Non-Asymptotic Convergence Rate of EM, and Improved Expectation Maximization Algorithms

Reza Babanezhad

Joint work with Raunak Kumar and Mark Schmidt
University of British Columbia

rezababa@cs.ubc.ca

Computer Science

Learning with Missing Values

- Missing values are very common in real datasets.
- For example, we could have a dataset like this:

$$X = \begin{bmatrix} N & 33 & 5 \\ L & 10 & 1 \\ F & ? & 2 \\ M & 22 & 0 \end{bmatrix}, y = \begin{bmatrix} -1 \\ +1 \\ -1 \\ ? \end{bmatrix}.$$

- We often want to learn with unobserved/missing/hidden/latent values.

Learning with Missing Values

- Missing values are very common in real datasets.
- For example, we could have a dataset like this:

$$X = \begin{bmatrix} N & 33 & 5 \\ L & 10 & 1 \\ F & ? & 2 \\ M & 22 & 0 \end{bmatrix}, y = \begin{bmatrix} -1 \\ +1 \\ -1 \\ ? \end{bmatrix}.$$

- We often want to learn with unobserved/missing/hidden/latent values.

- We'll focus on data that is missing at random (MAR):
  - Assume that the reason ? is missing does not depend on the missing value.

Expectation Maximization: Optimization with MAR Variables

- Expectation maximization (EM) is an optimization algorithm for MAR values:
  - Applies to problems that are easy to solve with "complete" data (i.e., you knew ?).

Expectation Maximization: Optimization with MAR Variables

- Expectation maximization (EM) is an optimization algorithm for MAR values:
  - Applies to problems that are easy to solve with "complete" data (i.e., you knew ?).
  - Based on probabilistic or "soft" assignments to MAR variables.
  - For many problems it leads to simple closed-form updates.

Expectation Maximization: Optimization with MAR Variables

- Expectation maximization (EM) is an optimization algorithm for MAR values:
  - Applies to problems that are easy to solve with "complete" data (i.e., you knew ?).
  - Based on probabilistic or "soft" assignments to MAR variables.
  - For many problems it leads to simple closed-form updates.

- EM is among the most cited papers across all fields (around 54,000 citations).

- Some common applications:
  - Filling in missing data.
  - Semi-supervised learning.

Expectation Maximization: Optimization with MAR Variables

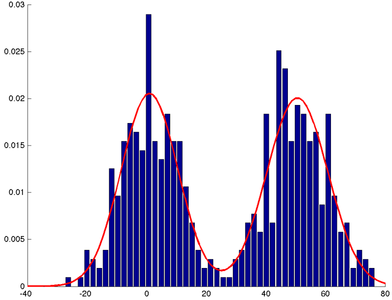
- Expectation maximization (EM) is an optimization algorithm for MAR values:
  - Applies to problems that are easy to solve with "complete" data (i.e., you knew ?).
  - Based on probabilistic or "soft" assignments to MAR variables.
  - For many problems it leads to simple closed-form updates.

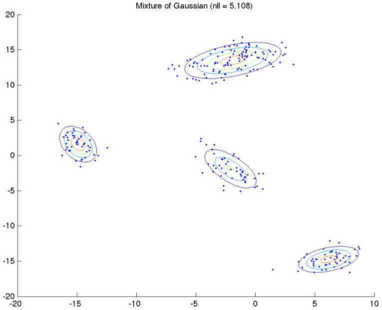
- EM is among the most cited papers across all fields (around 54,000 citations).

- Some common applications:
  - Filling in missing data.
  - Semi-supervised learning.
  - Mixture of Gaussians.
  - Hidden Markov models.
- In the two latter problems, statisticians introduce MAR variables to use EM.

Example: Mixture of Gaussians

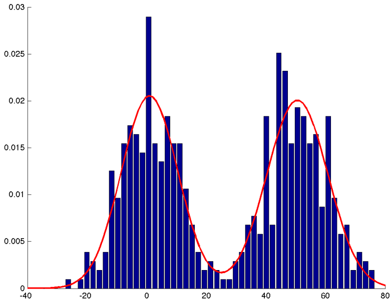
- Application: modeling multi-modal data with mixture of Gaussians

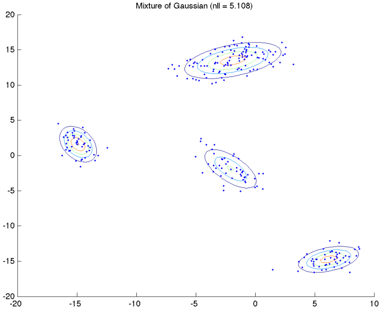




Example: Mixture of Gaussians

- Application: modeling multi-modal data with mixture of Gaussians

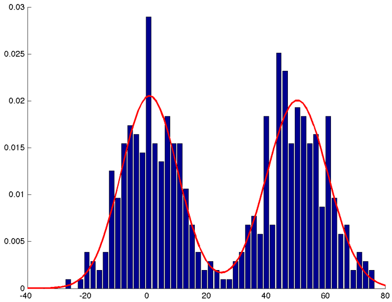


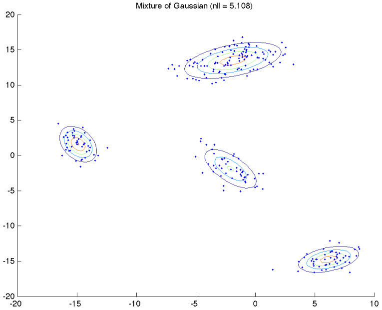


- We introduce an MAR variable for each sample, represent "which Gaussian it came from".
  - EM updates just compute weighted mean and variance of data based on these values.

Example: Mixture of Gaussians

- Application: modeling multi-modal data with mixture of Gaussians

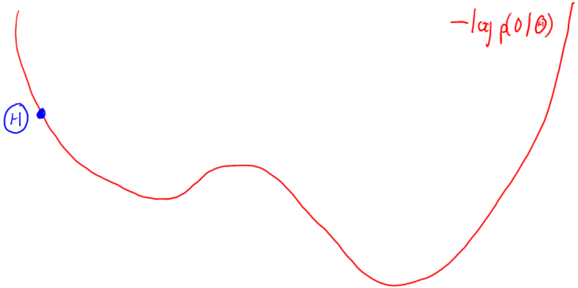




- We introduce an MAR variable for each sample, represent "which Gaussian it came from".
  - EM updates just compute weighted mean and variance of data based on these values.

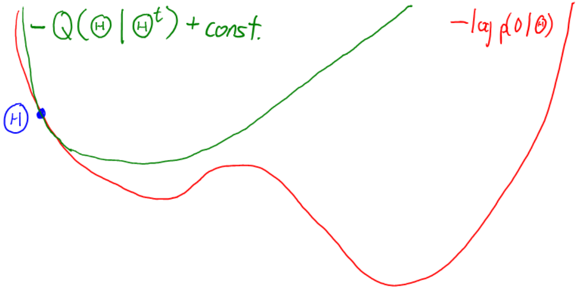
- As in typical applications of EM, the problem is highly non-convex.

- Expectation maximization is a "bound-optimization" method:
  - At each iteration t we optimize a bound on the function.



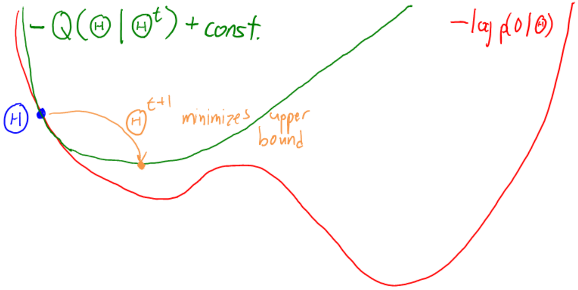
Expectation Maximization (Picture Version)

- Expectation maximization is a "bound-optimization" method:
  - At each iteration t we optimize a bound on the function.



Expectation Maximization (Picture Version)

- Expectation maximization is a "bound-optimization" method:
  - At each iteration t we optimize a bound on the function.



- Unlike gradient descent and Newton, the "surrogate" Q is not quadratic.
- In EM, our bound comes from expectation over hidden variables.

Expectation Maximization (Equation Version)

- We want to maximize likelihood of data X with MAR values z, and parameters λ,

$$\operatorname*{arg\,min}_{\lambda \in \Lambda} p(X \mid \lambda) = \sum_{z \in \mathcal{Z}} p(X, z \mid \lambda),$$

where I'm assuming z is discrete (you use an integral for continuous z).

Expectation Maximization (Equation Version)

- We want to maximize likelihood of data X with MAR values z, and parameters λ,

$$\underset{\lambda \in \Lambda}{\arg\min}\, p(X \mid \lambda) = \sum_{z \in \mathcal{Z}} p(X, z \mid \lambda),$$

  where I'm assuming z is discrete (you use an integral for continuous z).
- Instead of maximizing likelihood, we can equivalently minimize negative log-likelihood,

$$f(\lambda) = -\log\left\{\sum_{z \in \mathcal{Z}} p(X, z \mid \lambda)\right\}.$$

Expectation Maximization (Equation Version)

- We want to maximize likelihood of data X with MAR values z, and parameters λ,

$$\underset{\lambda \in \Lambda}{\arg\min}\, p(X \mid \lambda) = \sum_{z \in \mathcal{Z}} p(X, z \mid \lambda),$$

  where I'm assuming z is discrete (you use an integral for continuous z).
- Instead of maximizing likelihood, we can equivalently minimize negative log-likelihood,

$$f(\lambda) = -\log\left\{\sum_{z \in \mathcal{Z}} p(X, z \mid \lambda)\right\}.$$

- Unfortunately, this has a sum inside the log:
  - This will be non-convex even in common settings where $-\log p(X, z \mid \lambda)$ is convex.
  - This won't have closed-form solution even in common settings where minimizer given z does.

Expectation Maximization (Equation Version)

- At each iteration k, the expectation maximization algorithm optimizes a surrogate g_k,

$$\begin{aligned} g_k(\lambda) &= \mathbb{E}_{z \mid X,\, \lambda^k}[p(z \mid X, \lambda)] + \text{const.} \\ &= \sum_{z \in \mathcal{Z}} p(z \mid X, \lambda^k) \log p(X, z \mid \lambda) + \text{const.}, \end{aligned}$$

the expected negative log-probabilities under the current guess of the parameters λ^k.

Expectation Maximization (Equation Version)

- At each iteration k, the expectation maximization algorithm optimizes a surrogate g_k,

$$
\begin{aligned}
g_k(\lambda) &= \mathbb{E}_{z \mid X, \lambda^k}[p(z \mid X, \lambda)] + \text{const.} \\
&= \sum_{z \in \mathcal{Z}} p(z \mid X, \lambda^k) \log p(X, z \mid \lambda) + \text{const.},
\end{aligned}
$$

the expected negative log-probabilities under the current guess of the parameters λ^k.

  - This has a closed-form solution in cases where knowing z would give a closed-form solution.
  - This is convex if $-\log p(X, z \mid \lambda)$ is convex.

Expectation Maximization (Equation Version)

- At each iteration k, the expectation maximization algorithm optimizes a surrogate g_k,

$$g_k(\lambda) = \mathbb{E}_{z \mid X,\, \lambda^k}[p(z \mid X, \lambda)] + \text{const.}$$
$$= \sum_{z \in \mathcal{Z}} p(z \mid X, \lambda^k) \log p(X, z \mid \lambda) + \text{const.},$$

the expected negative log-probabilities under the current guess of the parameters λ^k.

  - This has a closed-form solution in cases where knowing z would give a closed-form solution.
  - This is convex if $-\log p(X, z \mid \lambda)$ is convex.

- Classic results regarding the relationship between function f and surrogate g:
  - Approximation: the functions g and f agree at λ^k. Formally, $f(\lambda^k) = g_k(\lambda^k)$.

Expectation Maximization (Equation Version)

- At each iteration k, the expectation maximization algorithm optimizes a surrogate g_k,

$$\begin{aligned} g_k(\lambda) &= \mathbb{E}_{z \mid X, \lambda^k}[p(z \mid X, \lambda)] + \text{const.} \\ &= \sum_{z \in \mathcal{Z}} p(z \mid X, \lambda^k) \log p(X, z \mid \lambda) + \text{const.}, \end{aligned}$$

  the expected negative log-probabilities under the current guess of the parameters λ^k.
  - This has a closed-form solution in cases where knowing z would give a closed-form solution.
  - This is convex if $-\log p(X, z \mid \lambda)$ is convex.

- Classic results regarding the relationship between function f and surrogate g:
  - Approximation: the functions g and f agree at λ^k. Formally, $f(\lambda^k) = g_k(\lambda^k)$.
  - Majorization: the function g bounds f from above. Formally, $f(\lambda) \leq g_k(\lambda)$ for all $\lambda \in \Lambda$.

Expectation Maximization (Equation Version)

- At each iteration k, the expectation maximization algorithm optimizes a surrogate g_k,

$$\begin{aligned} g_k(\lambda) &= \mathbb{E}_{z \mid X,\, \lambda^k}[p(z \mid X, \lambda)] + \text{const.} \\ &= \sum_{z \in \mathcal{Z}} p(z \mid X, \lambda^k) \log p(X, z \mid \lambda) + \text{const.}, \end{aligned}$$

  the expected negative log-probabilities under the current guess of the parameters λ^k.
  - This has a closed-form solution in cases where knowing z would give a closed-form solution.
  - This is convex if $-\log p(X, z \mid \lambda)$ is convex.

- Classic results regarding the relationship between function f and surrogate g:
  - Approximation: the functions g and f agree at λ^k. Formally, $f(\lambda^k) = g_k(\lambda^k)$.
  - Majorization: the function g bounds f from above. Formally, $f(\lambda) \leq g_k(\lambda)$ for all $\lambda \in \Lambda$.
- Together, these imply monotonic improvement in the objective (no step size needed).

Related Work

- We know less about EM convergence rate than standard optimization algorithms.

Related Work

- We know less about EM convergence rate than standard optimization algorithms.
  - Convergence to stationary point in original paper [Dempster et al., 1977] had an error.

Related Work

- We know less about EM convergence rate than standard optimization algorithms.
    - Convergence to stationary point in original paper [Dempster et al., 1977] had an error.
    - Wu [1983] showed convergence to stationary point under suitable continuity assumptions.

Related Work

- We know less about EM convergence rate than standard optimization algorithms.
  - Convergence to stationary point in original paper [Dempster et al., 1977] had an error.
  - Wu [1983] showed convergence to stationary point under suitable continuity assumptions.
  - Wu [1983] and Figueiredo & Nowak [2003] discuss local vs. global optima (without rate).
    - In practice, it typically does not find a global optimum.

Related Work

- We know less about EM convergence rate than standard optimization algorithms.
  - Convergence to stationary point in original paper [Dempster et al., 1977] had an error.
  - Wu [1983] showed convergence to stationary point under suitable continuity assumptions.
  - Wu [1983] and Figueiredo & Nowak [2003] discuss local vs. global optima (without rate).
    - In practice, it typically does not find a global optimum.

  - Tseng [2004] shows local linear convergence under suitable assumptions.
    - And conjectures that global rate is likely to be sublinear.

Related Work

- We know less about EM convergence rate than standard optimization algorithms.
  - Convergence to stationary point in original paper [Dempster et al., 1977] had an error.
  - Wu [1983] showed convergence to stationary point under suitable continuity assumptions.
  - Wu [1983] and Figueiredo & Nowak [2003] discuss local vs. global optima (without rate).
    - In practice, it typically does not find a global optimum.
  - Tseng [2004] shows local linear convergence under suitable assumptions.
    - And conjectures that global rate is likely to be sublinear.
  - Salakhutdinov et al. [2002] show local superlinear local convergence.
    - Assumes ratio of hidden to observed data is small, which tends not to be satisfied.

Related Work

- We know less about EM convergence rate than standard optimization algorithms.
  - Convergence to stationary point in original paper [Dempster et al., 1977] had an error.
  - Wu [1983] showed convergence to stationary point under suitable continuity assumptions.
  - Wu [1983] and Figueiredo & Nowak [2003] discuss local vs. global optima (without rate).
    - In practice, it typically does not find a global optimum.
  - Tseng [2004] shows local linear convergence under suitable assumptions.
    - And conjectures that global rate is likely to be sublinear.
  - Salakhutdinov et al. [2002] show local superlinear local convergence.
    - Assumes ratio of hidden to observed data is small, which tends not to be satisfied.
  - Balakrishnan et al. [2017] discuss infinite data or sufficiently-large finite datasets.
    - If initial parameters are near global optima, then linear convergence to a global optimum.
    - But we know that EM usually doesn't converge to a global optimum.

Related Work

- We know less about EM convergence rate than standard optimization algorithms.
  - Convergence to stationary point in original paper [Dempster et al., 1977] had an error.
  - Wu [1983] showed convergence to stationary point under suitable continuity assumptions.
  - Wu [1983] and Figueiredo & Nowak [2003] discuss local vs. global optima (without rate).
    - In practice, it typically does not find a global optimum.
  - Tseng [2004] shows local linear convergence under suitable assumptions.
    - And conjectures that global rate is likely to be sublinear.
  - Salakhutdinov et al. [2002] show local superlinear local convergence.
    - Assumes ratio of hidden to observed data is small, which tends not to be satisfied.
  - Balakrishnan et al. [2017] discuss infinite data or sufficiently-large finite datasets.
    - If initial parameters are near global optima, then linear convergence to a global optimum.
    - But we know that EM usually doesn't converge to a global optimum.
- This work: simpler analyses, mild asumptions, true from any starting point.

Surrogate Optimization

We view EM as a surrogate optimization method [Mairal, 2013]:

Algorithm 1 Surrogate Optimization Scheme

1: **Input**: $\lambda^0 \in \Lambda$, number of iterations t.
2: **for** k = 1 to t **do**
3: Compute a surrogate function g_k of f near λ^{k-1}.
4: Update solution $\lambda^k \in \arg\min_{\lambda \in \Lambda} g_k(\lambda)$.
5: **end for**
6: Output final estimate λ^t.

Surrogate Optimization

We view EM as a surrogate optimization method [Mairal, 2013]:

Algorithm 2 Surrogate Optimization Scheme

1: **Input**: $\lambda^0 \in \Lambda$, number of iterations t.
2: **for** k = 1 to t **do**
3: Compute a surrogate function g_k of f near λ^{k-1}.
4: Update solution $\lambda^k \in \arg\min_{\lambda \in \Lambda} g_k(\lambda)$.
5: **end for**
6: Output final estimate λ^t.

Our results hold in this general framework assuming that (Assumptions 1-3):

- $f(\lambda^{k-1}) = g(\lambda^{k-1})$.
- $f(\lambda) \leq g(\lambda)$ for all λ.
- $f(\lambda) \geq f^*$ for all λ.

Convergence Analysis

- To obtain a convergence rate we need additional assumptions.

Convergence Analysis

- To obtain a convergence rate we need additional assumptions.
- Our first set of additional assumptions is that (Assumption 4a):
  - f is differentiable.
  - $\nabla f(\lambda^{k-1}) = \nabla g_k(\lambda^{k-1})$ (which is true fo EM).
  - ∇g_k is Lipschitz-continuous.

Convergence Analysis

- To obtain a convergence rate we need additional assumptions.
- Our first set of additional assumptions is that (Assumption 4a):
  - f is differentiable.
  - $\nabla f(\lambda^{k-1}) = \nabla g_k(\lambda^{k-1})$ (which is true fo EM).
  - ∇g_k is Lipschitz-continuous.

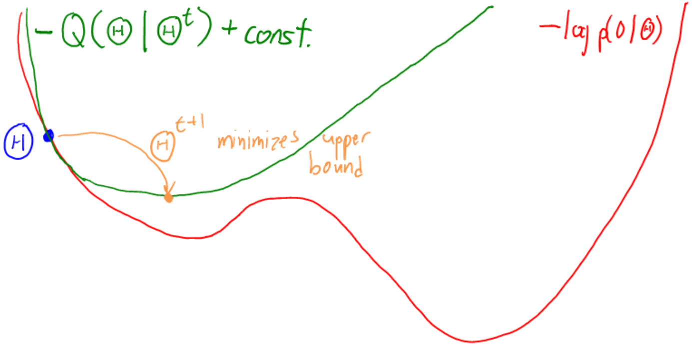
Theorem (Convergence rate of EM for differentiable functions)

Under Assumptions 1-3 and 4a, the EM algorithm starting from any λ^0 is guaranteed to find parameters λ' satisfying $\|\nabla f(\lambda')\|^2 \leq \epsilon$ once we have performed $t \geq \frac{2L[f(\lambda^0)-f^*]}{\epsilon}$ iterations.

- The same rate $O(1/\epsilon)$ rate as gradient descent (with a different constant L).

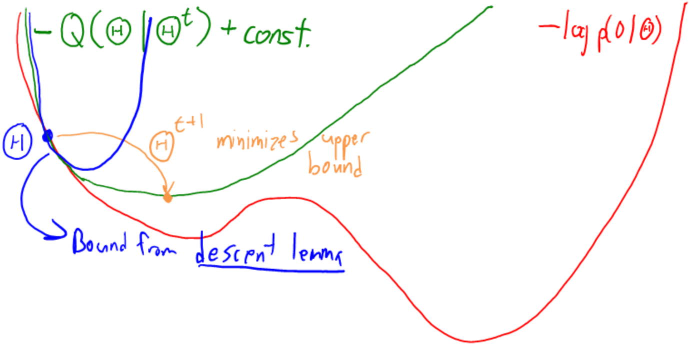
Non-Asymptotic Convergence Rate

- Proof:

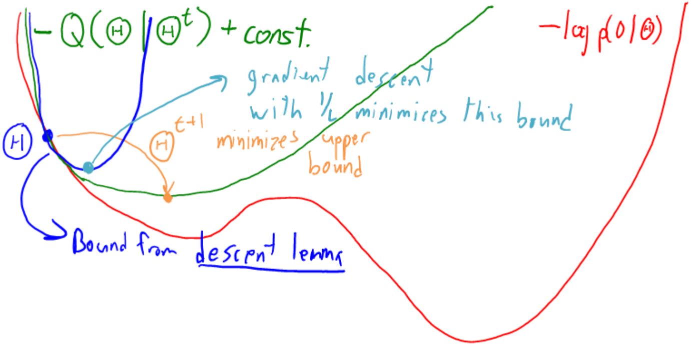


Non-Asymptotic Convergence Rate

- Proof:

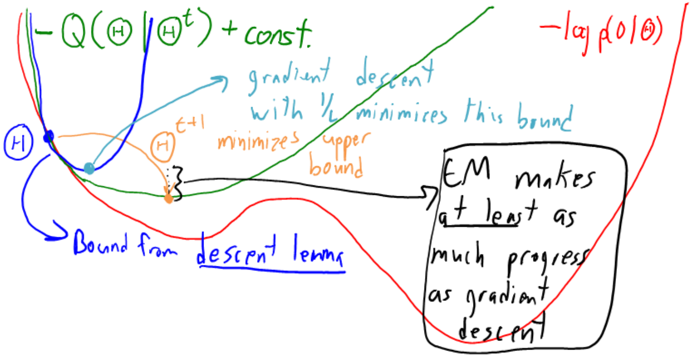


- Proof:



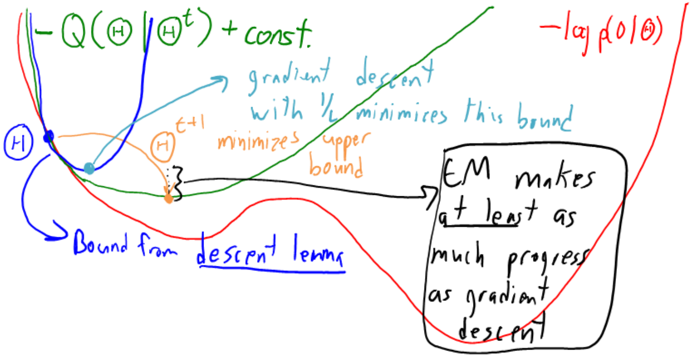
$-Q(\Theta \mid \Theta^t) + \text{const.}$
$-\log p(O \mid \Theta)$
gradient descent with 1/L minimizes this bound
Θ
Θ^{t+1} minimizes upper bound
Bound from descent lemma

- Proof:


$-Q(\Theta \mid \Theta^t) + \text{const.}$
$-\log p(O \mid \Theta)$
gradient descent with 1/L minimizes this bound
Θ^{t+1} minimizes upper bound
Θ
Bound from descent lemma
EM makes at least as much progress as gradient descent

Non-Asymptotic Convergence Rate

- Proof:



- We obtain faster rates under additional assumptions:
  - Rate in function values for convex f.
  - Linear rate for f satisfying PL.

Non-Asymptotic Convergence Rate: Non-Differentiable f

- EM is often used for non-smooth objectives like mixture of Gaussians.

Non-Asymptotic Convergence Rate: Non-Differentiable f

- EM is often used for non-smooth objectives like mixture of Gaussians.
- To llow non-smooth objectives, we consider Assumption 4b:
  - The g_k are strongly-convex (holds for mixture of Gaussians if we regularize).

Non-Asymptotic Convergence Rate: Non-Differentiable f

- EM is often used for non-smooth objectives like mixture of Gaussians.
- To llow non-smooth objectives, we consider Assumption 4b:
  - The g_k are strongly-convex (holds for mixture of Gaussians if we regularize).

- We state result in terms of what we call the EM mapping,

$$G_k(\lambda^{k-1}) = \lambda^{k-1} - \arg\min_{\lambda} g_k(\lambda),$$

which is analogous to the gradient mapping for proximal-gradient methods.

Non-Asymptotic Convergence Rate: Non-Differentiable f

- EM is often used for non-smooth objectives like mixture of Gaussians.
- To llow non-smooth objectives, we consider Assumption 4b:
  - The g_k are strongly-convex (holds for mixture of Gaussians if we regularize).

- We state result in terms of what we call the EM mapping,

$$G_k(\lambda^{k-1}) = \lambda^{k-1} - \arg\min_{\lambda} g_k(\lambda),$$

which is analogous to the gradient mapping for proximal-gradient methods.

Theorem (Convergence rate of EM for non-differentiable functions)

Under Assumptions 1-3 and 4b, the EM algorithm is guaranteed to find parameters λ' satisfying $\|G_k(\lambda')\|^2 \leq \epsilon$ once we have performed $t \geq \frac{2[f(\lambda^0)-f^*]}{\mu\epsilon}$ iterations.

Discussion of EM as an Optimization Algorithm

- We obtain the same $O(1/\epsilon)$ rate in the smooth and non-smooth case.
  - EM is appealing compared to subgradient methods because of monotonicity.

Discussion of EM as an Optimization Algorithm

- We obtain the same $O(1/\epsilon)$ rate in the smooth and non-smooth case.
  - EM is appealing compared to subgradient methods because of monotonicity.

- Given this optimization perspective on EM, many extensions are possible:
  - Generalized EM (can't exactly minimize surrogate function).
  - Second-order optimality (variant that escapes saddle points).

Discussion of EM as an Optimization Algorithm

- We obtain the same $O(1/\epsilon)$ rate in the smooth and non-smooth case.
  - EM is appealing compared to subgradient methods because of monotonicity.

- Given this optimization perspective on EM, many extensions are possible:
  - Generalized EM (can't exactly minimize surrogate function).
  - Second-order optimality (variant that escapes saddle points).
  - Accelerated EM (faster rates for locally-convex objectives).
  - Coordinate-wise, stochastic, and stochastic variance-reduced EM (large-scale).
  - Proximal and mirror descent variants.
- See the paper coming soon...

Generalized EM

- In many applications computing the $\arg\min_\lambda\{g_k(\lambda)\}$ is not possible.

Generalized EM

- In many applications computing the $\arg\min_\lambda\{g_k(\lambda)\}$ is not possible.
- Generalized EM only tries to decrease g_k.
    - This gives monotonicity but not a convergence rate.

Generalized EM

- In many applications computing the $\arg\min_\lambda\{g_k(\lambda)\}$ is not possible.
- Generalized EM only tries to decrease g_k.
  - This gives monotonicity but not a convergence rate.
- We considered two assumptions that are sufficient to maintain the $O(1/\epsilon)$ rate:
  1. *Summable Errors*: $g_k(\lambda^k) \leq \min_\lambda\{g_k(\lambda)\} + \epsilon_k$, and $\sum_{k=1}^{\infty} \epsilon_k < \infty$

Generalized EM

- In many applications computing the $\arg\min_\lambda\{g_k(\lambda)\}$ is not possible.
- Generalized EM only tries to decrease g_k.
  - This gives monotonicity but not a convergence rate.
- We considered two assumptions that are sufficient to maintain the $O(1/\epsilon)$ rate:
  1. *Summable Errors*: $g_k(\lambda^k) \leq \min_\lambda\{g_k(\lambda)\} + \epsilon_k$, and $\sum_{k=1}^{\infty} \epsilon_k < \infty$
  2. *Sufficient decrease*: $g_k(\lambda^k) \leq g_k(\lambda^{k-1}) - \alpha\|\nabla g_k(\lambda^{k-1})\|^2$ for some $\alpha > 0$.

The latter condition is easy to check.

Escaping Saddle Points

- Similar to recent work on gradients methods, we can consider finding a (ϵ, γ)-solution,

$$\|\nabla f(\lambda)\| \leq \epsilon, \quad \nabla^2 f(\lambda) \succ -\gamma I.$$

Escaping Saddle Points

- Similar to recent work on gradients methods, we can consider finding a (ϵ, γ)-solution,

$$\|\nabla f(\lambda)\| \leq \epsilon, \quad \nabla^2 f(\lambda) \succ -\gamma I.$$

- Additional assumptions:
  - f is twice differentiable and its Hessian is M-Lipschitz continuous

$$\|\nabla^2 f(\lambda) - \nabla^2 f(\lambda')\| \leq M\|x - y\|$$

Escaping Saddle Points

- Similar to recent work on gradients methods, we can consider finding a (ϵ, γ)-solution,

$$\|\nabla f(\lambda)\| \leq \epsilon, \quad \nabla^2 f(\lambda) \succ -\gamma I.$$

- Additional assumptions:
  - f is twice differentiable and its Hessian is M-Lipschitz continuous

$$\|\nabla^2 f(\lambda) - \nabla^2 f(\lambda')\| \leq M\|x - y\|$$

Theorem

SPESO return a (ϵ, γ)-solution after $\frac{3M^2[f(\lambda^0)-f^*]}{\gamma^3}t^*$ total iterations, where t^* is the number of iterations of the first order algorithm for finding a point with gradient smaller than ϵ.

Escaping Saddle Points

Algorithm 3 Saddle Point Escape for Surrogate Optimization (SPESO)

Input: $\lambda^0 \in \Lambda$, $\epsilon > 0$ and $0 < \gamma$
for $s = 1, \ldots.$ **do**
  find $\bar{\lambda}$ such that $\|\nabla f(\bar{\lambda})\| \leq \epsilon$ by executing one of the above algorithm for T^* iteration
  if $\nabla^2 f(\bar{\lambda}) \succ -\gamma I$ **then**
    $\lambda^s = \bar{\lambda}$
    return λ^s
  else
    $\lambda^s = NCJ(\lambda, \gamma)$
  end if
end for

Escaping Saddle Points

Algorithm 4 NJC: Jump along the Negative Curvature

Input: $\lambda \in \Lambda$, $0 < \gamma$
use an algorithm to compute the smallest negative eigenvalue and eigenvector of $\nabla^2 f(\lambda)$ namely μ_{min} and ν s.t. $\|\nu\| = 1$
if $\nabla f(\lambda) \neq 0$ **then**
  return $\lambda^+ = \lambda - \frac{\langle \nu, \nabla f(\lambda) \rangle}{|\langle \nu, \nabla f(\lambda) \rangle|} \frac{\gamma}{M} \nu$
else
  return $\lambda^+ = \lambda + \frac{\gamma}{M} \nu$
end if

- As in recent works, we can avoid Hessian computation by using Hessian-vector products.

Summary

- Expectation maximization (EM) is a popular algorithm for handling missing data.
  - It's a special case of surrogate optimization.

Summary

- Expectation maximization (EM) is a popular algorithm for handling missing data.
  - It's a special case of surrogate optimization.

- We give non-asymptotic convergence rates for EM under fairly weak assumptions.
  - Differentiable objective and gradient of surrogate is Lipschitz.
  - Non-differentaible objective and surrogate is strongly-convex.

Summary

- Expectation maximization (EM) is a popular algorithm for handling missing data.
  - It's a special case of surrogate optimization.
- We give non-asymptotic convergence rates for EM under fairly weak assumptions.
  - Differentiable objective and gradient of surrogate is Lipschitz.
  - Non-differentaible objective and surrogate is strongly-convex.
- We've explored a variety of extensions, notably:
  - Generalized EM where we don't exactly optimize the surrogate.
  - Variant that escapes saddle points.
  - Many more...