Triple Jump Acceleration for the EM Algorithm and Its Extrapolation-based Variants

Han-Shen Huang (黃漢申)
Bo-Hou Yang (楊博厚)
Chun-Nan Hsu (許鈞南)
Institute of Information Science, Academia Sinica

2006/7/24

Motivation

- Given an incomplete data set, the EM *[Dempster et al. 1977]* algorithm iteratively searches for the maximum likelihood estimate of a probabilistic model. *However, the search usually converges slowly under these conditions because more iterations or time for each iteration are required:*
  - High missing rate
  - Large training data set
  - Large parameter vector
- Therefore, accelerating EM is desired for training probabilistic models.

Brief Summary of Our Work

- **Accelerates the following:**
  - EM
  - Parameterized EM (pEM) *[Bauer et al. 1997]*
  - Adaptive overrelaxed EM (aEM) *[Salakhutdinov & Roweis 2003]*
- **Should be able to accelerate:**
  - GIS for conditional random field
  - Those can be formulated as fixed-point iteration methods: $\theta = M(\theta)$

Parameter Estimation Problem

- **Goal: find θ^* that maximizes $L(\theta)$**
  - θ : parameter vector of a probabilistic model
  - $L(\theta)$: log-likelihood with the training data
  - θ^* : maximum likelihood estimate

- **Influence of incomplete data**
  - $L(\theta)$ contains many local maxima
  - Search for local maxima

The EM Algorithm

Repeat (in iteration t)

$$\theta^{(t)} = M(\theta^{(t-1)})$$

Until $L(\theta^{(t)}) - L(\theta^{(t-1)}) < \delta$

- ***M*** **: an EM mapping, E-step + M-step**
- **Likelihood increases monotonically:**

$$L(\theta^{(t)}) \geq L(\theta^{(t-1)})$$

- **Local maximum:** $\theta^* = M(\theta^*)$

Taylor Expansion of M

◆In the neighbor of θ^* , we apply Taylor expansion to *M* *[Dempster et al. 1977]* :

$$\theta^{(t+1)} = M(\theta^{(t)}) \approx \theta^* + M'(\theta^*)(\theta^{(t)} - \theta^*) = \theta^* + J(\theta^{(t)} - \theta^*)$$

where *J* is the Jacobian of *M*.

◆Applying *M* to $\theta^{(t)}$ for h times, we have:

$$\theta^{(t+h)} = \theta^* + J^h(\theta^{(t)} - \theta^*)$$

Eigenvalues of J

- The eigen decomposition of J is:

$$J = Q \begin{pmatrix} \lambda_1 & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \lambda_n \end{pmatrix} Q^{-1} = Q\Lambda Q^{-1}$$

- The eigenvalues of J are expected to lie in [0, 1) *[Dempster et al. 1977]* .

Convergence Rate of EM

◆**Since** $0 \leq \lambda_i < 1$ **, we have** $\lim_{h \to \infty} J^h = 0$

$$J^h = Q \begin{pmatrix} \lambda_1^h & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & \lambda_n^h \end{pmatrix} Q^{-1} = Q \Lambda^h Q^{-1}$$

$$\theta^{(t+h)} = \theta^* + J^h(\theta^{(t)} - \theta^*)$$

◆**Therefore, the convergence rate is determined by** λ_{max} *[Dempster et al. 1977]* **.**

Parameterized EM (pEM)

Repeat (in iteration t)

$$\theta^{(t)} = M_\eta(\theta^{(t-1)})$$

Until $L(\theta^{(t)}) - L(\theta^{(t-1)}) < \delta$

- $M_\eta(\theta^{(t-1)}) = \theta^{(t-1)} + \eta(M(\theta^{(t-1)}) - \theta^{(t-1)})$
- **Likelihood increases monotonically in the neighborhood of** θ^* **if** $0 < \eta < 2$ *[Bauer et al. 1997]* **. pEM with** $\eta = 1$ **is EM.**
- **Local maximum:** $\theta^* = M(\theta^*) = M_\eta(\theta^*)$

Convergence Rate of pEM (1)

◆The eigenvalues of the Jacobian of M' are:

$$\lambda_{\eta i} = (1-\eta) * 1.0 + \eta\lambda_i$$

◆ Convergence rate is determined by $\max\{|\lambda_{\eta max}|, |\lambda_{\eta min}|\}$ because $\lambda_{\eta i} < 0$ is possible.

◆pEM is faster than EM if

$$\max\{|\lambda_{\eta max}|, |\lambda_{\eta min}|\} < \lambda\max$$

Convergence Rate of pEM (2)

- Optimal learning rate η^* is:

$$\eta^* = \frac{2}{2-\lambda_{max}-\lambda_{min}}$$

which minimizes $\max\{|\lambda_{\eta max}|, |\lambda_{\eta min}|\}$

- η^* is obtained by solving $\lambda_{\eta max} = -\lambda_{\eta min}$
- pEM with η^* is faster than EM

Adaptive Overrelaxed EM (aEM)

[Salakhutdinov & Roweis 2003]

- ◆pEM with dynamic η.
- ◆If $L(\theta^{(t)}) - L(\theta^{(t-1)}) \geq \delta$, use $\eta = 1.1 * \eta$ in the next iteration.
- ◆If $L(\theta^{(t)}) - L(\theta^{(t-1)}) < \delta$, discard the update and use $\eta = 1.0$ in the next iteration.

Aitken’s Acceleration for EM (1)

[McLachlan & Krishnan, 1997]

◆In the neighborhood of θ^*, we have

$$\theta^* = \theta^{(t)} + \sum_{h=0}^{\infty} (\theta^{(t+h+1)} - \theta^{(t+h)}).$$

$$\begin{aligned} \theta^* &\approx \theta^{(t)} + \sum_{h=0}^{\infty} J^h (\theta^{(t+1)} - \theta^{(t)}) \\ &= \theta^{(t)} + (I - J)^{-1} (\theta_{EM}^{(t)} - \theta^{(t)}) \end{aligned}$$

where $\theta_{EM}^{(t)} = M(\theta^{(t)})$.

Aitken's Acceleration for EM (2)

$$
\begin{aligned}
(I-J)^{-1} &= \left[Q\left[I-\Lambda\right]Q^{-1}\right]^{-1} \\
&= Q[I-\Lambda]^{-1}Q^{-1} \\
&= Q\begin{pmatrix} \frac{1}{1-\lambda_1} & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & \frac{1}{1-\lambda_n} \end{pmatrix} Q^{-1}
\end{aligned}
$$

$$\frac{1}{1-\lambda_i} = 1 + \lambda_i + \lambda_i^2 + \cdots$$

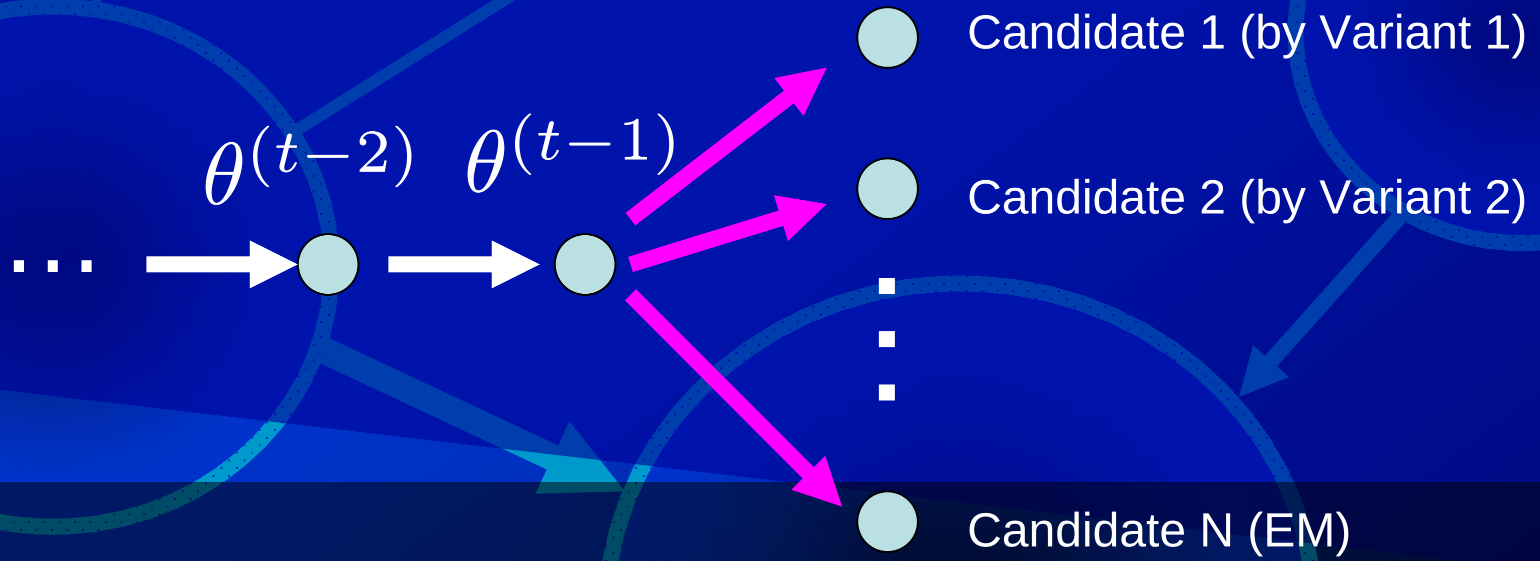
- However, exact estimation of J might be intractable for complicated models so that Aitken's acceleration is hard to use *[Hesterberg 2005]*.

Our Solution to Accelerate EM

- Triple jump framework to integrate previous algorithms.
- Simple approximation of J to accelerate the slowest direction (along the eigenvector corresponding to $\max\{|\lambda_{\eta max}|, |\lambda_{\eta min}|\}$).
- Theoretical and empirical verification

Triple Jump Framework (1)

◆In iteration *t*, TJ selects the first candidate as $\theta^{(t)}$ that satisfies $L(\theta^{(t)}) - L(\theta^{(t-1)}) \geq \delta$.



$\theta^{(t-2)}$ $\theta^{(t-1)}$

…

Candidate 1 (by Variant 1)

Candidate 2 (by Variant 2)

⋮

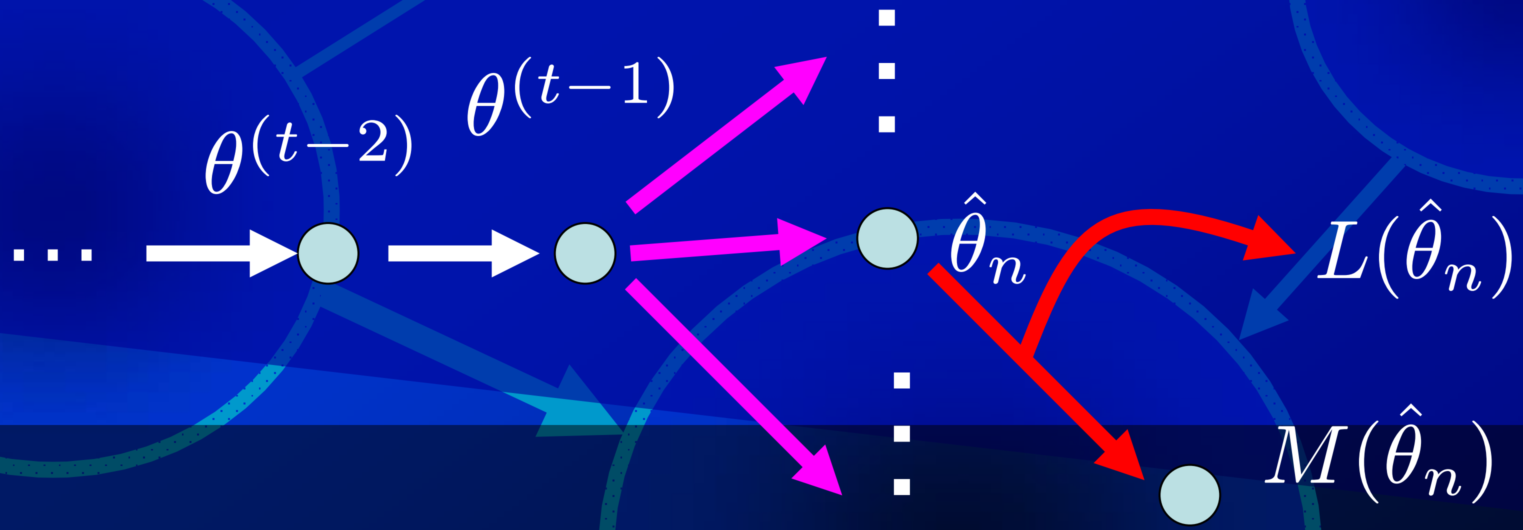
Candidate N (EM)

Triple Jump Framework (2)

- Candidate n is checked by

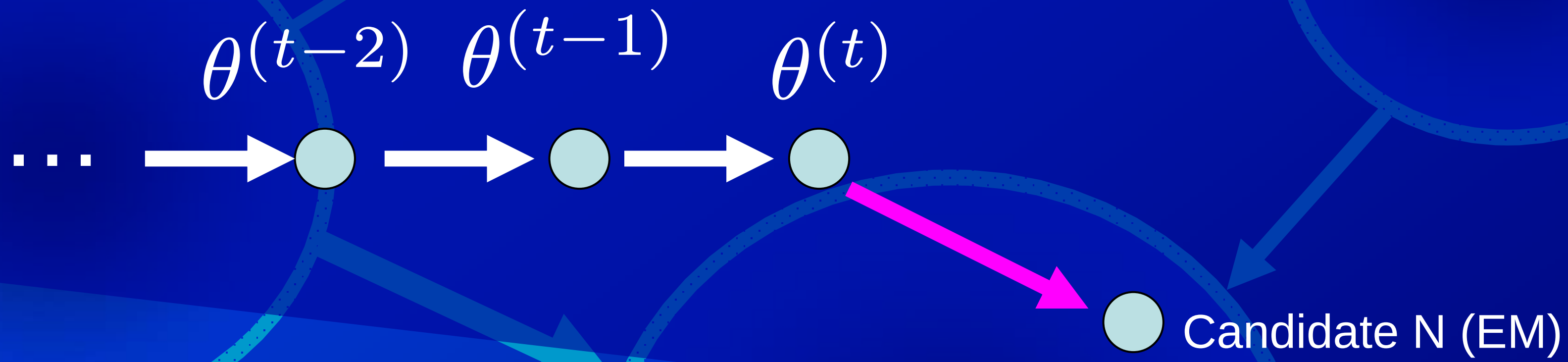
$$[M(\theta), L(\theta)] = M1(\theta)$$

which is the EM mapping plus few additional cost to compute the likelihood of the input.



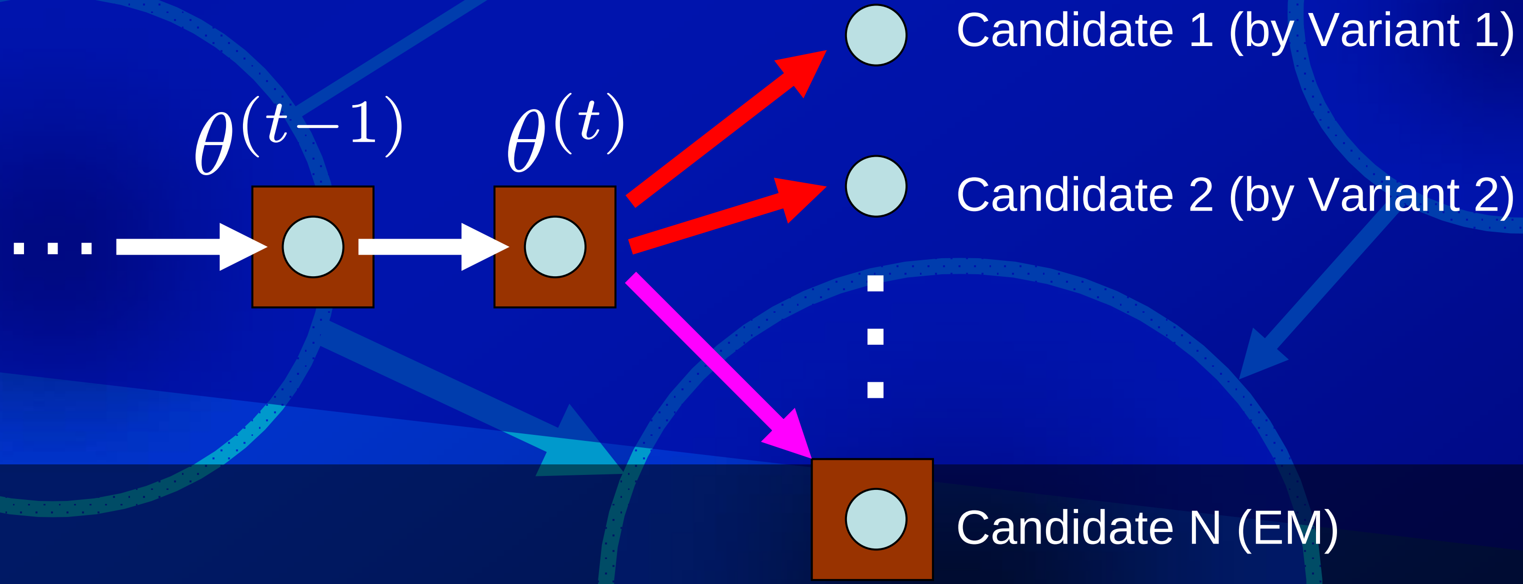
Triple Jump Framework (3)

- If $L(\hat{\theta}_n) - L(\theta^{(t-1)}) \geq \delta$, unchecked candidates are discarded. $\hat{\theta}_n$ becomes $\theta^{(t)}$, and $M(\hat{\theta}_n)$ becomes Candidate N for iteration t+1.



Triple Jump Framework (4)

- Other candidates are generated by candidate N and previous parameter vectors by extrapolation.



Advantages of TJ Framework

- ◆Easy to achieve acceleration by using EM directly as a subroutine
- ◆Easy to integrate many EM variants
- ◆Needless to handle the failure of extrapolation (naturally handled by EM, the last candidate)

TJEM Extrapolation (1)

- **Estimate largest eigenvalue with:**

$$\gamma^{(t)} \equiv \frac{\|\theta_{EM}^{(t)} - \theta^{(t)}\|}{\|\theta^{(t)} - \theta^{(t-1)}\|}$$

where $\theta^{(t)} = M(\theta^{(t-1)})$ **based on the requirement of the Aitken’s acceleration. Therefore, the cycle of a TJEM extrapolation is two EM operations and a far jump, like hop, step, and jump in a triple jump.**

TJEM Extrapolation (2)

- **Assign $J = \gamma^{(t)}$ and perform the Aitken's acceleration. That is,**

$$J = Q \begin{pmatrix} \gamma^{(t)} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \gamma^{(t)} \end{pmatrix} Q^{-1}$$

TJEM Algorithm

- Triple Jump Framework
- Candidate 1: by TJEM Extrapolation
- Candidate 2: by EM

TJpEM Extrapolation

◆Use M_η instead of M in the Aitken's acceleration.

$$\gamma_\eta^{(t)} \equiv \frac{\|\theta_\eta^{(t)} - \theta^{(t)}\|}{\|\theta^{(t)} - \theta^{(t-1)}\|}$$

where $\theta^{(t)} = M_\eta(\theta^{(t-1)})$.

TJpEM Algorithm

- ◆Triple Jump Framework
- ◆Candidate 1: TJpEM Extrapolation
- ◆Candidate 2: pEM Extrapolation
- ◆Candidate 3: EM

Convergence Properties of TJpEM Algorithm

- **Suppose that**
  - TJpEM extrapolation is successful
  - $\max\{|\lambda_{\eta max}|, |\lambda_{\eta min}|\}$ is estimated accurately
- **The i-th eigenvalue of the Jacobian of the composition of pEM + TJpEM extrapolation is:**

$$\alpha_{\eta i} = \lambda_{\eta i}(1 - \eta' + \eta'\lambda_{\eta i}) = \lambda_{\eta i}\frac{\lambda_{\eta i} - \gamma_{\eta}^{(t)}}{1 - \gamma_{\eta}^{(t)}}.$$

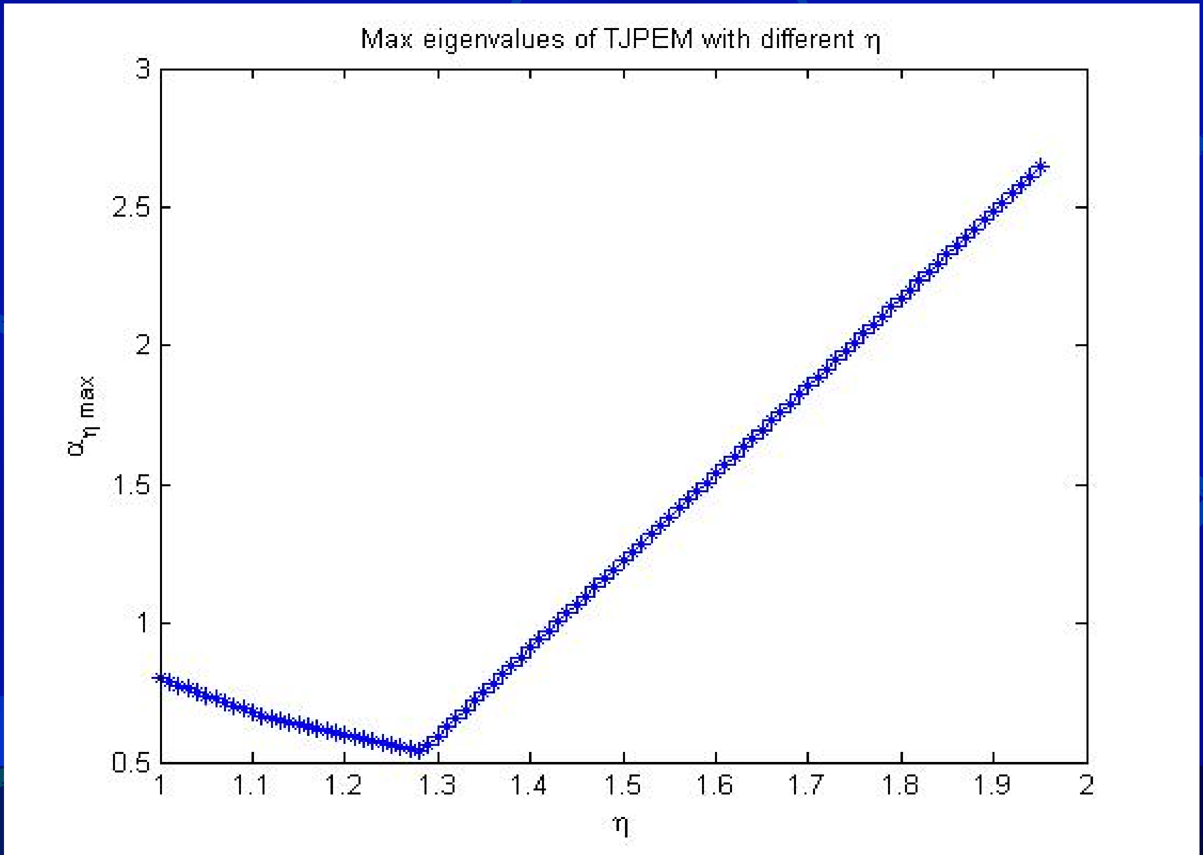
$$\alpha_{\eta i} = \lambda_{\eta i}\frac{\eta(\lambda_i - \lambda_{max})}{\eta(1 - \lambda_{max})} = \lambda_{\eta i}\frac{\lambda_i - \lambda_{max}}{1 - \lambda_{max}}$$

Convergence Rates of TJEM and TJpEM

- **Theorem :** *The TJpEM algorithm with a proper learning rate converges faster than the TJEM algorithm.*

Convergence Rates of TJpEM with Different Learning Rates

- Eigenvalues are 0.1, 0.2, … 0.9.



TJ²pEM Extrapolation

◆The goal of TJ²pEM is to reduce the impact of negative eigenvalues.

◆Conceptually, we combine two pEM operations into one (M_η^2) so that the all eigenvalues (λ_η^2) become positive.

$$
\begin{aligned}
\theta^* &= \theta^{(t-1)} + \sum_{h=0}^{\infty} J_\eta^h (\theta_\eta^{(t-1)} - \theta^{(t-1)}) \\
&= \theta^{(t-1)} + (I - J_\eta^2)^{-1} (\theta_\eta^{(t)} - \theta^{(t-1)}).
\end{aligned}
$$

Comparison of TJ²pEM & TJpEM

$$\theta^{(t+1)} = \theta^{(t-1)} + \frac{1}{1-(\gamma_\eta^{(t)})^2}(\theta_\eta^{(t)} - \theta^{(t-1)})$$ (TJ²pEM)

$$\theta^{(t+1)} = \theta^{(t)} + (1-\gamma_\eta^{(t)})^{-1}(\theta_\eta^{(t)} - \theta^{(t)})$$ (TJpEM)

- $\gamma_\eta^{(t)} s$ are identical.

- TJpEM extrapolates from $\theta^{(t)}$, while TJ²pEM from $\theta^{(t-1)}$.

TJ^{2}pEM Algorithm

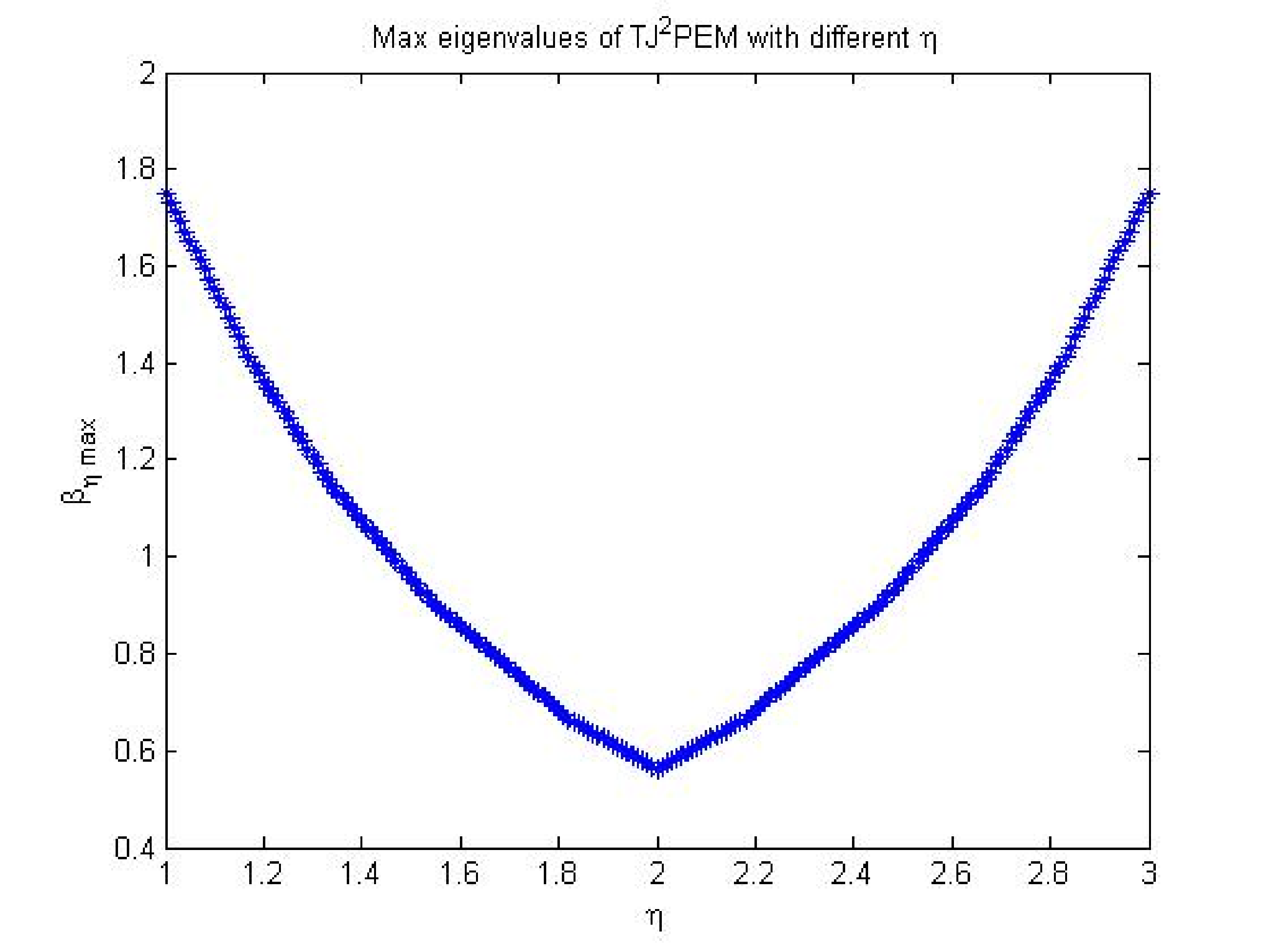
- ◆Triple Jump Framework
- ◆Candidate 1: TJ^{2}pEM Extrapolation
- ◆Candidate 2: pEM Extrapolation
- ◆Candidate 3: EM

Convergence Rate of TJ^{2}pEM

- ◆The i-th eigenvalue of TJ^{2}pEM is:

$$\beta_{\eta i} = 1 - \frac{1}{1-(\gamma_\eta^{(t)})^2} + \frac{1}{1-(\gamma_\eta^{(t)})^2}(\lambda_{\eta i})^2 = \frac{(\lambda_{\eta i})^2 - (\gamma_\eta^{(t)})^2}{1-(\gamma_\eta^{(t)})^2}.$$

Convergence Rates of TJ^{2}pEM with Different Learning Rates



TJ^2aEM Algorithm

- TJ^2pEM with dynamic learning rates (similar to aEM)
- η iterates among 1.2, 1.4, 1.6 and 1.8.

Why Dynamic Learning Rates

- ◆From our experiments, we found that aEM outperforms pEM with the optimal learning rate.
- ◆We can prove that pEM with dynamic learning rates can accelerate pEM with the optimal learning rate.

Proof Sketch

- Assume that we use two learning rates:

$$\eta^{(1)} = \eta^* + \Delta \text{ and } \eta^{(2)} = \eta^* - \Delta$$

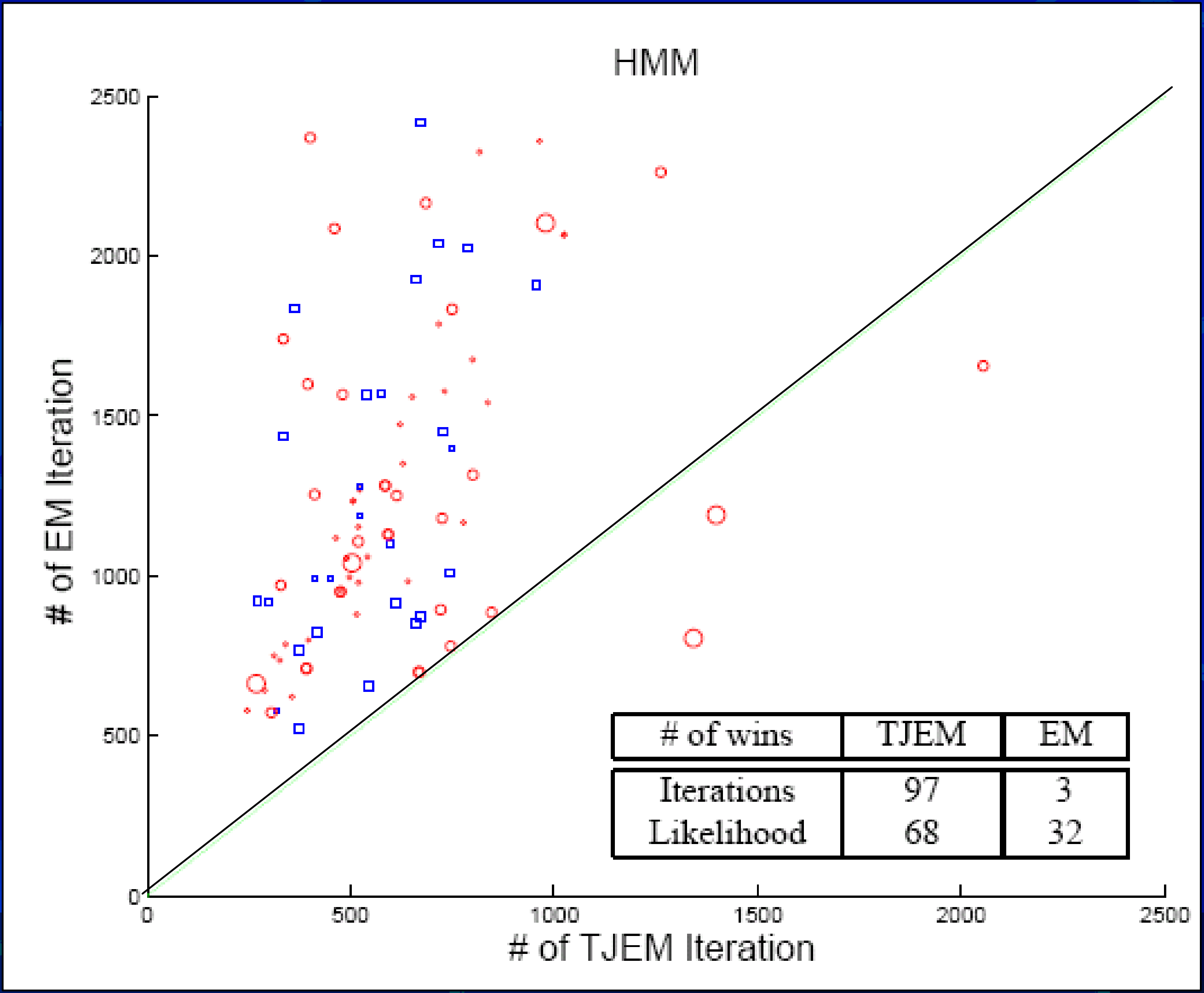
- The eigenvalues of dynamic learning rates are smaller than their counterparts of the optimal learning rate

$$\begin{aligned}
& (1-\eta^{(1)}+\eta^{(1)}\lambda_i)(1-\eta^{(2)}+\eta^{(2)}\lambda_i) \\
= \; & (1-\eta^*+\eta^*\lambda_i-\Delta(1-\lambda_i))(1-\eta^*+\eta^*\lambda_i+\Delta(1-\lambda_i)) \\
= \; & (\lambda_{\eta^*i}-\Delta(1-\lambda_i))(\lambda_{\eta^*i}+\Delta(1-\lambda_i)) \\
= \; & (\lambda_{\eta^*i})^2-(\Delta(1-\lambda_i))^2 \\
\leq \; & (\lambda_{\eta^*i})^2.
\end{aligned}$$

Data sets

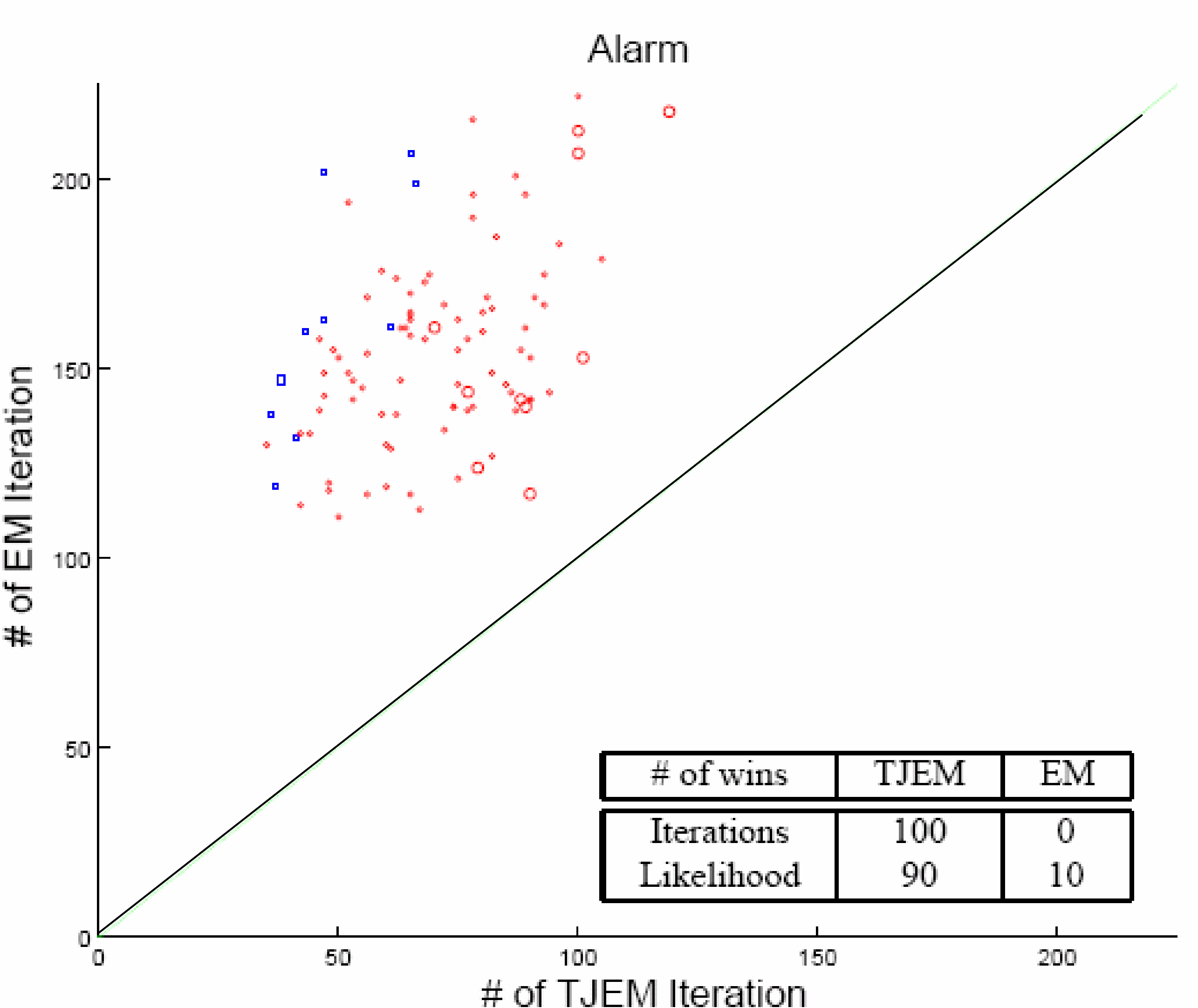
- **100 synthesized data sets for each model**
  - HMM: 5-state, 20-symbol, 500 sequences with length 100
  - Bayesian net (ALARM) *[Cooper & Herskovitz]*: 2,000 cases with different missing rates for all random variables
  - GMM: 5 equal-weight Gaussian of means= {(0,0), (1,0), (-1,0), (0,1), (0,-1)} and var = 0.8. 2,000 cases.
  - Semisupervised Bayesian classifier: 5-class, 100 10-state features, 3,000 cases with unequal missing rates

TJEM Faster than EM (HMM)



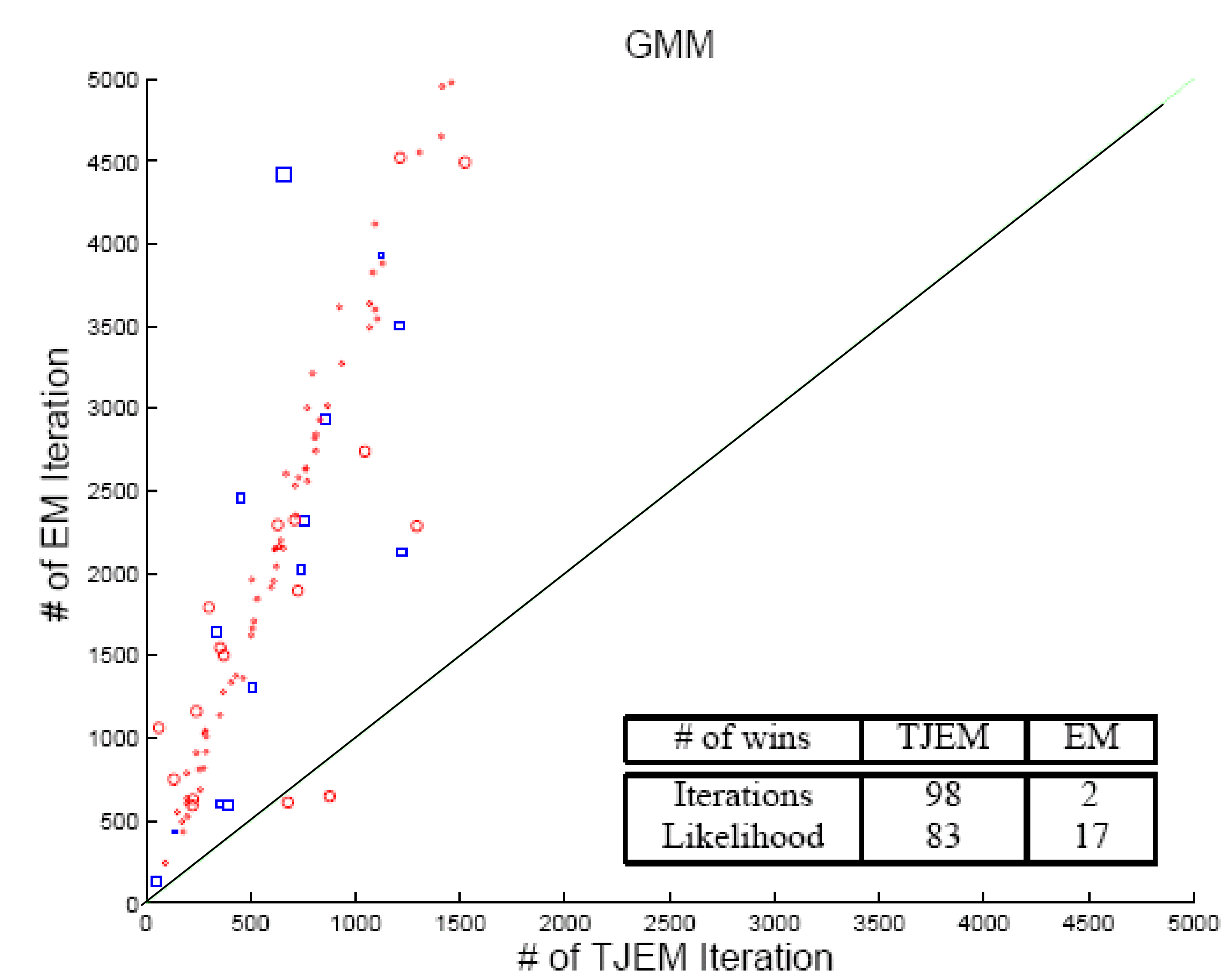
# of wins	TJEM	EM
Iterations	97	3
Likelihood	68	32

TJEM Faster than EM (Alarm)



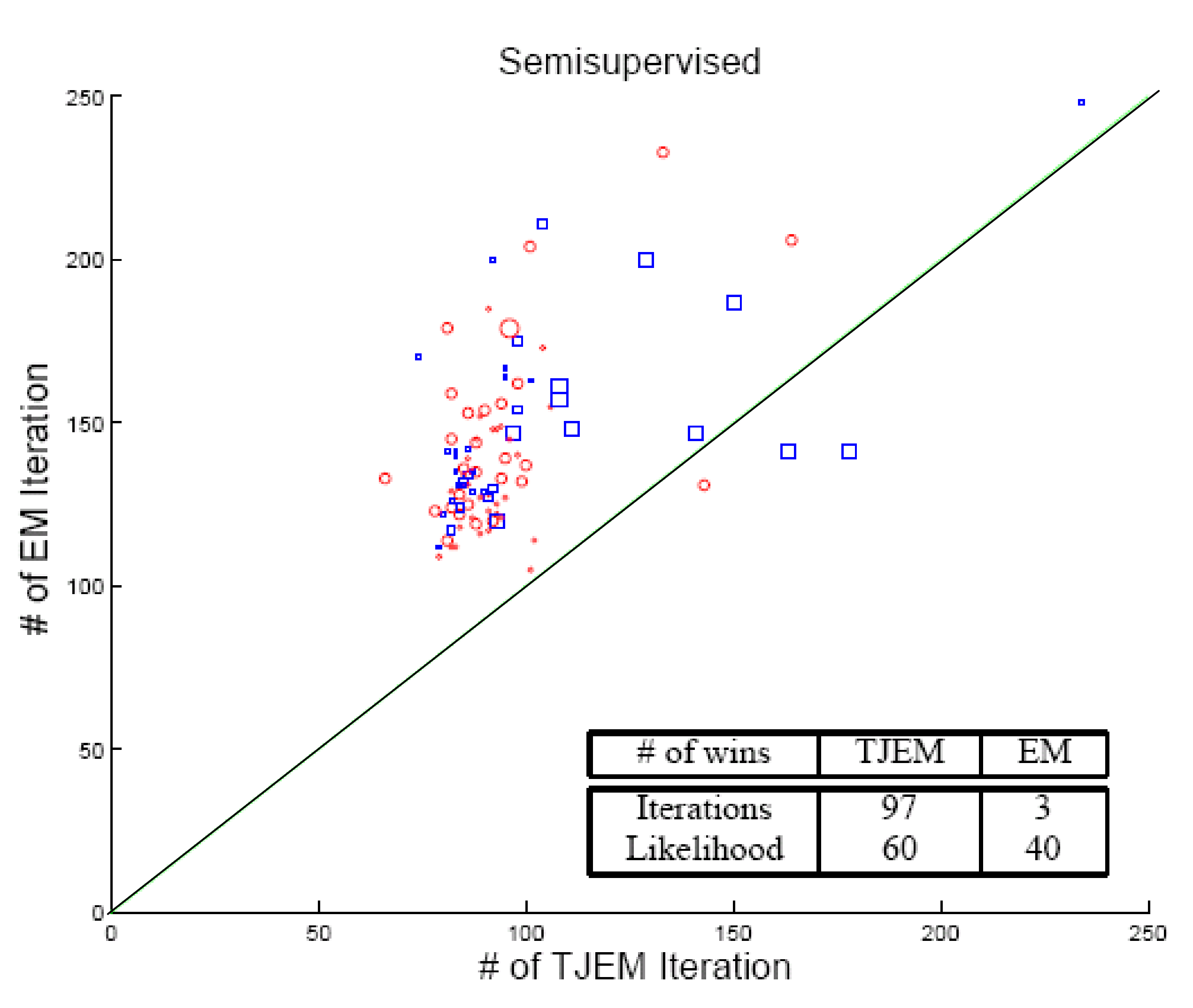
# of wins	TJEM	EM
Iterations	100	0
Likelihood	90	10

TJEM Faster than EM (GMM)



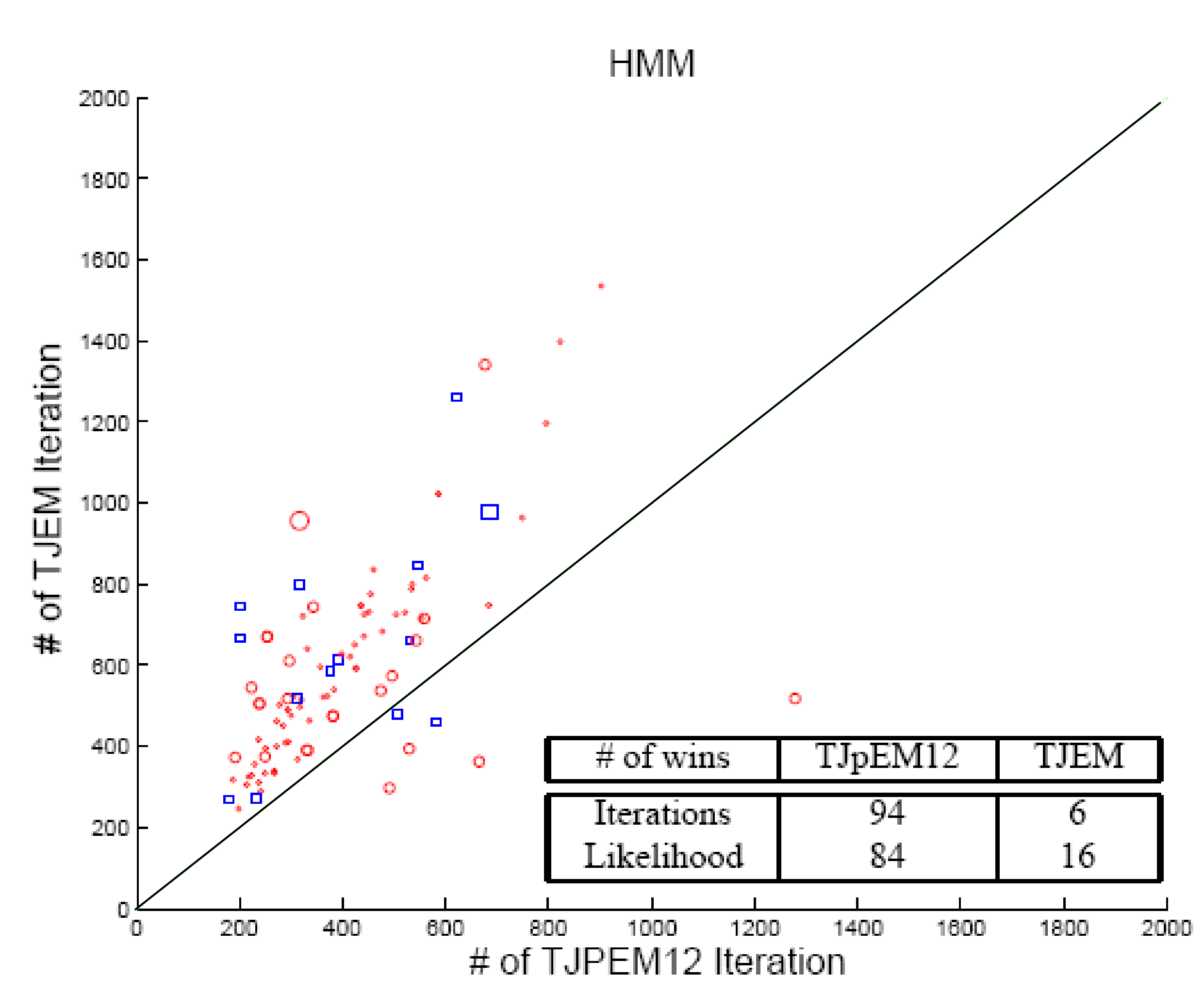
# of wins	TJEM	EM
Iterations	98	2
Likelihood	83	17

TJEM Faster than EM (Bayesian Classifier)



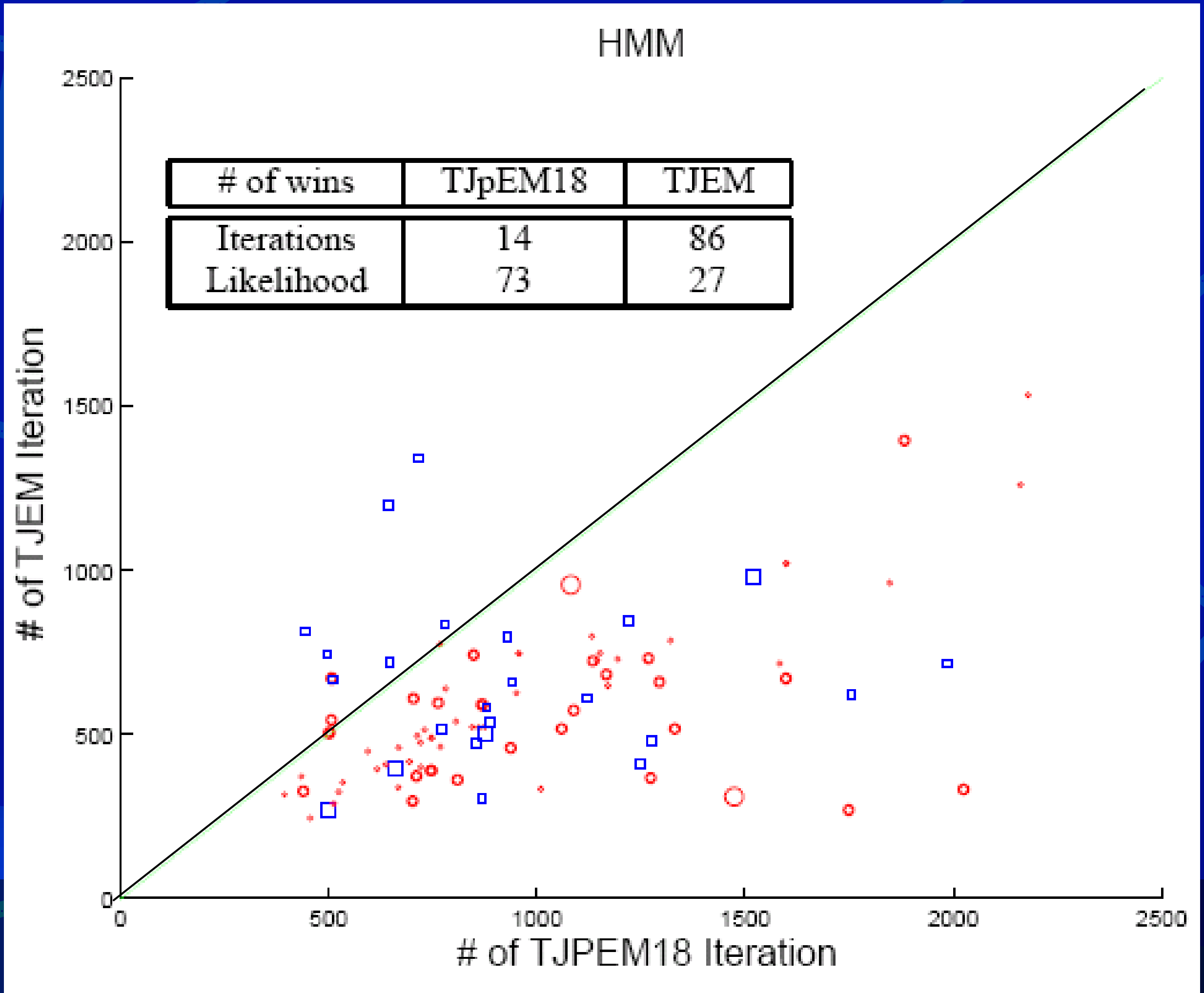
# of wins	TJEM	EM
Iterations	97	3
Likelihood	60	40

TJpEM with Proper Learning Rate Faster than TJEM



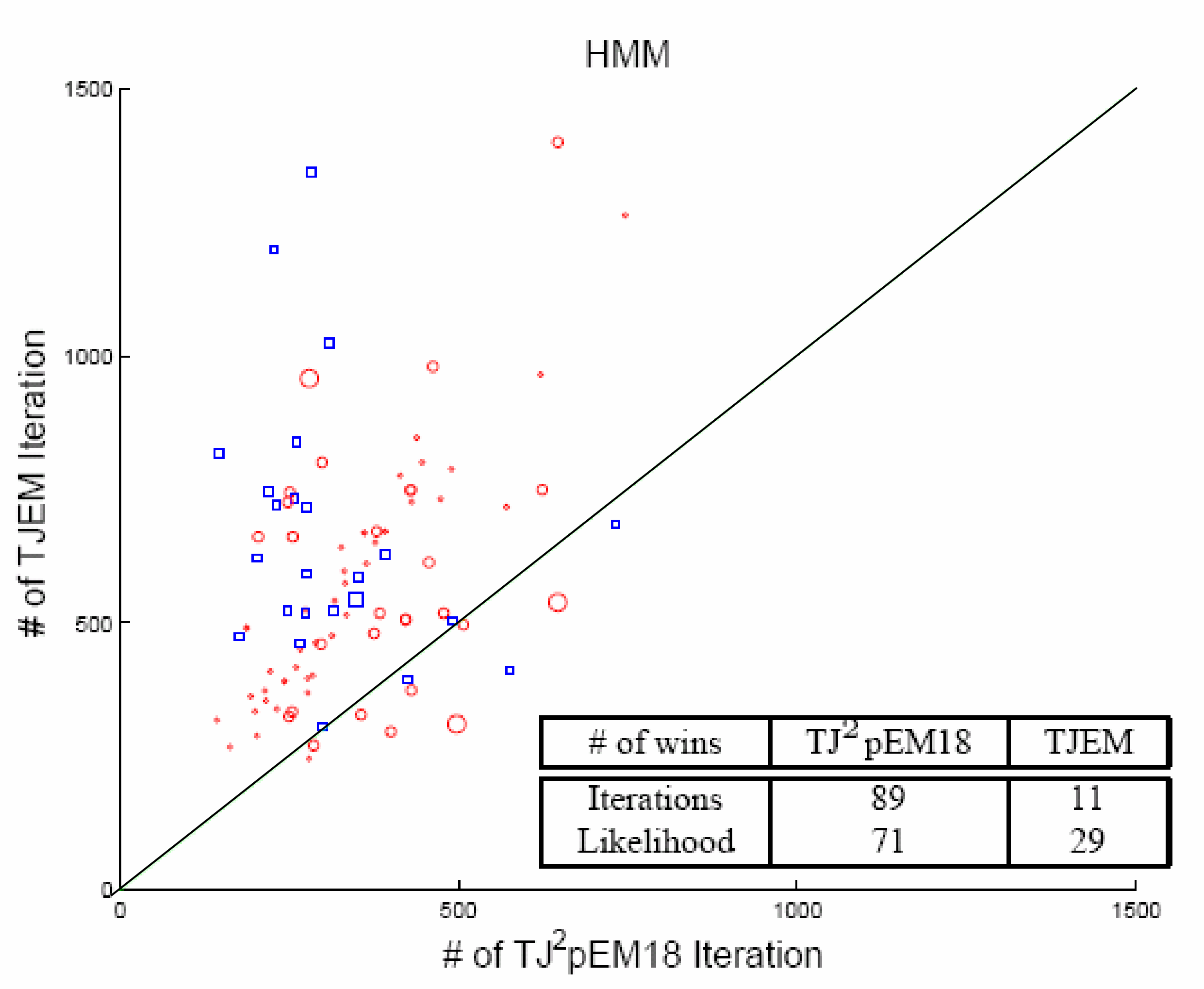
# of wins	TJpEM12	TJEM
Iterations	94	6
Likelihood	84	16

TJpEM with Large Learning Rates Slower than TJEM



# of wins	TJpEM18	TJEM
Iterations	14	86
Likelihood	73	27

TJ^2pEM Overcomes the Impact of Large Learning Rates

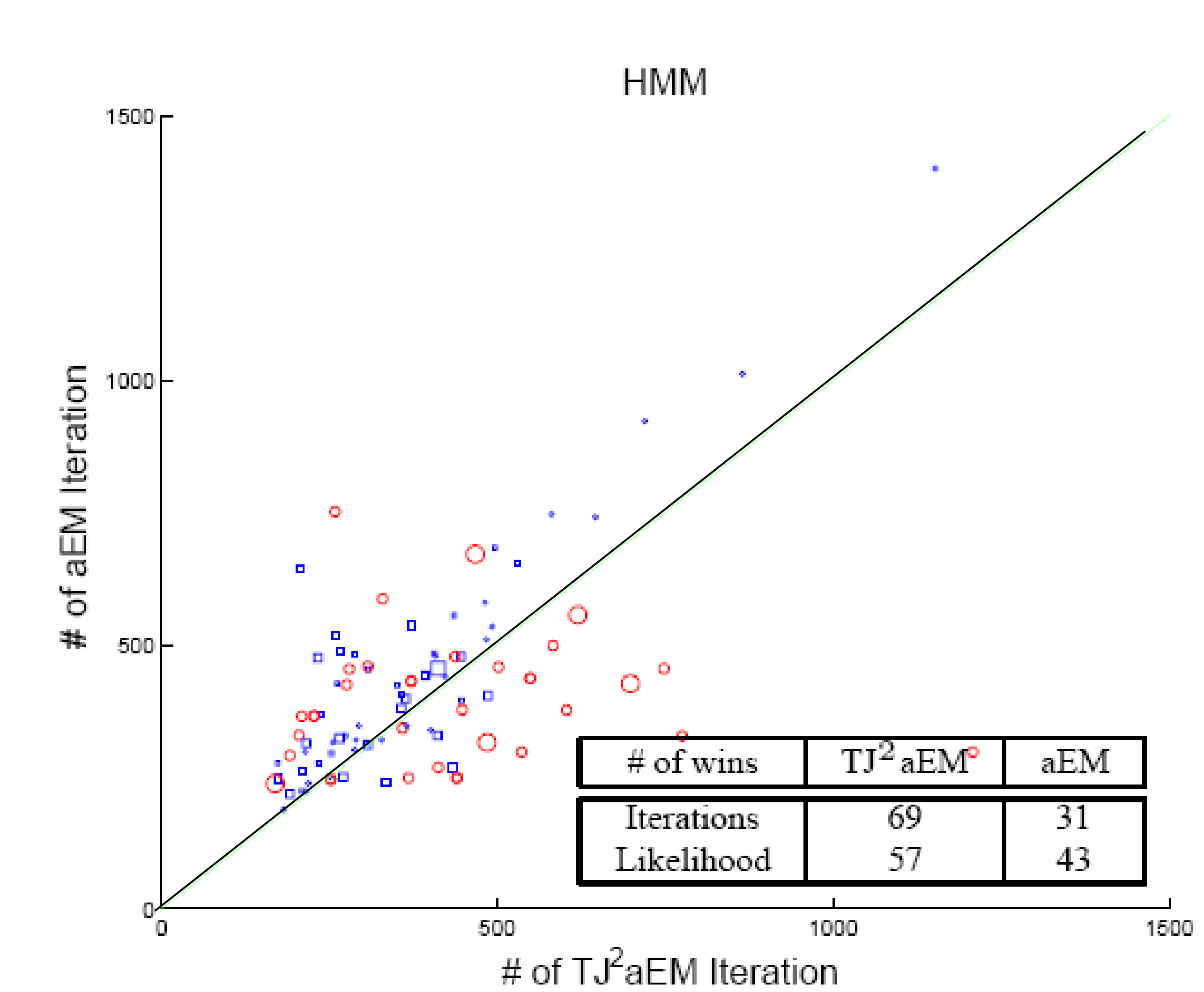


# of wins	TJ^2pEM18	TJEM
Iterations	89	11
Likelihood	71	29

aEM Faster than pEM

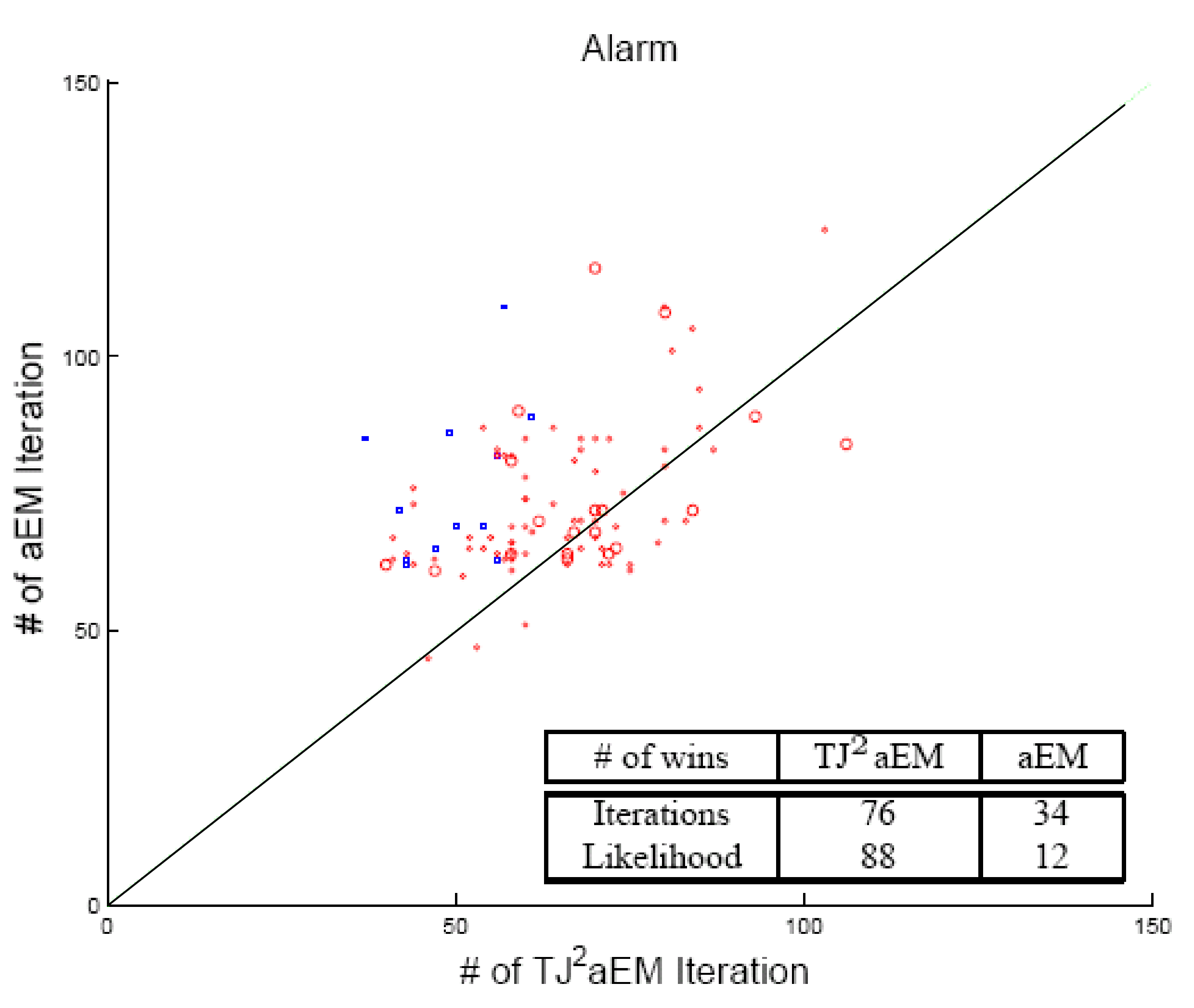
- GMM
- Empirically find that $\eta^* = 1.96$
- pEM with η^* converged in 1,327 iterations.
- aEM in 766 iterations
- TJ^{2}aEM in 527 iterations

TJ^{2}aEM Faster than aEM (HMM)



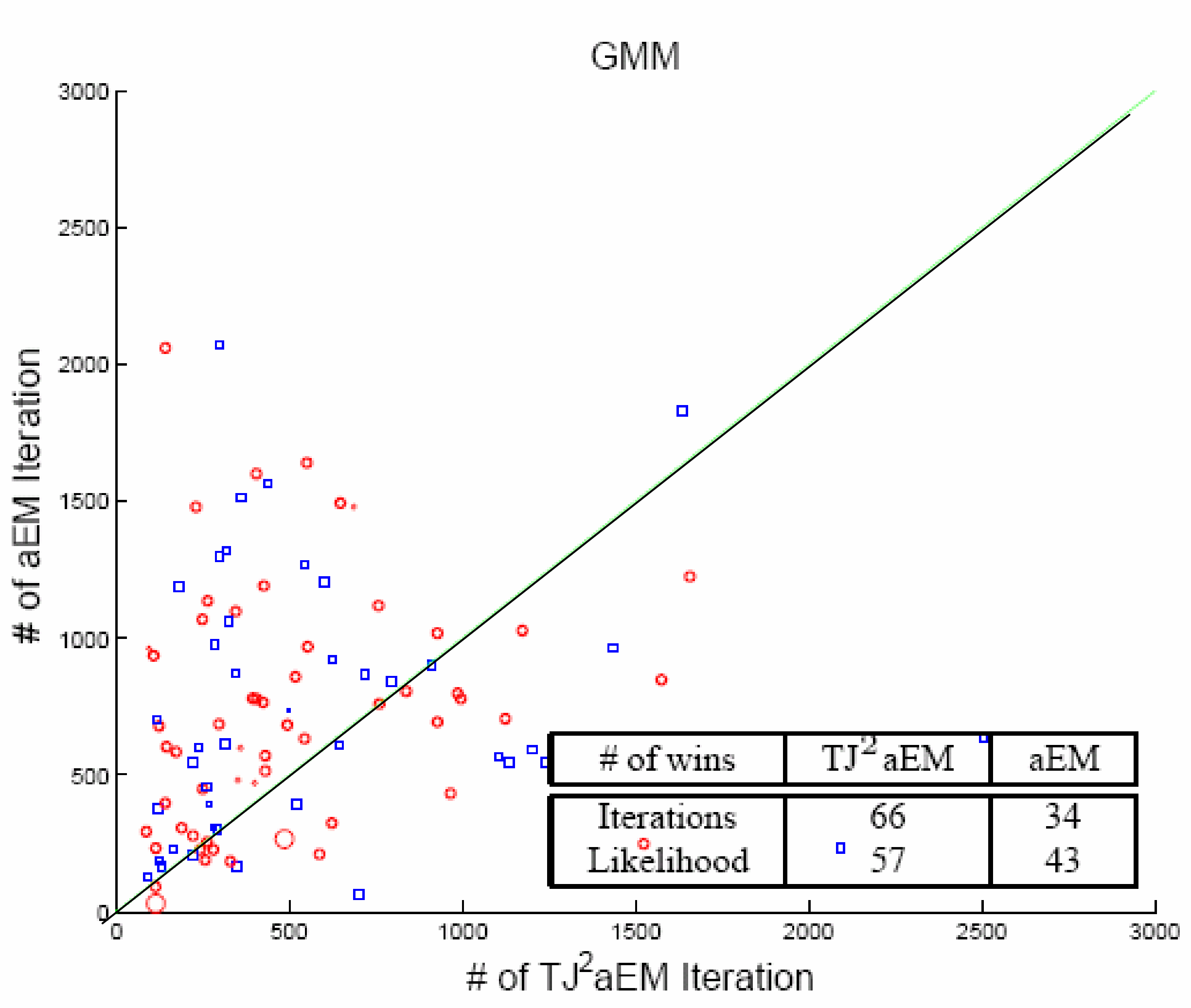
# of wins	TJ^{2}aEM	aEM
Iterations	69	31
Likelihood	57	43

TJ^2aEM Faster than aEM (Alarm)



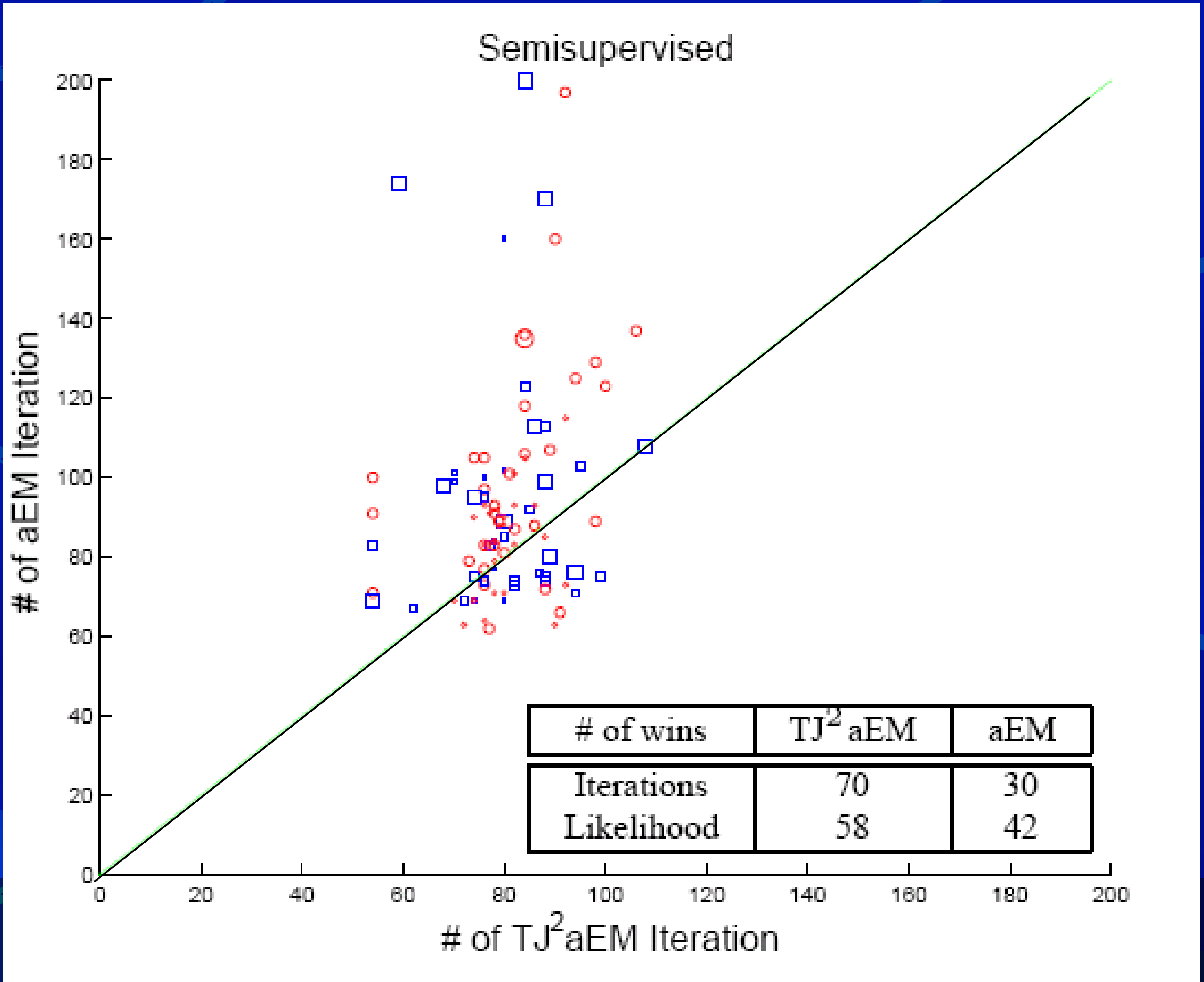
# of wins	TJ^2 aEM	aEM
Iterations	76	34
Likelihood	88	12

TJ²aEM Faster than aEM (GMM)



# of wins	TJ²aEM	aEM
Iterations	66	34
Likelihood	57	43

TJ^{2}aEM Faster than aEM (Bayesian Classifier)



# of wins	TJ^{2}aEM	aEM
Iterations	70	30
Likelihood	58	42

Componentwise TJEM

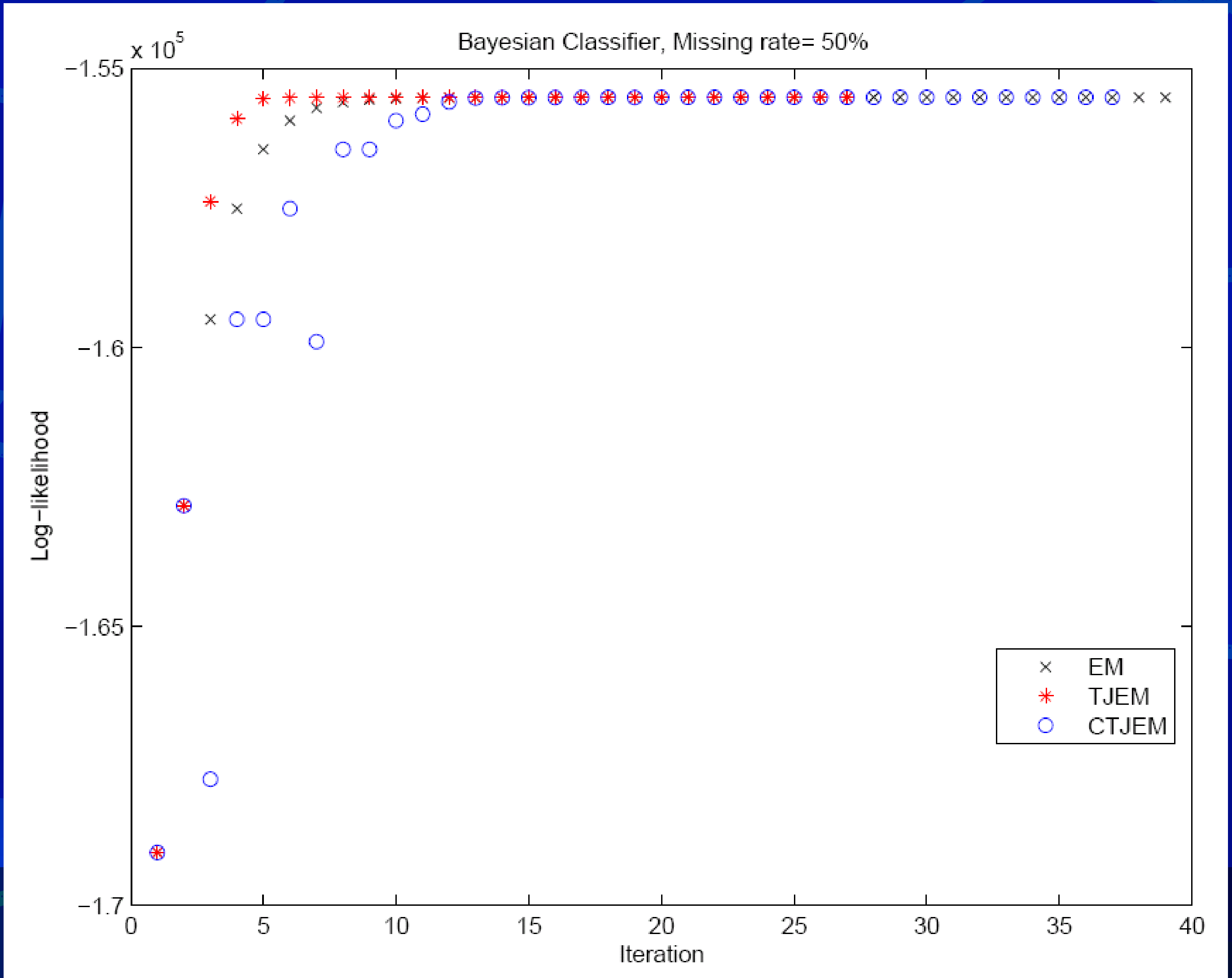
- We can further accelerate previous TJ algorithms when the Jacobian of M is close to a diagonal or block diagonol matrix by using different approximation for each block.

$$\begin{pmatrix} B_{11} & 0 & \cdots & 0 \\ 0 & B_{12} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & B_{ij} \end{pmatrix}$$

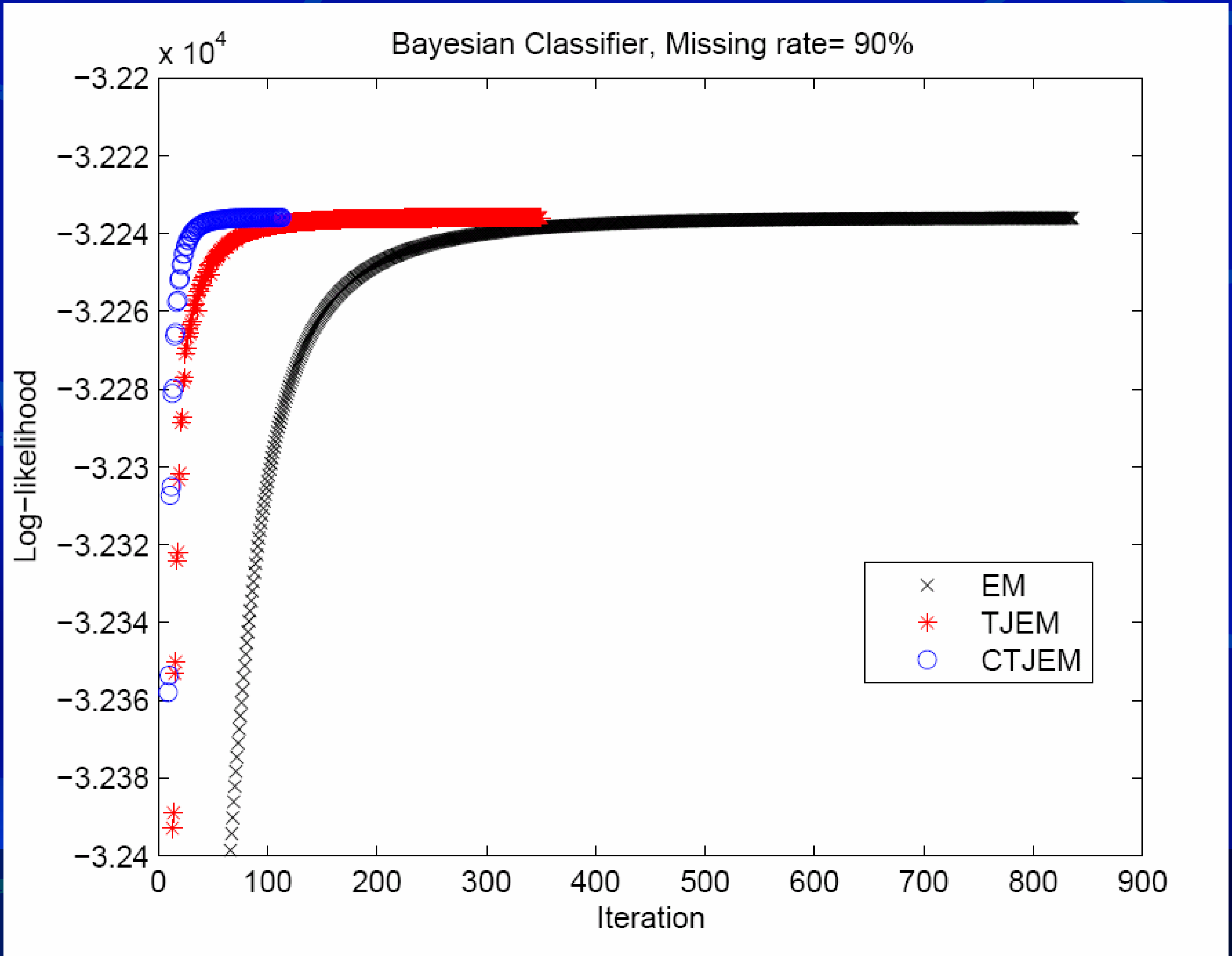
Case Study: Bayesian Classifier

- With the increase of missing values, the Jacobian of EM for a semisupervised Bayesian classifier is closer to block diagonal matrix.

Missing Rate = 50%



Missing Rate = 90%



Summary

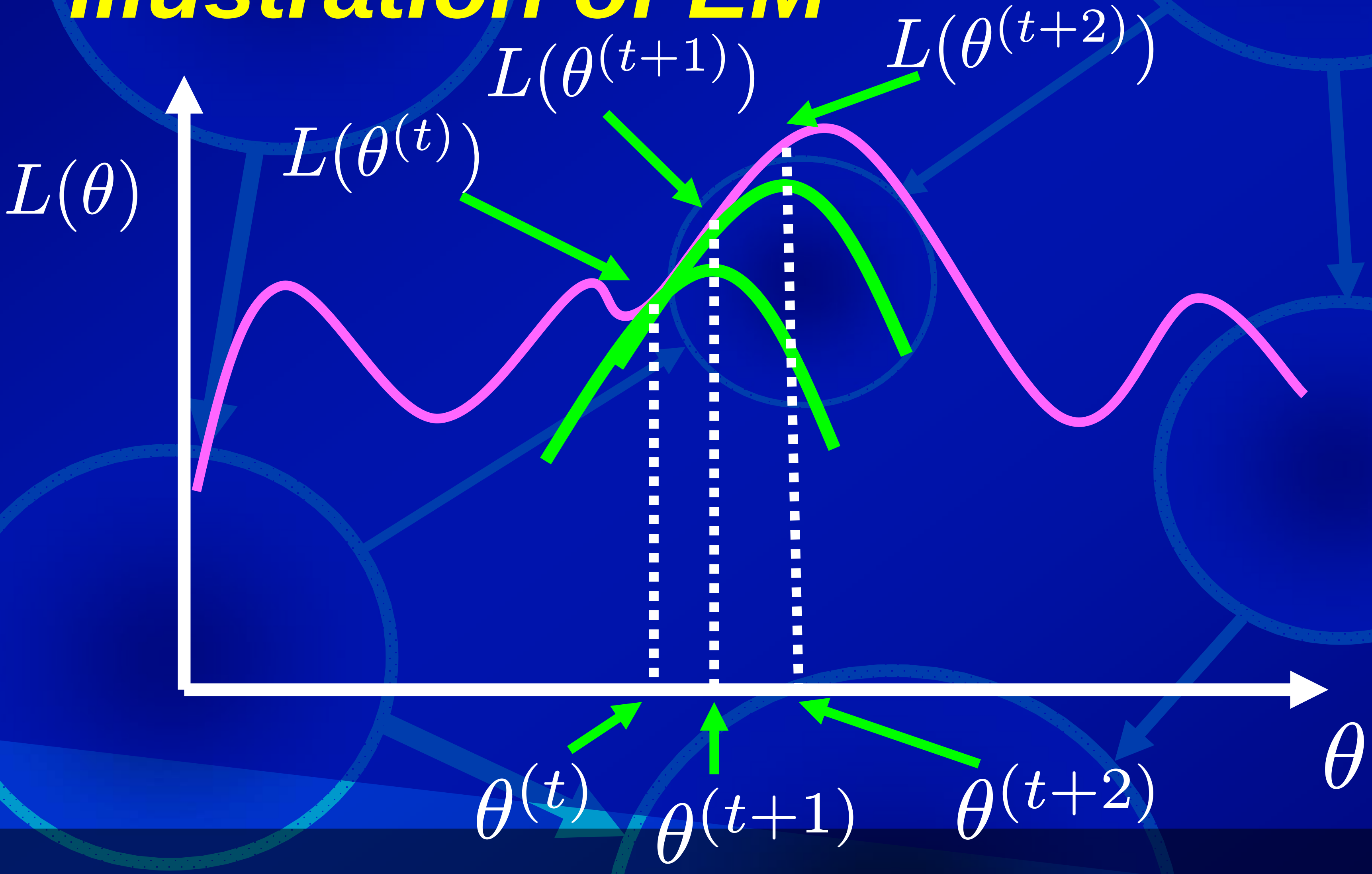
- ◆Triple jump framework to integrate EM and its extrapolation-based variants
- ◆Improving convergence rate from TJEM, TJpEM, TJ^{2}pEM, to TJ^{2}aEM
- ◆CTJEM for sparse data sets where the Jacobian might be close to block diagonal.

Thank You

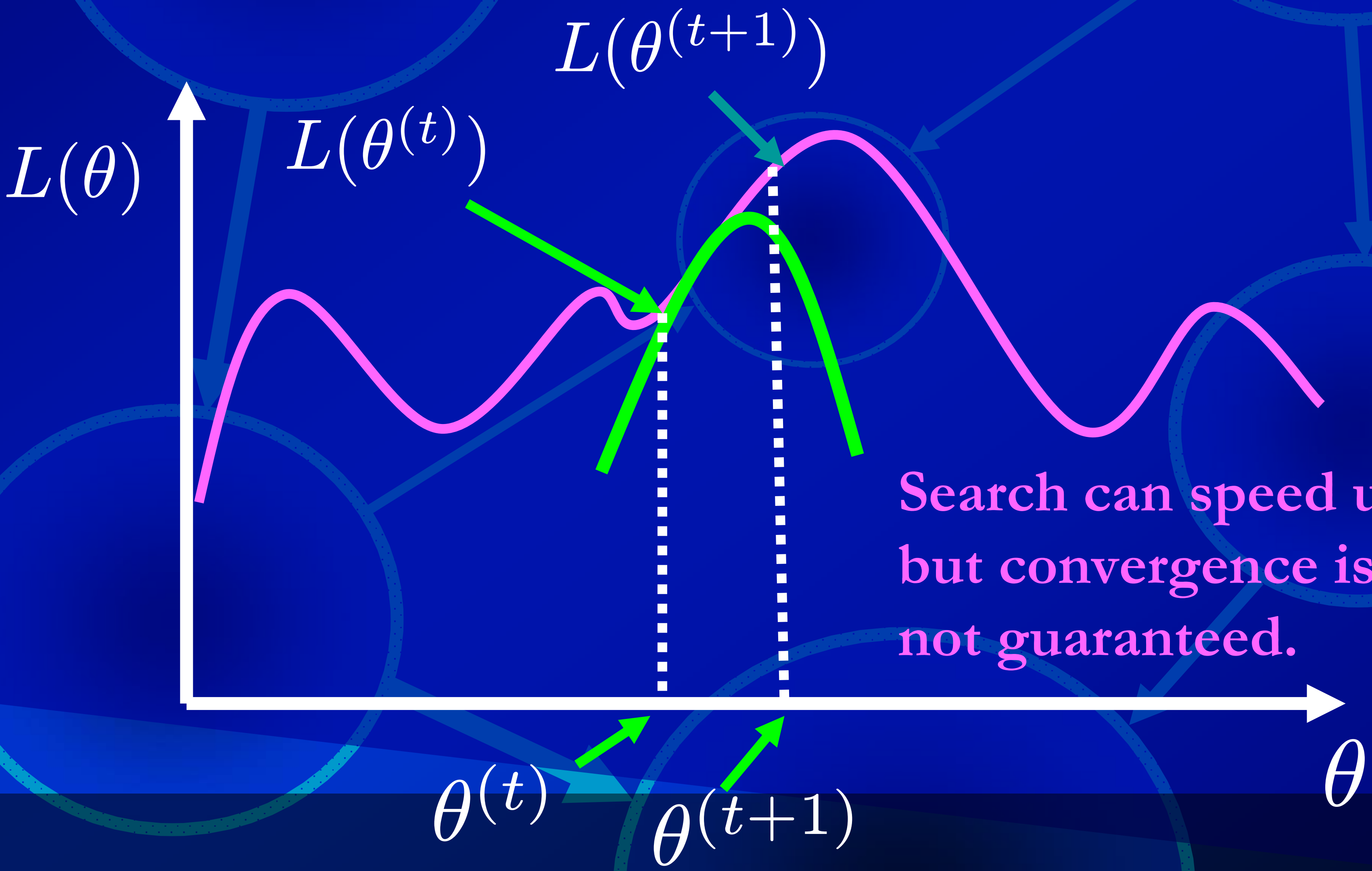
References

- E. Bauer, D. Koller, & Y.Singer. Update rules for parameter estimation in Bayesian networks. *UAI97,* pages 3—13, 1997.
- G. F. Cooper & E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309—347, 1992
- A. Dempster, N. Laird, & D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, Series B, 39(1):1-38, 1977.
- T. Hesterberg. Staggered Aitken acceleration for EM. In *Proceedings of the Statistical Computing Section of the American Statistical Association*, 2005.
- T. A. Louis. Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 44:226—233, 1982.
- G. J. McLachlan & Thriyambakam Krishnan. The EM algorithm and extensions. Wiley-Interscience, 1997.
- R. Salakhutdinov & S. Roweis. Adaptive overrelaxed bound optimization methods. *ICML03,* pages 664—671, 2003.

Illustration of EM



Extrapolation-based Variants

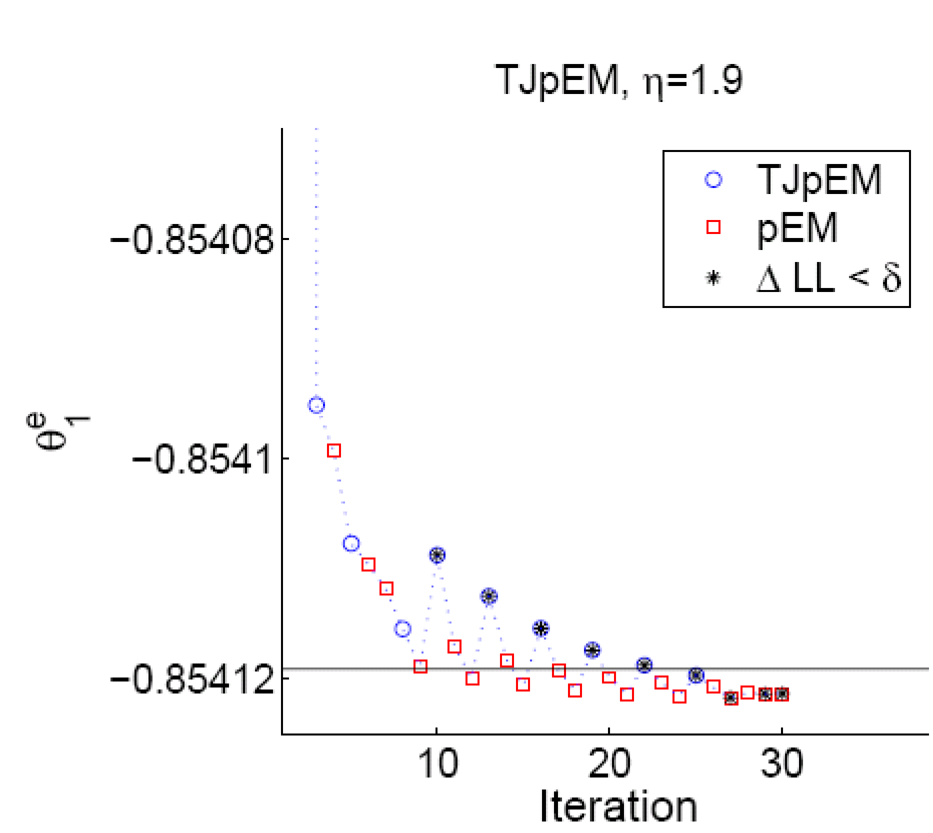


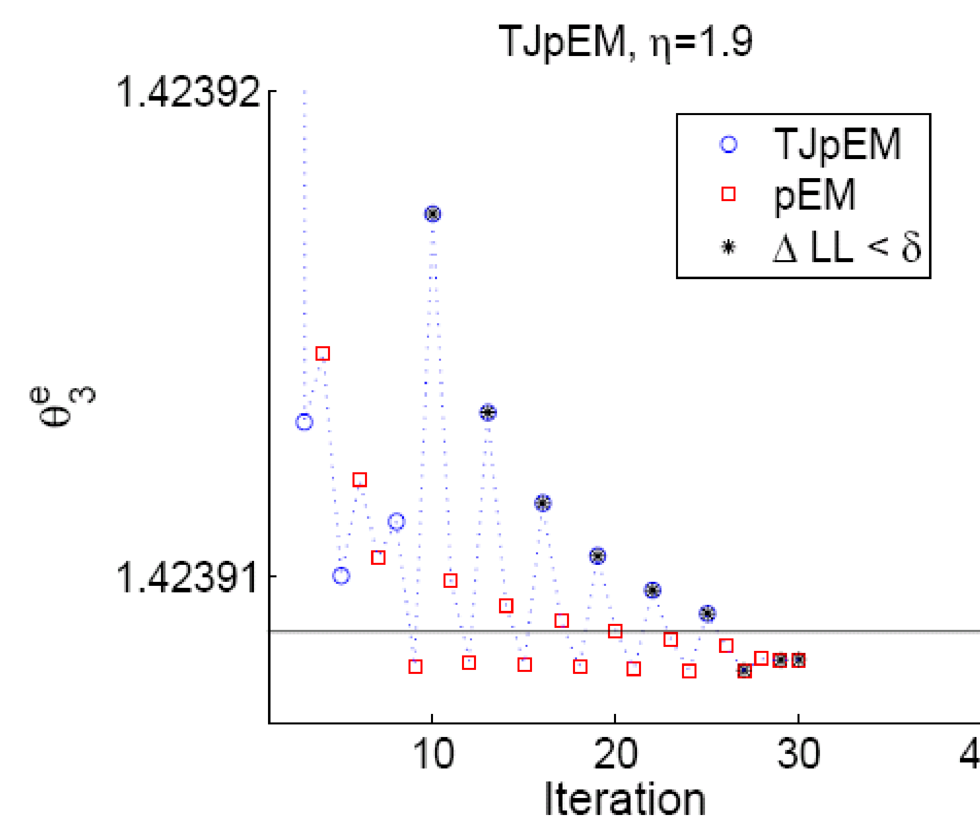
Search can speed up but convergence is not guaranteed.

Advantage of TJ^{2}pEM with Large Learning Rates

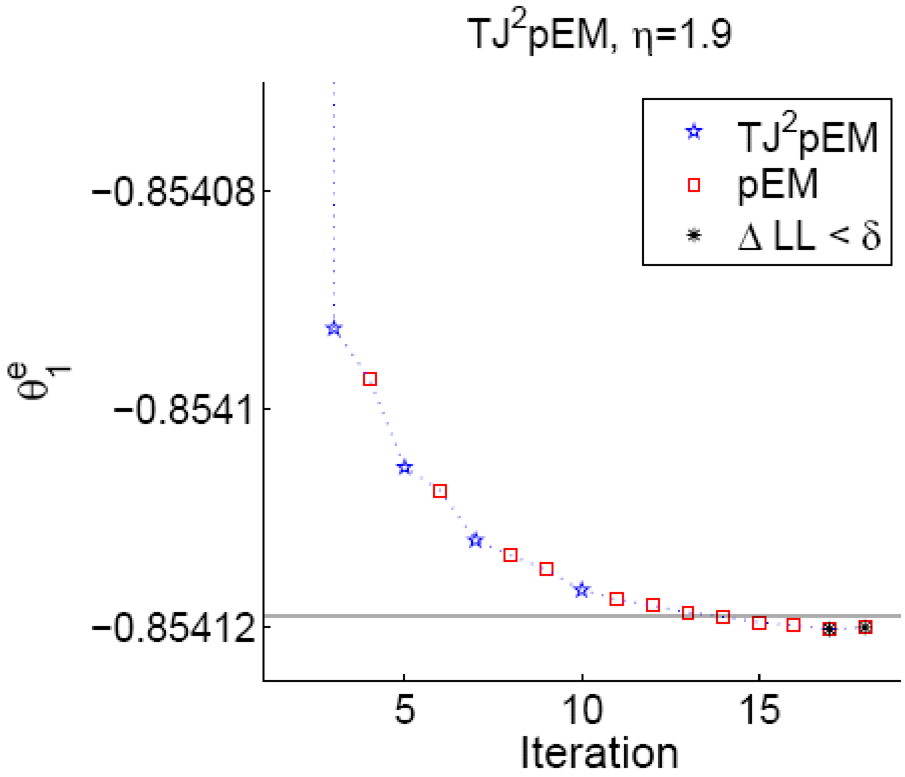
- Toy model: Mixture of two 1-dimensional Gaussian with fixed variance.
- 500 training examples
- Parameter vector: (p_0 , μ_1, μ_2)
- J can be estimated *[Louis 1982]*.
- Eigenvalues: (0.78, 0.31, 0.26)
- Apply ´ $= 1.9$: (0.58, -0.31, -0.41)

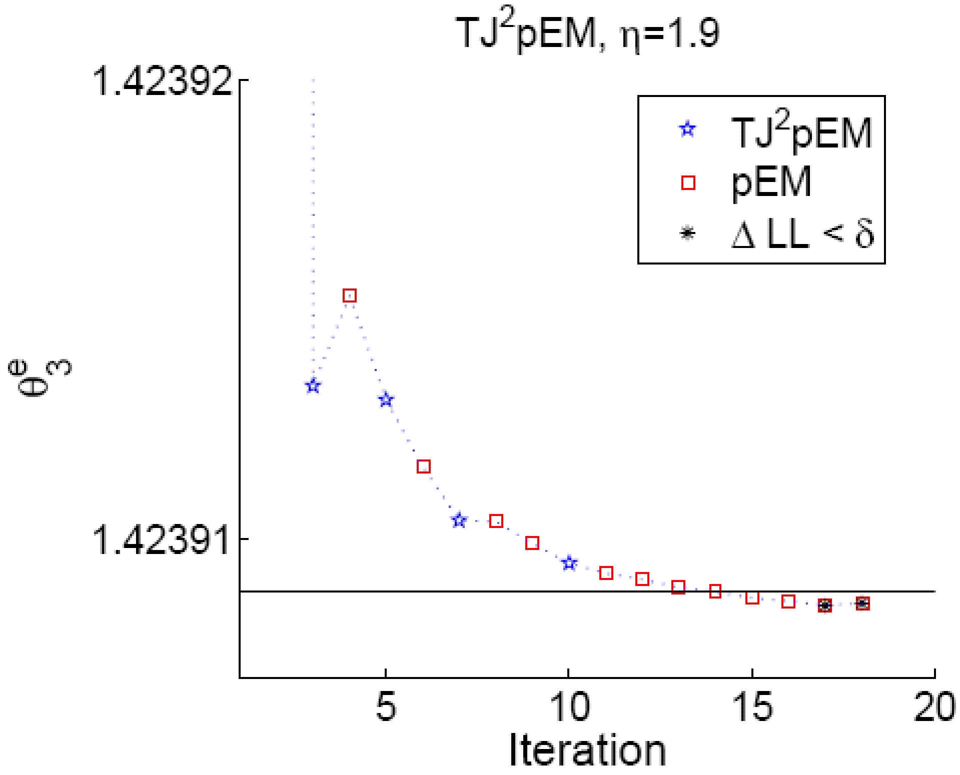
Parameter Vector on the Eigenspace by TJpEM





Parameter Vector on the Eigenspace by TJ^{2}pEM


TJ^{2}pEM, η=1.9
θ^e_1
Iteration
TJ^{2}pEM
pEM
Δ LL < δ


TJ^{2}pEM, η=1.9
θ^e_3
Iteration
TJ^{2}pEM
pEM
Δ LL < δ