Bayesian learning with big data

Yuan (Alan) Qi
alanqi@purdue.edu
Purdue University
Data
A lot of data
Outline

• Bayesian online learning

• Low-rank Gaussian process models
Ubiquitous data streams
How to handle massive data or data stream?

Online learning
Bayesian treatment

- Linear dynamic systems: Kalman filtering
- Nonlinear dynamic systems: Particle filtering
Previous online learning work

- Stochastic gradient: Perceptron (Rosenblatt 1962)
- Natural gradient (Le Roux et al., 2007)
- Online Gaussian process classification (Csató & Opper, 2002)
- Passive-Aggressive algorithm (Crammer et al. 2006)
What has been ignored?

Previous data points
New idea

Summarize all previous samples by representative virtual points & dynamically update them
Key intuition

• Many data points are not important to classification
  - Prune them
• Many data points are similar
  - Group them
Algorithm: virtual vector machine

Apply this intuition in a Bayesian framework

\[ q(w) \propto r(w) \prod_{i} f(b_i; w) \]

\[ r(w) \sim \mathcal{N}(m_r, V_r) \]

- \( q(w) \): Approximate posterior of classifier \( w \)
- \( b_i \): A virtual point in a small buffer
- \( r(w) \): Gaussian Residue
Dynamic update

Two operations to maintain a fixed buffer size:

• Eviction

• Merging
Eviction

Remove point with smallest impact on classifier posterior distribution
Eviction

Point with biggest “Bayesian margin”

\[ m_w^T x / \sqrt{x^T V_w x} \]

Remove point with smallest impact on classifier posterior distribution
Version space

The subset of all hypotheses that are consistent with the observed training examples

Four data points: hyperplanes
Version space: brown area
Deleting unimportant point

Version space: brown area
EP approximation: red ellipse
Four data points: hyperplanes

Version space with three points after deleting one point (with the largest distance to version space)
Merging

Merge similar points leading to smallest impact on classifier posterior distribution
Merging

Merge similar points leading to smallest impact on classifier posterior distribution
Merging similar points

Version space: brown area
EP approximation: red ellipse
Four data points: hyperplanes

Version space with three points after merging two similar points
New algorithm

• Assumed density filtering (ADF):
  - Project a posterior distribution to the exponential family given a data point

• Inverse ADF:
  - Find a virtual point $b'$ that will lead to a desired projection from two real points

$$\text{proj} \left[ f(b_1; w) f(b_2; w) \tilde{q}_+^{12}(w) \right] =$$

$$\text{proj} \left[ f(b'; w) g(w) \tilde{q}_+^{12}(w) \right]$$
 Estimation accuracy

- VVM achieves a smooth trade-off between computation cost and estimation accuracy.
- ADF uses a buffer of size one and EP uses the whole sequence of 300 data points.
Thyroid classification

Results averaged over 10 random permutations of data
Email spam classification

Results averaged over 10 random permutations.

Buffer size:  VVM 30
             SOGP 143
             PA 80
Outline

- Bayesian online learning
- Low-rank Gaussian process models
Gaussian process

- Nonparametric Bayesian prior over functions
- Computational bottleneck: $O(N^3)$ for regression
Sparse GP

• Summarize data by a few pseudo inputs (Snelson & Ghahramani 2006)

• Summarize data by a few pseudo data clouds (Qi et al. 2010)
Exact GP

Pseudo input (i.e., FITC)
Key observation

• Previous approaches: *compress* data into a sparse representation including pseudo data points or clouds

• **PCA**: the optimal compact representation among all orthogonal bases

• How about PCA for GP?
EigenGP: sparse PCA + GP

• KL expansion of GP prior by Nystrom method:
  \( O(N^2) \rightarrow O(M^2N) \)

• Select eigenfunctions by evidence maximization

• Handle nonlinear likelihood by Expectation propagation
Eigenfunctions of Gaussian kernel

Top four eigenfunctions

Selected eigenfunctions
Boston Housing (400/506 for training)
RMSE of Nystrom: 312.5, 41.68, 15.17 and 6.480

Pumadyn-8nm (2000/8192 for training)
Classification results on digits 8 vs 9

<table>
<thead>
<tr>
<th>Number of Basis/Rank</th>
<th>Classification Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.015</td>
</tr>
<tr>
<td>150</td>
<td>0.015</td>
</tr>
<tr>
<td>250</td>
<td>0.015</td>
</tr>
</tbody>
</table>

EigenGP*: fix the eigenvalues
EigenGP: sparsify eigenvalues

EIGEN−GP*
EIGEN−GP
Classification results

Spambase

3 vs 8

5 vs 8
Semi-supervised classification

Ionosphere (351 points)

20 Newsgroup (1976 points)

TDT2 (3672 points)
Conclusions

- Virtual vector machines: Bayesian approach that outperforms alternative state-of-the-art online learning approaches.

- EigenGP: low-rank GP for fast inference and high prediction accuracy