>
<td><strong>0.61</strong></td>
<td><strong>0.72</strong></td>
<td><strong>0.95</strong></td>
<td><strong>0.98</strong></td>
<td><strong>0.98</strong></td>
<td><strong>0.98</strong></td>
</tr>
</tbody>
</table>

- AMG-WSVM is more robust towards imbalanced data
- As \(r\) increases, performance measures improve but computational time increases
## Network Generation and Modeling

**Practical task**

- Simulate and verify algorithms, policies, and scenarios on networks that can be created by similar processes

<table>
<thead>
<tr>
<th>Original network</th>
<th>Artificial networks</th>
<th>Artificial network</th>
</tr>
</thead>
<tbody>
<tr>
<td>US Western States Power Grid Watts, Strogatz 1998</td>
<td><img src="image" alt="Artificial networks" /></td>
<td>This network has similar degrees, some eigenvalues, diameter but ...</td>
</tr>
</tbody>
</table>
<pre><code>                                                                                       |                                                                                     | is it really similar to the original network?                                        |
</code></pre>
Properties that are preserved by most of the existing network generators (such as Chung-Lu, Stochastic Kronecker Graph and Block Two-Level Erdös–Rényi):

- degree distribution
- clustering coefficient
- some eigenvalues
- diameter, etc.

Common algorithm: 1) start with empty or small graph 2) add some components at random, at the end preserving several properties.

**What makes the resulting graphs non-realistic?**

- These properties are different at different resolutions
- Too many operations such as randomization and replication take us away from the realistic structure
What makes a synthetic network realistic?

A good synthetic network must meet two criteria

• **Realism** with respect to structural features that govern domain-specific processes. For example,
  ➢ Social networks should emulate emergent sociological phenomena.
  ➢ Interdependent infrastructure systems should demonstrate realistic resilience, joint performance, and potential mutual failures.
  ➢ Metabolic interactions should ultimately reflect biochemical properties of a cell.

• Normally-occurring **diversity** in a system.

Goals: benchmarking, robustness evaluation, algorithm verification, anonymization, and generating scenarios.
Multiscale Entropic Network Generator

http://www.cs.clemson.edu/~isafro/musketeer

Control of similarities and properties

Original network → Generated network

Coarsening → Coarsening

Control of similarities and properties

local, controlled randomization and property generation at all scales
To create a new edge $uv$

- $d_2(i, j) :=$ second shortest path between two neighbors
- Estimate $P[d_2(i, j) = k]$

1. Sample $x$ from the estimated distribution
2. Randomly select $u$ and find $v$ within distance $x$
3. Create edge $uv$ with edge weight from a given distribution
Toy Example: Mesh 33x33 by

Original graph mesh 33x33

Fine level changes

Coarse level changes
Example: Power Grid by

Original graph: Watts, Strogatz 1998
100% of fine levels’ changes
Coarse levels’ changes

Generated graph is 3 times bigger
... + coarse level changes
Several coarse levels’ changes
Example: Power Grid by Multiscale Methods for Networks

Ilya Safro, Clemson University
Example: Barabasi-Albert Model by Ilya Safro, Clemson University

Multiscale Methods for Networks
Ilya Safro, Clemson University
SEIR cascade on Expanded Colorado Springs Network

susceptible → exposed → recovered → susceptible
• Ron, S, Brandt “Relaxation-based Coarsening and Multiscale Graph Organization”, SIAM Multiscale Modeling and Simulation, 2011

• Chen, S “Algebraic Distance on Graphs” SIAM Journal on Scientific Computing, 2011

• Leyffer, S “Fast Response to Infection Spread and Cyber Attacks on Large-scale Networks”, Journal on Complex Networks, 2013

• Goldberg, Leyffer, S “Optimal Response to Epidemics and Cyber Attacks in Networks”, Networks, 2015

• Gutfraind, S, Meyers “Multiscale Network Generation”, FUSION, 2015
  http://www.cs.clemson.edu/~isafro/musketeer (implemented in Python, soon in C++)
  (can be used to generate graphs and matrices for your experiments!)

• Razzaghi, S “Scalable Support Vector Machines”, ICCS 2015

Thank you!
Coarse Variables: Extended Independent Set

 Algorithm 1 The Coarsening

1: **Input:** \( G = (V, E) \) for class \( C \)
2: \( \hat{V} \leftarrow \text{select maximal independent set in } G \)
3: \( \hat{U} \leftarrow V \setminus \hat{V} \)
4: while \( |\hat{V}| < Q \cdot |V| \) do
5:   while \( \hat{U} \neq \emptyset \) do
6:     randomly pick \( i \in \hat{U} \)
7:     \( \hat{U} \leftarrow \hat{U} \setminus \{i\} \)
8:     \( \hat{U} \leftarrow \hat{U} \setminus \{\text{neighbors of } i \text{ in } \hat{U}\} \)
9:     \( \hat{V} \leftarrow \hat{V} \cup \{i\} \)
10: end while
11: \( \hat{V} \leftarrow \hat{V} \cup \{i\} \)
12: end while
13: \( \hat{U} \leftarrow V \setminus \hat{V} \)
14: end while
15: return \( \hat{V} \)
Algorithm 1 The Refinement at level $i$

1. if $i$ is the coarsest level then
2. \hspace{1em} $S_i \leftarrow$ Apply SVM on $X_i$ (find the best parameters using UD)
3. end if

19: Return $S_i, C_i, \gamma_i$
Performance Measures

- **Accuracy**: the percent of the correctly classified examples
  \[
  \text{Accuracy} = \frac{TP}{TP + FN}
  \]

- **Sensitivity** = \( \frac{TP}{TP + FN} \), **Specificity** = \( \frac{TN}{TN + FP} \)

- **G-mean** = \( \sqrt{\text{Sensitivity} \times \text{Specificity}} \)

- **F-Measure** = \( \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \)

- **Precision** = \( \frac{TP}{TP + FP} \)

**Table 1: Confusion Matrix**

<table>
<thead>
<tr>
<th></th>
<th>Positive class</th>
<th>Negative class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive class</td>
<td>TP</td>
<td>FP</td>
</tr>
<tr>
<td>Negative class</td>
<td>FN</td>
<td>TN</td>
</tr>
</tbody>
</table>
Figure: Sensitivity of AMG-(W)SVM in terms of G-mean metric and computational time to $r$ for Buzz data
Table: Probabilistic SVM results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Gmean</th>
<th>SN</th>
<th>SP</th>
<th>ACC</th>
<th>Preci.</th>
<th>F1</th>
<th>time(sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Letter</td>
<td>0.98</td>
<td>1.00</td>
<td>0.97</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>9</td>
</tr>
<tr>
<td>Clean</td>
<td>0.99</td>
<td>1.00</td>
<td>0.99</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>6</td>
</tr>
<tr>
<td>Twonorm</td>
<td>0.97</td>
<td>0.97</td>
<td>0.97</td>
<td>0.97</td>
<td>0.97</td>
<td>0.97</td>
<td>5</td>
</tr>
<tr>
<td>Ringnorm</td>
<td>0.98</td>
<td>0.99</td>
<td>0.97</td>
<td>0.98</td>
<td>0.99</td>
<td>0.98</td>
<td>7</td>
</tr>
<tr>
<td>EEG</td>
<td>0.83</td>
<td>0.88</td>
<td>0.78</td>
<td>0.83</td>
<td>0.83</td>
<td>0.85</td>
<td>33</td>
</tr>
<tr>
<td>Cod-rna</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
<td>0.96</td>
<td>123</td>
</tr>
<tr>
<td>Advertisement</td>
<td>0.66</td>
<td>0.99</td>
<td>0.44</td>
<td>0.91</td>
<td>0.92</td>
<td>0.95</td>
<td>43</td>
</tr>
<tr>
<td>Nursary</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>20</td>
</tr>
<tr>
<td>Isolet</td>
<td>0.46</td>
<td>0.72</td>
<td>0.29</td>
<td>0.71</td>
<td>0.72</td>
<td>0.83</td>
<td>43</td>
</tr>
<tr>
<td>Buzz</td>
<td>0.91</td>
<td>0.99</td>
<td>0.84</td>
<td>0.96</td>
<td>0.99</td>
<td>0.97</td>
<td>898</td>
</tr>
</tbody>
</table>

- Probabilistic AMG-SVM is competitive in terms of accuracy performance measures.
- Computational time is similar or larger than Multilevel(W)SVM for some cases, since each data point $x_i$ belongs to both classes, whereas in Multilevel(W)SVM case it only belong to one, and therefore increases the runtime during training.
Benchmark II – Missing Values, Noise

In collaboration with O. Roderick, Geisinger Health System

Datasets are not huge but large enough to make nonlinear classification slow enough when model selection is required.

Dataset: **80.000 points, 13 features, WSVM**

<table>
<thead>
<tr>
<th></th>
<th>G-mean</th>
<th>SN</th>
<th>SP</th>
<th>ACC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adaptive Reg/Lasso</td>
<td>0.7516</td>
<td>0.8903</td>
<td>0.6345</td>
<td>0.7619</td>
</tr>
<tr>
<td>MLSVM</td>
<td>0.8012</td>
<td>0.9750</td>
<td>0.6583</td>
<td>0.8496</td>
</tr>
<tr>
<td>MLWSVM</td>
<td>0.8016</td>
<td>0.9739</td>
<td>0.6598</td>
<td>0.8495</td>
</tr>
</tbody>
</table>

Dataset: **240.000 points, 16 features, WSVM**

<table>
<thead>
<tr>
<th>Class</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adaptive Reg/Lasso</td>
<td>0.58</td>
<td>0.54</td>
<td>0.53</td>
<td>0.51</td>
<td>0.59</td>
</tr>
<tr>
<td>WMLSVM</td>
<td>0.73</td>
<td>0.50</td>
<td>0.44</td>
<td>0.50</td>
<td>0.71</td>
</tr>
</tbody>
</table>
Customer Survey Classification Problem
Dataset: 30K points, 170K features

- There are five classes of customer surveys: Design, Fault, Satisfaction, Irrelevant, Feature
- The data is **imbalanced**, it is very important to detect small classes!
## WSVM

<table>
<thead>
<tr>
<th></th>
<th>Gmean</th>
<th>SN</th>
<th>SP</th>
<th>ACC</th>
<th>F1 Score</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class2</td>
<td>0.81</td>
<td>0.79</td>
<td>0.82</td>
<td>0.81</td>
<td>0.83</td>
<td>0.82</td>
</tr>
<tr>
<td>Class3</td>
<td>0.46</td>
<td>0.99</td>
<td>0.22</td>
<td>0.97</td>
<td>0.88</td>
<td>0.91</td>
</tr>
<tr>
<td>Class4</td>
<td>0.94</td>
<td>0.94</td>
<td>0.94</td>
<td>0.94</td>
<td>0.93</td>
<td>0.96</td>
</tr>
<tr>
<td>Class5</td>
<td>0.43</td>
<td>0.99</td>
<td>0.20</td>
<td>0.98</td>
<td>0.98</td>
<td>0.98</td>
</tr>
<tr>
<td>Class6</td>
<td>0.65</td>
<td>0.91</td>
<td>0.46</td>
<td>0.85</td>
<td>0.88</td>
<td>0.91</td>
</tr>
</tbody>
</table>

## MLWSVM

<table>
<thead>
<tr>
<th></th>
<th>Gmean</th>
<th>SN</th>
<th>SP</th>
<th>ACC</th>
<th>F1 Score</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class2</td>
<td>0.80</td>
<td>0.81</td>
<td>0.80</td>
<td>0.80</td>
<td>0.82</td>
<td>0.83</td>
</tr>
<tr>
<td>Class3</td>
<td>0.63</td>
<td>0.85</td>
<td>0.47</td>
<td>0.80</td>
<td>0.88</td>
<td>0.91</td>
</tr>
<tr>
<td>Class4</td>
<td>0.92</td>
<td>0.90</td>
<td>0.93</td>
<td>0.91</td>
<td>0.93</td>
<td>0.96</td>
</tr>
<tr>
<td>Class5</td>
<td>0.43</td>
<td>0.97</td>
<td>0.20</td>
<td>0.96</td>
<td>0.98</td>
<td>0.98</td>
</tr>
<tr>
<td>Class6</td>
<td>0.64</td>
<td>0.85</td>
<td>0.48</td>
<td>0.79</td>
<td>0.88</td>
<td>0.91</td>
</tr>
<tr>
<td>Class</td>
<td>Single (W)SVM</td>
<td>ML (W)SVM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>---------------</td>
<td>-----------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class2</td>
<td>1703</td>
<td>93</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class3</td>
<td>948</td>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class4</td>
<td>1131</td>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class5</td>
<td>1243</td>
<td>47</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class6</td>
<td>1638</td>
<td>62</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Distance between points

- Any kernel can be used instead of $\langle x_i, x_j \rangle$
- *Approximate* k-NN graph is created to preserve a good complexity

The algorithm can be very sensitive to this, so advanced connectivity strength measure of can be critical

The Lagrangian dual is given by

$$
\max \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j)
$$

subject to

$$
\sum_{j=1}^{n} \alpha_i y_i = 0
$$

$$
0 \leq \alpha_i \leq C^+
$$

if $y_i = +1$ and $i = 1, \ldots, n$

$$
0 \leq \alpha_i \leq C^-
$$

if $y_i = -1$ and $i = 1, \ldots, n$
Coarse Variables: Strict and AMG

- Choose a dominating set $C \subset V$ s.t. all others from $F = V \setminus C$ are “strongly coupled” to $C$
- “Strongly coupled” = Kernel coupling + algebraic distances $\rho_{ij}$
  S, Chen “Algebraic distance on graphs”, 2012
- Restriction matrix is sparsified by algebraic distances
Uncoarsening

Initial training data → Finest level SVs

Coarsening

Initial classifier and SVs → Refined classifier

Projected classifier

Coarsest level
Simulation of Cyber Attack

Each point is an average over 100 experiments.

- Black dots: Original network
- Red squares: Artificial network

Graph showing the Average Threat Level against Attack/Defense Iterations.
To create a new edge $uv$

- $rw(u, v, w) :=$ length of a random walk from $v$ to $w$, where both $v$ and $w$ are neighbors of $u$.
- Estimate $P[rw(u, v, w) = k$ for any $w$ ]

1. Choose node $x$ and sample from the estimated distribution of random walks
2. Perform a random walk from $x$ and close it
To create a new node $u$

1. Sample degree on $u$ from the degree distribution of the same level
2. Randomly select $v$ and connect it to $u$
3. Subsequent edges are inserted using edge sampling
Examples of Computational Problems

- VLSI placement
- Graph partitioning
- Eigensolvers
- Clustering
- Linear arrangement
- Community detection
- Modularity
- Traveling salesman
- Visualization
- Compression-friendly ordering
- Coloring
- Spectral problems
Susceptible-Infected-Susceptible Model

The Kephart-White SIS model parameters:

- $S$ - number of susceptible nodes;
- $I$ - number of infected nodes;
- $\beta$ - infection transmission rate;
- $\delta$ - rate of recovery from infection.

\[
\begin{align*}
\frac{dI}{dt} &= \lambda S - \delta I \\
\frac{dS}{dt} &= \delta I - \lambda S.
\end{align*}
\]

Chakrabarti et al. proposed a dynamical system of SIS

\[
1 - \phi_{i,t} = (1 - \phi_{i,t-1}) h_{i,t} + \delta \phi_{i,t-1} h_{i,t}, \quad i = 1 \ldots |V|,
\]

to describe the probability of keeping $i$ in $S$, where

\[
h_{i,t} = \prod_{j \in N(i)} (1 - p_{ij} \phi_{j,t-1}).
\]

Epidemic threshold $\tau$, a measure to predict when the infection outbreak disappears (comparable to $\beta/\delta$).
Example: C-18 Optimization Matrix by

Original graph: C-18  

100% of fine level changes  

Coarse level changes

Generated graph is 3 times bigger  

... + coarse level changes  

Several coarse level changes

Multiscale Methods for Networks
Ilya Safro, Clemson University
Iterated Local Search vs Multiscale HIV spread model, $|V|=20K$