Examples of MM Algorithms

Kenneth Lange

Departments of Biomathematics, Human Genetics, and Statistics
University of California, Los Angeles

joint work with Eric Chi (NCSU), Joong-Ho Won (Seoul NU),
Jason Xu (Duke), and Hua Zhou (UCLA)

de Leeuw Seminar, April 26, 2018

Introduction to the MM Principle

1. The MM principle is not an algorithm, but a prescription or principle for constructing optimization algorithms.
2. The EM algorithm from statistics is a special case.
3. An MM algorithm operates by creating a surrogate function that minorizes or majorizes the objective function. When the surrogate function is optimized, the objective function is driven uphill or downhill as needed.
4. In minimization MM stands for majorize/minimize, and in maximization MM stands for minorize/maximize.

History of the MM Principle

1. Anticipators: HO Hartley (1958, EM algorithms), AG McKendrick (1926, epidemiology), CAB Smith (1957, gene counting), E Weiszfeld (1937, facilities location), F Yates (1934, multiple classification)
2. Ortega and Rheinboldt (1970) enunciate the principle in the context of line search methods.
3. de Leeuw (1977) presents an MM algorithm for multidimensional scaling contemporary with the classic Dempster et al. (1977) paper on EM algorithms.

MM Application Areas

a) robust regression, b) logistic regression,c) quantile regression, d) variance components, e) multidimensional scaling, f) correspondence analysis, g) medical imaging, h) convex programming, i) DC programming, j) geometric programming, k) survival analysis, l) nonnegative matrix factorization, m) discriminant analysis, n) cluster analysis, o) Bradley-Terry model, p) DNA sequence analysis, q) Gaussian mixture models, r) paired and multiple comparisons, s) variable selection, t) support vector machines, u) X-ray crystallography, v) facilities location, w) signomial programming, x) importance sampling, y) image restoration, and z) manifold embedding.

Rationale for the MM Principle

1. It can generate an algorithm that avoids matrix inversion.
2. It can separate the parameters of a problem.
3. It can linearize an optimization problem.
4. It can deal gracefully with equality and inequality constraints.
5. It can restore symmetry.
6. It can turn a non-smooth problem into a smooth problem.

Majorization and Definition of the Algorithm

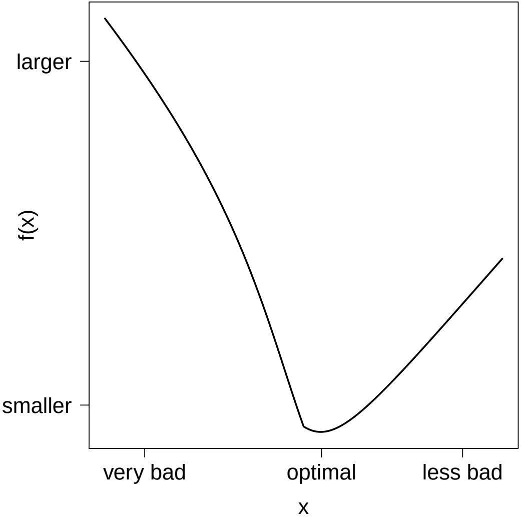
1. A function $g(\boldsymbol{\theta} \mid \boldsymbol{\theta}_n)$ is said to majorize the function $f(\boldsymbol{\theta})$ at $\boldsymbol{\theta}_n$ provided

$$\begin{array}{rcll} f(\boldsymbol{\theta}_n) & = & g(\boldsymbol{\theta}_n \mid \boldsymbol{\theta}_n) & \quad \text{tangency at } \boldsymbol{\theta}_n \\ f(\boldsymbol{\theta}) & \leq & g(\boldsymbol{\theta} \mid \boldsymbol{\theta}_n) & \quad \text{domination for all } \boldsymbol{\theta}. \end{array}$$

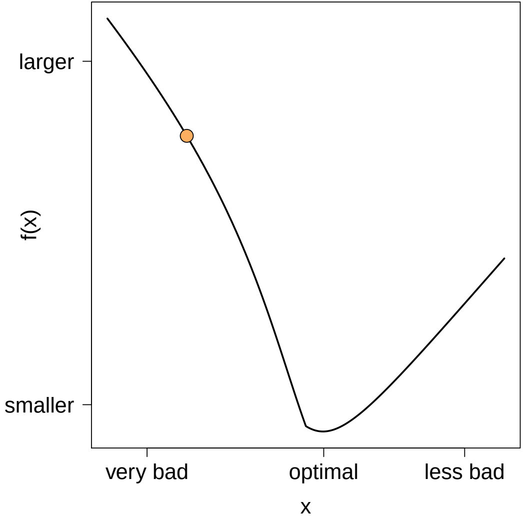
   The majorization relation between functions is closed under the formation of sums, nonnegative products, limits, and composition with an increasing function.

2. A function $g(\boldsymbol{\theta} \mid \boldsymbol{\theta}_n)$ is said to minorize the function $f(\boldsymbol{\theta})$ at $\boldsymbol{\theta}_n$ provided $-g(\boldsymbol{\theta} \mid \boldsymbol{\theta}_n)$ majorizes $-f(\boldsymbol{\theta})$.
3. In minimization, we choose a majorizing function $g(\boldsymbol{\theta} \mid \boldsymbol{\theta}_n)$ and minimize it. This produces the next point $\boldsymbol{\theta}_{n+1}$ in the algorithm.

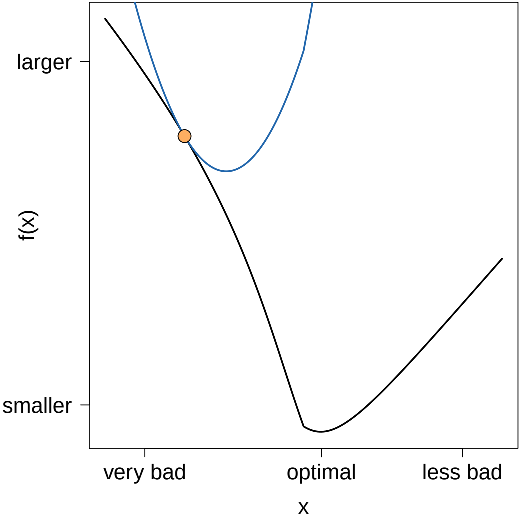
MM Algorithm in Action



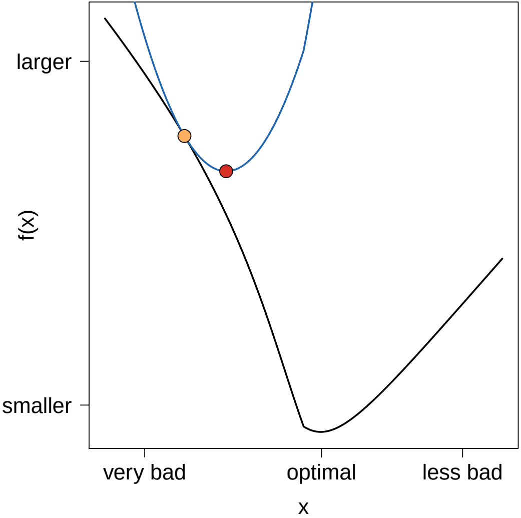
MM Algorithm in Action



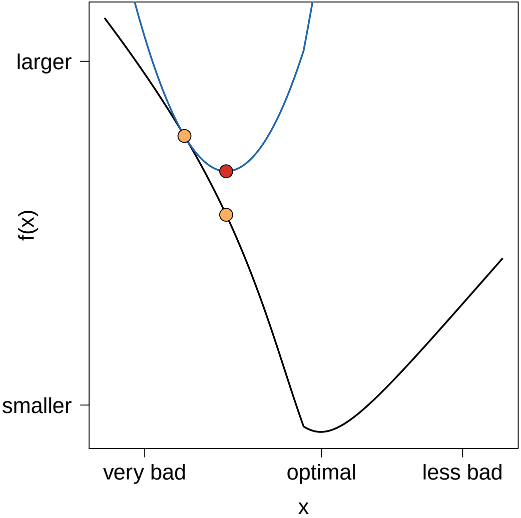
MM Algorithm in Action



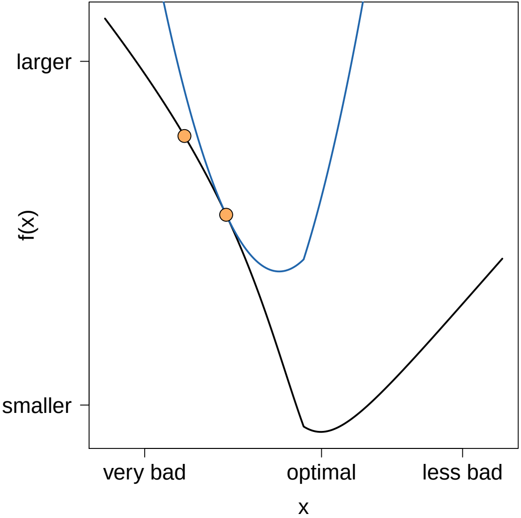
MM Algorithm in Action



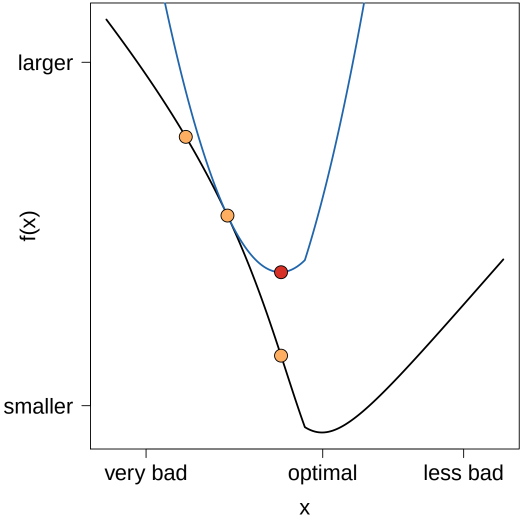
MM Algorithm in Action



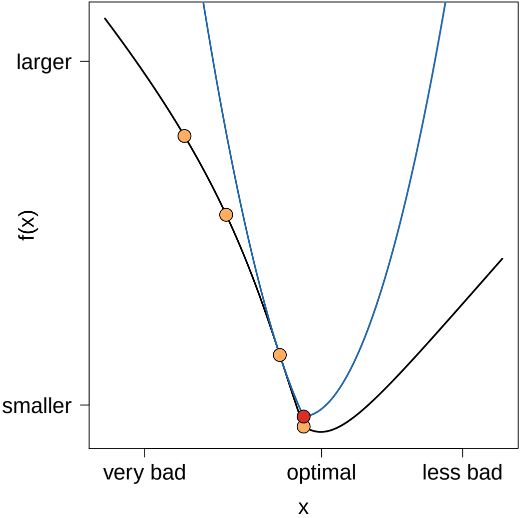
MM Algorithm in Action



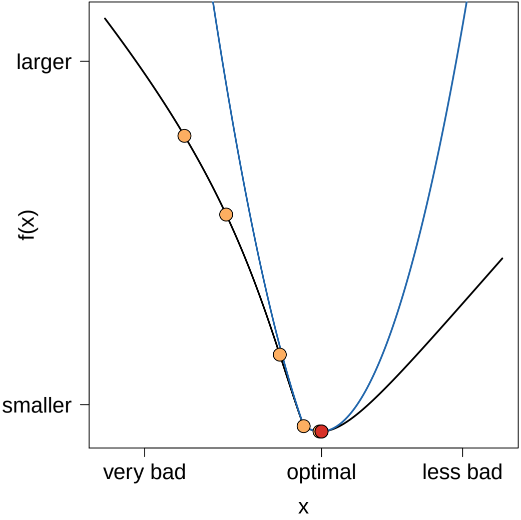
MM Algorithm in Action



MM Algorithm in Action



MM Algorithm in Action



Descent Property

1. An MM minimization algorithm satisfies the descent property $f(\boldsymbol{\theta}_{n+1}) \leq f(\boldsymbol{\theta}_n)$ with strict inequality unless both

$$\begin{aligned} g(\boldsymbol{\theta}_{n+1} \mid \boldsymbol{\theta}_n) &= g(\boldsymbol{\theta}_n \mid \boldsymbol{\theta}_n) \\ f(\boldsymbol{\theta}_{n+1}) &= g(\boldsymbol{\theta}_{n+1} \mid \boldsymbol{\theta}_n). \end{aligned}$$

2. The descent property follows from the definitions and

$$f(\boldsymbol{\theta}_{n+1}) \;\leq\; g(\boldsymbol{\theta}_{n+1} \mid \boldsymbol{\theta}_n) \;\leq\; g(\boldsymbol{\theta}_n \mid \boldsymbol{\theta}_n) \;=\; f(\boldsymbol{\theta}_n).$$

3. The descent property makes the MM algorithm very stable.

Example 1: Minimum of $\cos(x)$

The univariate function $f(x) = \cos(x)$ achieves its minimum of -1 at odd multiples of π and its maximum of 1 at even multiples of π. For a given x_n, the second-order Taylor expansion

$$\cos(x) \quad = \quad \cos(x_n) - \sin(x_n)(x - x_n) - \frac{1}{2}\cos(z)(x - x_n)^2$$

holds for some z between x and x_n. Because $|\cos(z)| \le 1$, the surrogate function

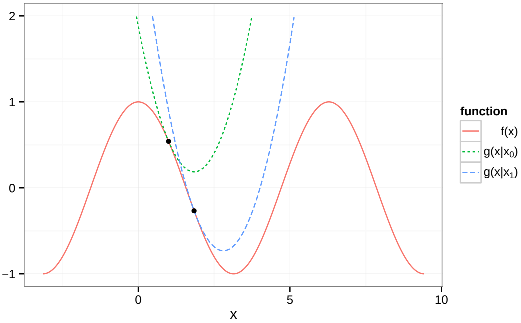
$$g(x \mid x_n) \quad = \quad \cos(x_n) - \sin(x_n)(x - x_n) + \frac{1}{2}(x - x_n)^2$$

majorizes $f(x)$. Solving $\frac{d}{dx}g(x \mid x_n) = 0$ gives the MM algorithm

$$x_{n+1} \quad = \quad x_n + \sin(x_n)$$

for minimizing $f(x)$ and represents an instance of the quadratic upper bound principle.

Majorization of $\cos x$



MM and Newton Iterates for Minimizing $\cos(x)$

	MM		**Newton**	
n	x_n	$\cos(x_n)$	y_n	$\cos(y_n)$
0	2.00000000	-0.41614684	2.00000000	-0.41614684
1	2.90929743	-0.97314057	4.18503986	-0.50324437
2	3.13950913	-0.99999783	2.46789367	-0.78151929
3	3.14159265	-1.00000000	3.26618628	-0.99224825
4	3.14159265	-1.00000000	3.14094391	-0.99999979
5	3.14159265	-1.00000000	3.14159265	-1.00000000

Example 2: Robust Regression

According to Geman and McClure, robust regression can be achieved by minimizing the amended linear regression criterion

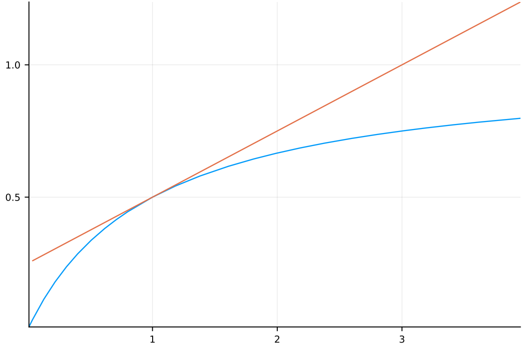
$$f(\boldsymbol{\beta}) \quad = \quad \sum_{i=1}^{m} \frac{(y_i - \boldsymbol{x}_i^* \boldsymbol{\beta})^2}{c + (y_i - \boldsymbol{x}_i^* \boldsymbol{\beta})^2}.$$

Here y_i and $\boldsymbol{x}_i$ are the response and the predictor vector for case i and $c > 0$. Majorization is achieved via the concave function $h(s) = \frac{s}{c+s}$. In view of the linear majorization $h(s) \le h(s_n) + h'(s_n)(s - s_n)$, substitution of $(y_i - \boldsymbol{x}_i^* \boldsymbol{\beta})^2$ for s gives the surrogate function

$$g(\boldsymbol{\beta} \mid \boldsymbol{\beta}_n) \quad = \quad \sum_{i=1}^{m} w_{ni}(y_i - \boldsymbol{x}_i^* \boldsymbol{\beta})^2 + \text{constant},$$

where the weight w_{ni} equals $h'(s)$ evaluated at $s_n = (y_i - \boldsymbol{x}_i^* \boldsymbol{\beta}_n)^2$. The update $\boldsymbol{\beta}_{n+1}$ is found by minimizing this weighted least squares criterion.

Majorization of $h(s) = \frac{s}{1+s}$ at $s_n = 1$



Example 3: Missing Data in K-Means Clustering

Lloyd's algorithm is one of the earliest and simplest algorithms for K-means clustering. A recent paper extends K-means clustering to missing data. For subject i we observe an indexed set of components y_{ij} of a vector $\boldsymbol{y}_i \in \mathrm{R}^d$. Call the index set O_i. Subjects must be assigned to one of K clusters. Let C_k denote the set of subjects currently assigned to cluster k. With this notation we seek to minimize the objective function

$$\sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in O_i} (y_{ij} - \mu_{kj})^2,$$

where $\boldsymbol{\mu}_k$ is the center of cluster k.
Reference: Chi JT, Chi EC, Baraniuk RG (2016) k-POD: A method for k-means clustering of missing data. *The American Statistician* 70:91–99

Reformulation of Lloyd's Algorithm

Lloyd's algorithm alternates cluster reassignment with re-estimation of cluster centers. If we fix the centers, then subject i should be reassigned to the cluster k minimizing the quantity

$$\sum_{j \in O_i} (y_{ij} - \mu_{kj})^2.$$

Re-estimation of the cluster centers relies on the MM principle. The surrogate function

$$\sum_{k=1}^{K} \sum_{i \in C_k} \Big[\sum_{j \in O_i} (y_{ij} - \mu_{kj})^2 + \sum_{j \notin O_i} (\mu_{nkj} - \mu_{kj})^2 \Big].$$

majorizes the objective around the cluster centers $\boldsymbol{\mu}_{nk}$ at the current iteration n. Note that the extra terms are nonnegative and vanish when $\boldsymbol{\mu}_k = \boldsymbol{\mu}_{nk}$.

Center Updates under Lloyd's Algorithm

If we define

$$\tilde{y}_{nij} \quad = \quad \begin{cases} y_{ij} & j \in O_i \\ \mu_{nkj} & j \notin O_i, \end{cases}$$

then the surrogate can be rewritten as $\sum_{k=1}^{K} \sum_{j \in C_i} \|\tilde{\boldsymbol{y}}_{nj} - \boldsymbol{\mu}_k\|^2$. Its minimum is achieved at the revised centers

$$\boldsymbol{\mu}_{n+1,i} \quad = \quad \frac{1}{|C_i|} \sum_{j \in C_i} \tilde{\boldsymbol{y}}_{nj}.$$

In other words, the center equals the within cluster average over the combination of the observed data and the imputed data. The MM principle restores symmetry and leads to exact updates.

Robust Version of Lloyd's Algorithm

It is worth mentioning that the same considerations apply to other objective functions. For instance, if we substitute ℓ_1 norms for sums of squares, then the missing component majorization works with the term $|\mu_{nkj} - \mu_{kj}|$ replacing the term $(\mu_{nkj} - \mu_{kj})^2$. In this case, each component of the update $\boldsymbol{\mu}_{n+1,kj}$ equals the corresponding median of the completed data points $\tilde{\boldsymbol{y}}_{ni}$ assigned to cluster k. This version of clustering is less subject to the influence of outliers.

Strengths and Weaknesses of K-Means

1. Strength: Speed and simplicity of implementation
2. Strength: Ease of interpretation
3. Weakness: Based on spherical clusters
4. Weakness: Lloyd's algorithm attracted to local minima
5. Weakness: Distortion by outliers
6. Weakness: Choice of number classes K

K-Harmonic Means

The K-harmonic means clustering algorithm (KHM) is a clustering method that is less sensitive to initialization than K-means (B Zhang et al (1999) Hewlett-Packard Technical Report). It minimizes the criterion

$$f_{-1}(\boldsymbol{\mu}) \quad = \quad \sum_{i=1}^{n} \frac{1}{\sum_{k=1}^{K} \frac{1}{\|\boldsymbol{y}_i - \boldsymbol{\mu}_k\|^2}}.$$

The corresponding K-means criterion without missing data is

$$f_{-\infty}(\boldsymbol{\mu}) \quad = \quad \sum_{i=1}^{n} \min_{1 \le k \le K} \|\boldsymbol{y}_i - \boldsymbol{\mu}_k\|^2.$$

Zhang et al devised an ad hoc algorithm for minimizing $f_{-1}(\boldsymbol{\mu})$ without realizing that it is an MM algorithm. Can we justify their algorithm and extend it to a broader context?

Power Means

The power mean of order s of K nonnegative numbers $x_1, \ldots, x_K$ is

$$M_s(\boldsymbol{x}) \quad = \quad \Big(\frac{1}{K}\sum_{k=1}^{K} x_k^s\Big)^{\frac{1}{s}}.$$

The choices $s = 1$ and $s = -1$ correspond to the arithmetic and harmonic means. The special case $s = 0$ is defined by continuity to be the geometric mean $\sqrt[K]{x_1 \cdots x_K}$. One can check that $M_s(\boldsymbol{x})$ is continuous, positively homogeneous, and symmetric in its arguments. Again by continuity, $M_s(\mathbf{0}) = 0$. The gradient

$$\frac{\partial}{\partial x_j} M_s(\boldsymbol{x}) \quad = \quad \Big(\frac{1}{K}\sum_{k=1}^{K} x_k^s\Big)^{\frac{1}{s}-1} \frac{1}{K} x_j^{s-1}$$

shows that $M_s(\boldsymbol{x})$ is strictly increasing in each variable. The inequality $M_s(\boldsymbol{x}) \le M_t(\boldsymbol{x})$ for $s \le t$ and limits $\lim_{s\to-\infty} M_s(\boldsymbol{x}) = \min\{x_1, \ldots, x_K\}$ and $\lim_{s\to\infty} M_s(\boldsymbol{x}) = \max\{x_1, \ldots, x_K\}$ are exercises in classical analysis.

Relevance of Power Means to K-Means

Our comments on power means suggest the clustering criterion

$$\begin{aligned} f_s(\boldsymbol{\mu}) &= \sum_{i=1}^{n} M_s(\|\boldsymbol{y}_i - \boldsymbol{\mu}_1\|^2, \ldots, \|\boldsymbol{y}_i - \boldsymbol{\mu}_K\|^2) \\ &= \sum_{i=1}^{n} \Big(\frac{1}{K}\sum_{k=1}^{K} \|\boldsymbol{y}_i - \boldsymbol{\mu}_k\|^{2s}\Big)^{\frac{1}{s}} \end{aligned}$$

consistent with our previous notation $f_{-\infty}(\boldsymbol{\mu})$ (K-means) and f_{-1} (harmonic mean). The cluster centers $\boldsymbol{\mu}_k$ (columns of $\boldsymbol{\mu}$) can be estimated by minimizing $f_s(\boldsymbol{\mu})$. We can track the solution matrices to the minimum of $f_{-\infty}(\boldsymbol{\mu})$. The advantage of this strategy is that the surface $f_s(\boldsymbol{\mu})$ is less bumpy that the surface $f_{-\infty}(\boldsymbol{\mu})$. For example, in the linear case $s = 1$, all centers coincide at the single global minimum. The following slides illustrate how most local minima flatten into nonexistence as $s \to 1$.

Objective function surface: K-means

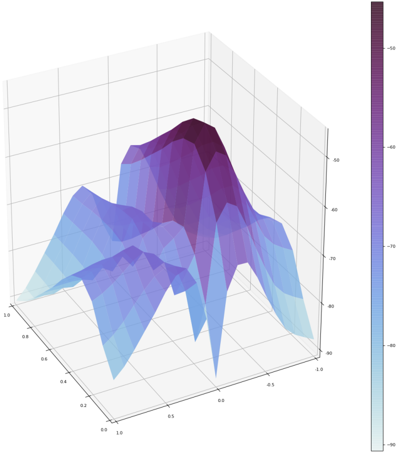
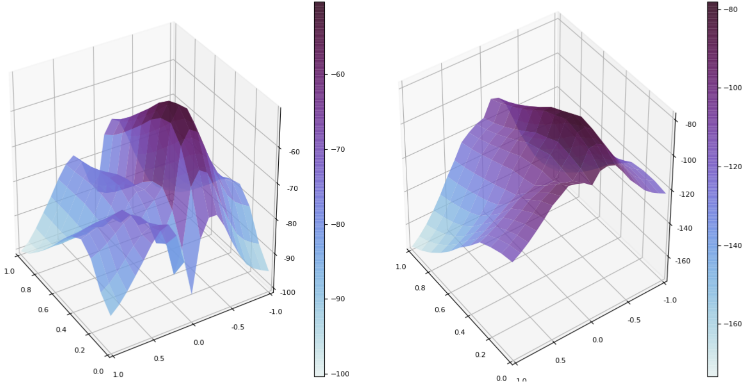


Figure: A cross-section of the K-means objective for $n = 100$ simulated data points from $K = 3$ clusters in dimension $d = 1$. Two cluster centers vary along the axes, holding the third center fixed at its true value.

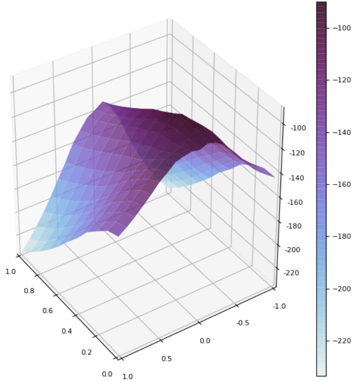
Objective function surface: power means



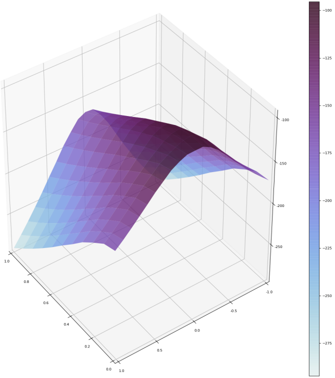
(a) $s = -10.0$

(b) $s = -1.0$ (KHM)

Objective function surface: power means



(c) $s = -0.2$



(d) $s = 0.3$

An MM Power Means Clustering Algorithm

Derivation of the MM algorithm depends on the concavity of the power mean function $M_s(\boldsymbol{x})$ for $s \leq 1$. For $s > 1$, $M_s(\boldsymbol{x})$ is convex. (Proofs omitted.) Concavity entails the inequality,

$$M_s(\boldsymbol{x}) \leq M_s(\boldsymbol{x}_n) + dM_s(\boldsymbol{x}_n)(\boldsymbol{x} - \boldsymbol{x}_n)$$

for all $\boldsymbol{x} \geq \boldsymbol{0}$. Substituting $\|\boldsymbol{y}_i - \boldsymbol{\mu}_k\|^2$ for x_k yields the majorization

$$f_s(\boldsymbol{\mu}) \quad \leq f_s(\boldsymbol{\mu}_n) + \textstyle\sum_{i=1}^n \sum_{k=1}^K w_{nik}(\|\boldsymbol{y}_i - \boldsymbol{\mu}_k\|^2 - \|\boldsymbol{y}_i - \boldsymbol{\mu}_{nk}\|^2),$$

where the weights are positive numbers derived from the partial derivatives of $M_s(\boldsymbol{x})$. The MM algorithm gives the minimum of the surrogate as

$$\boldsymbol{\mu}_{n+1,k} \quad = \quad \frac{1}{\sum_{i=1}^n w_{nik}} \sum_{i=1}^n w_{nik} \boldsymbol{x}_i .$$

Thus, all updates $\boldsymbol{\mu}_{n+1,k}$ stay within the convex hull of the data points.

Simulation study

- Sample $n = 2500$ points according to standard multivariate normal distribution from $K = 50$ randomly sized clusters
- When $d = 2$, this is exactly the same setting as the original K-harmonic means paper, but we will vary d.
- The center matrix $\boldsymbol{\mu}_{true}$ has uniform random entries scaled up by a scale factor of r randomly chosen between 15 and 30
- Performance measure: $\sqrt{\dfrac{KM(\boldsymbol{x}, \hat{\boldsymbol{\mu}})}{KM(\boldsymbol{x}, \boldsymbol{\mu}_{opt})}}$

  where KM denotes the K-means objective function, $\hat{\boldsymbol{\mu}}$ is the estimate of the centers, and $\boldsymbol{\mu}_{opt}$ is the estimate obtained by running Lloyd's algorithm initialized at $\boldsymbol{\mu}_{true}$.

Performance comparison

	$d = 2$	$d = 5$	$d = 10$	$d = 30$	$d = 100$	$d = 200$
Lloyd's	1.151	1.415	1.538	1.617	1.603	1.794
KHM	1.012	1.934	2.636	2.599	2.485	2.665
$s_0 = -1.0$	**1.012**	**1.066**	1.111	1.509	2.308	2.190
$s_0 = -3.0$	1.032	1.082	**1.081**	1.143	1.662	1.485
$s_0 = -10.0$	1.035	1.197	1.212	**1.138**	**1.104**	**1.131**
$s_0 = -20.0$	1.066	1.268	1.272	1.231	1.140	1.178

- Here s_0 is the initial power mean index; recall that $s \to -\infty$.
- Initialized each algorithm from matching randomized centers, averaged over 25 trials
- Same message under K-means++ and other initializations and different performance measures (variation of information, adjusted random index)
- Power means perform best. Harmonic means outperforms standard K-means only in low dimensions.

Background on Distance Majorization

1. The Euclidean distance $\operatorname{dist}(\boldsymbol{x}, C) = \min_{\boldsymbol{y} \in C} \|\boldsymbol{x} - \boldsymbol{y}\|$ can be equivalently expressed using projection onto C:

$$\operatorname{dist}(\boldsymbol{x}, C) \quad = \quad \|\boldsymbol{x} - P_C(\boldsymbol{x})\|$$

2. The closest point $P_C(\boldsymbol{x})$ in C to $\boldsymbol{x}$ exists and is unique when C is closed and convex. For a nonconvex set, $P_C(\boldsymbol{x})$ may multi-valued. Many projection operators $P_C(\boldsymbol{x})$ have explicit formulas or reduce to simple algorithms.
3. The standard distance majorization is

$$\operatorname{dist}(\boldsymbol{x}, C) \quad \leq \quad g(\boldsymbol{x} \mid \boldsymbol{x}_n) \quad = \quad \|\boldsymbol{x} - P_C(\boldsymbol{x}_n)\|.$$

4. The function $\operatorname{dist}(\boldsymbol{x}, C)$ is typically non-differentiable at boundary points even for convex C; however, $\operatorname{dist}(\boldsymbol{x}, C)^2$ is differentiable whenever $P_C(\boldsymbol{x})$ is single valued. In this case, one can calculate $\nabla \operatorname{dist}(\boldsymbol{x}, C)^2 = 2[\boldsymbol{x} - P_C(\boldsymbol{x})]$.

Sample Projection Operators

1. If $C = \{\boldsymbol{x} \in \mathrm{R}^p : \|\boldsymbol{x} - \boldsymbol{z}\| \leq r\}$ is a closed ball, then

$$P_C(\boldsymbol{y}) \quad = \quad \begin{cases} \boldsymbol{z} + \frac{r}{\|\boldsymbol{y}-\boldsymbol{z}\|}(\boldsymbol{y} - \boldsymbol{z}) & \boldsymbol{y} \notin C \\ \boldsymbol{y} & \boldsymbol{y} \in C\,. \end{cases}$$

2. If $C = [\boldsymbol{a}, \boldsymbol{b}]$ is a closed rectangle in R^p, then $P_C(\boldsymbol{y})$ has entries

$$P_C(\boldsymbol{y})_i \quad = \quad \begin{cases} a_i & y_i < a_i \\ y_i & y_i \in [a_i, b_i] \\ b_i & y_i > b_i\,. \end{cases}$$

3. If $C = \{\boldsymbol{x} \in \mathrm{R}^p : \boldsymbol{a}^*\boldsymbol{x} = b\}$ for $\boldsymbol{a} \neq \boldsymbol{0}$ is a hyperplane, then

$$P_C(\boldsymbol{y}) \quad = \quad \boldsymbol{y} - \frac{\boldsymbol{a}^*\boldsymbol{y} - b}{\|\boldsymbol{a}\|^2}\boldsymbol{a}.$$

4. If C is the unit sphere (surface of the unit ball), then $P_C(\boldsymbol{x}) = \boldsymbol{x}/\|\boldsymbol{x}\|$ for all $\boldsymbol{x} \neq \boldsymbol{0}$. However, $P_C(\boldsymbol{0}) = C$.

Example 4a: Averaged Projections

Let $S_1, \ldots, S_m$ be closed sets. The method of averaged projections attempts to find a point in their intersection $S = \cap_{j=1}^m S_j$. To derive the algorithm, consider the proximity function

$$f(\boldsymbol{x}) \quad = \quad \sum_{j=1}^m \mathrm{dist}(\boldsymbol{x}, S_j)^2.$$

It's minimum value of 0 is attained by any $\boldsymbol{x} \in \cap_{j=1}^m S_j$. The surrogate

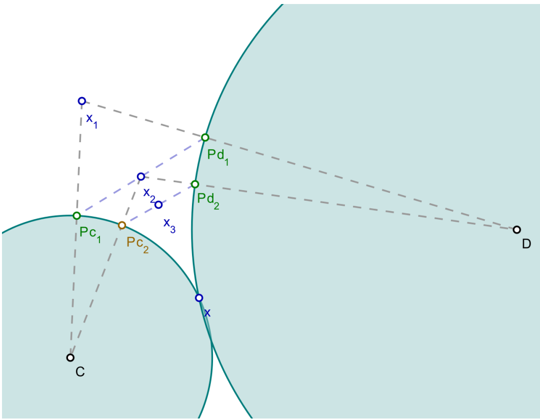
$$g(\boldsymbol{x} \mid \boldsymbol{x}_n) \quad = \quad \sum_{j=1}^m \|\boldsymbol{x} - P_{S_j}(\boldsymbol{x}_n)\|^2$$

majorizes $f(\boldsymbol{x})$. The minimum point of $g(\boldsymbol{x} \mid \boldsymbol{x}_n)$,

$$\boldsymbol{x}_{n+1} \quad = \quad \frac{1}{m} \sum_{j=1}^m P_{S_j}(\boldsymbol{x}_n),$$

defines the averaged projection. The MM principle guarantees that $\boldsymbol{x}_{n+1}$ decreases the proximity function.

Depiction of Averaged Projections



Example 4b: Alternating Projections

For two sets closed S_1 and S_2, consider the problem of minimizing the proximity function

$$f(\boldsymbol{x}) \quad = \quad \mathrm{dist}(\boldsymbol{x}, S_2)^2$$

subject to the constraint $\boldsymbol{x} \in S_1$. Clearly, $S_1 \cap S_2 \neq \emptyset$ is equivalent to a minimum value of 0. The function

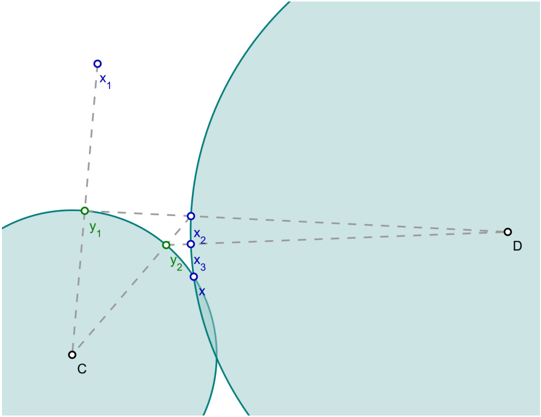
$$g(\boldsymbol{x} \mid \boldsymbol{x}_n) \quad = \quad \|\boldsymbol{x} - P_{S_2}(\boldsymbol{x}_n)\|^2$$

majorizes $f(\boldsymbol{x})$ on S_1 and is minimized by taking

$$\boldsymbol{x}_{n+1} \quad = \quad P_{S_1} \circ P_{S_2}(\boldsymbol{x}_n).$$

This is Von Neumann's method of alternating projections for finding $\boldsymbol{x} \in S_1 \cap S_2$.

Depiction of Alternating Projections



Example 5: Intensity-Modulated Radiation Therapy

This problem involves optimizing beamlet intensities in radiation oncology. Mathematically, both domain and range constraints are important. The tumor and surrounding tissues are divided into voxels.

The goals/constraints:

1. Sufficiently irradiate cancerous (target) tissue
2. Minimize radiation to normal tissue
3. Impose nonnegativity constraints on the entries of $\boldsymbol{x}$.

The dose $\boldsymbol{d} = \boldsymbol{A}\boldsymbol{x}$ is a linear map of beamlet intensities $\boldsymbol{x}$.

Lower bound L_j on target regions j: for all voxels i in region j

$$d_i \geq L_j$$

Upper bound U_j on non-target regions j: for all voxels i in region j cap the radiation

$$d_i \leq U_j.$$

MM for Multiset Nonlinear Split Feasibility

For a smooth function $h(\boldsymbol{x})$, consider the problem of finding $\boldsymbol{x} \in \cap_i C_i$ such the $h(\boldsymbol{x}) \in \cap_j Q_j$. This problem can be attacked by minimizing

$$f(\boldsymbol{x}) \quad = \quad \frac{1}{2}\sum_i \operatorname{dist}(\boldsymbol{x}, C_i)^2 + \frac{1}{2}\sum_j \operatorname{dist}[h(\boldsymbol{x}), Q_j]^2.$$

A split feasible point exits if and only if the minimum value is 0. The MM principle suggests minimizing the surrogate

$$g(\boldsymbol{x} \mid \boldsymbol{x}_n) = \frac{1}{2}\sum_i \|\boldsymbol{x} - P_{C_i}(\boldsymbol{x}_n)\|^2 + \frac{1}{2}\sum_j \|h(\boldsymbol{x}) - P_{Q_j}[h(\boldsymbol{x}_n)]\|^2$$

to find an improved point $\boldsymbol{x}_{n+1}$. When $h(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x}$, the MM update involves solving a system of linear equations and reduces to the iterative projection algorithm of Censor & Elfving (1994). In the nonlinear case, one can exploit the inexact minimization

$$\boldsymbol{x}_{n+1} \quad = \quad \boldsymbol{x}_n - d^2 g(\boldsymbol{x}_n \mid \boldsymbol{x}_n)^{-1} \nabla g(\boldsymbol{x}_n \mid \boldsymbol{x}_n)$$

provided by applying one step of Newton's method to the surrogate.

MM for Multiset Nonlinear Split Feasibility

The gradient and Hessian of the surrogate are

$$\begin{aligned}
\nabla g(\boldsymbol{x}_n \mid \boldsymbol{x}_n) &= \sum_i [\boldsymbol{x}_n - P_{C_i}(\boldsymbol{x}_n)] + \sum_j \nabla h(\boldsymbol{x}_n)\{h(\boldsymbol{x}_n) - \mathcal{P}_{Q_j}[h(\boldsymbol{x}_n)\} \\
d^2 g(\boldsymbol{x}_n \mid \boldsymbol{x}_n) &= \sum_i \boldsymbol{I} + \sum_j \nabla h(\boldsymbol{x}_n) dh(\boldsymbol{x}_n) \\
&\quad + \sum_j d^2 h(\boldsymbol{x})\{h(\boldsymbol{x}_n) - P_{Q_j}[h(\boldsymbol{x}_n)]\} \\
&\approx (\#\text{ of } i\text{'s })\boldsymbol{I} + (\#\text{ of } j\text{'s })\nabla h(\boldsymbol{x}_n) dh(\boldsymbol{x}_n).
\end{aligned}$$

When all constraints Q_j are satisfied, $P_{Q_j}[h(\boldsymbol{x}_n)] = h(\boldsymbol{x}_n)$, and the approximation is exact. Dropping the second sum in the Hessian to avoid the tensor $d^2 h(\boldsymbol{x}_n)$ is analogous to the Gauss-Newton maneuver in nonlinear regression. The approximation to the Hessian is positive definite and well conditioned. Step halving is seldom necessary.

Graphical Display of IMRT Solution

1,000-5,000 beamlets and nearly 100,000 voxels, but only 5-10 regions

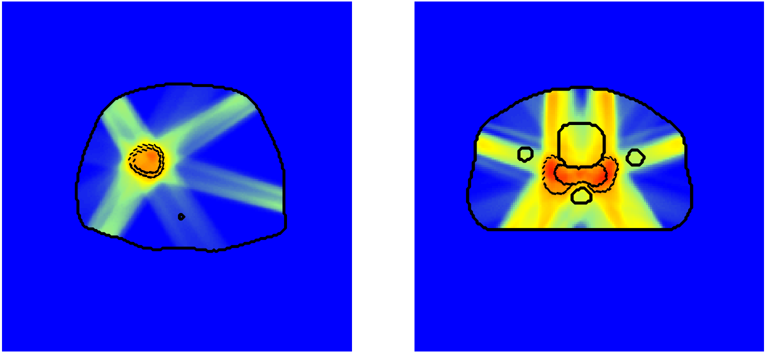


Figure: Solutions to the voxel-by-voxel split feasibility problem on a cross-section of liver data (left) and prostate data (right).

Proximal Distance Algorithm

1. Problem: Minimize a continuous function $f(\boldsymbol{x})$ subject to $\boldsymbol{x} \in C$.
2. Let $\boldsymbol{x}_\rho$ minimize the unconstrained function $f(\boldsymbol{x}) + \frac{\rho}{2}\operatorname{dist}(\boldsymbol{x}, C)^2$ for $\rho > 0$. Then any cluster point of $\boldsymbol{x}_\rho$ as $\rho \to \infty$ is feasible and attains the constrained minimum value of $f(\boldsymbol{x})$. If $f(\boldsymbol{x})$ is coercive and possesses a unique minimum point $\boldsymbol{x}_\infty$, then $\boldsymbol{x}_\rho \to \boldsymbol{x}_\infty$.
3. The proximal distance method minimizes $f(\boldsymbol{x}) + \frac{\rho}{2}\operatorname{dist}(\boldsymbol{x}, C)^2$ by distance majorization. If $f(\boldsymbol{x})$ is convex, then this MM procedure is a concave-convex algorithm.
4. For many choices of $f(\boldsymbol{x})$, the proximal operator

   $$\boldsymbol{x}_{n+1} \quad = \quad \operatorname{prox}_{\rho^{-1}f}(\boldsymbol{x}_n) \quad = \quad \operatorname{argmin}_{\boldsymbol{x}}\left[f(\boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{x} - P_C(\boldsymbol{x}_n)\|^2\right]$$

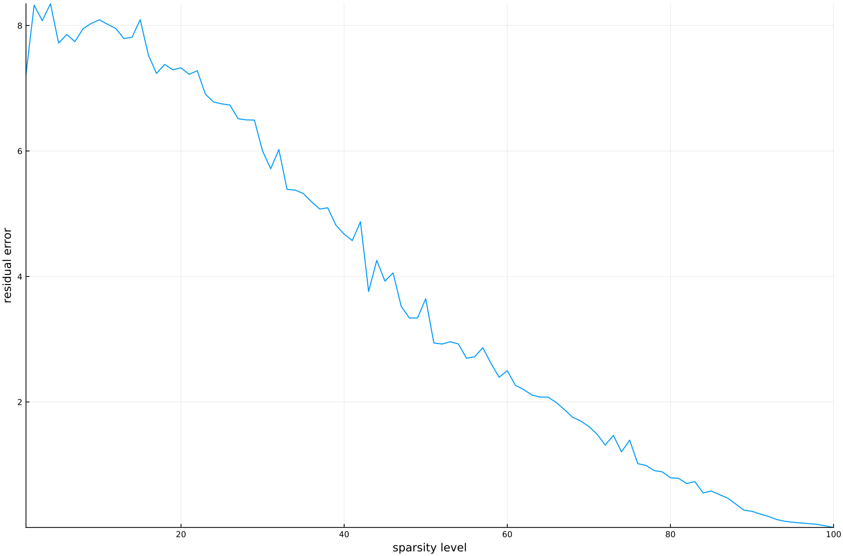
   is explicitly known.
5. In practice, ρ is gradually increased to some large value, say 10^5.

Example 6: Sparse Dominant Eigenvector

1. For a symmetric matrix $\boldsymbol{A}$, the dominant eigenvector maximizes $\boldsymbol{x}^t\boldsymbol{A}\boldsymbol{x}$ subject to $\|\boldsymbol{x}\| = 1$.
2. One can introduce sparsity by requiring that at most k components of $\boldsymbol{x}$ be nonzero. The constraint set S_k is the unit sphere with this additional sparsity constraint.
3. The projection operator $P_{S_k}(\boldsymbol{y})$ sets to 0 all but the k largest components of $\boldsymbol{y}$ in absolute value. It then replaces the result $\tilde{\boldsymbol{y}}$ by $\tilde{\boldsymbol{y}}/\|\tilde{\boldsymbol{y}}\|$.
4. A sparse dominant eigenvector is then found by minimizing $f(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}^t\boldsymbol{A}\boldsymbol{x}$ subject to $\boldsymbol{x} \in S_k$.
5. The proximal distance update solves $\mathbf{0} = -\boldsymbol{A}\boldsymbol{x} + \rho[\boldsymbol{x} - P_{S_k}(\boldsymbol{x}_n)]$ in the form

$$\boldsymbol{x}_{n+1} \quad = \quad (\rho\boldsymbol{I} - \boldsymbol{A})^{-1}\rho P_{S_k}(\boldsymbol{x}_n) \quad = \quad \sum_{n=0}^{\infty}(\rho^{-1}\boldsymbol{A})^n P_{S_k}(\boldsymbol{x}_n).$$

Plot of $\|\boldsymbol{Ax} - \lambda \boldsymbol{x}\|$ for $\boldsymbol{A}$ a 100×100 Symmetric Matrix



Remaining Challenges

1. Devise new MM algorithms, particularly for high dimensional and nonconvex problems.
2. Quantify the local rate of convergence of the MM algorithm in the presence of complex constraints. When does an MM algorithm converge at a sublinear rate?
3. Estimate the computational complexity of various MM algorithms.
4. Devise new annealing schemes to avoid local minima.
5. Devise better ways of accelerating MM and EM algorithms.
6. Write Julia and R packages for various MM algorithms. Parallel and GPU versions especially needed.

References

1. de Leeuw J (1977) Applications of convex analysis to multidimensional scaling. *Recent Developments in Statistics* (editors Barra JR, Brodeau F, Romie G, Van Cutsem B), North Holland, Amsterdam, pp 133–146
2. de Leeuw J, Heiser WJ (1977), Convergence of correction matrix algorithms for multidimensional scaling. *Geometric Representations of Relational Data* (editors Lingoes JC, Roskam E , Borg I), pp. 735–752, Mathesis Press
3. de Leeuw J (2016) *Block Relaxation Methods in Statistics*. Internet Book
4. Hunter DR, Lange K (2004) A tutorial on MM algorithms. *American Statistician* 58:30–37
5. Lange K (2013) *Optimization, 2nd Edition*. Springer
6. Lange K (2016) *MM Optimization Algorithms*. SIAM