MOMENT MATCHING/MAXIMUM ENTROPY OPTIMIZATION

M. Milman, F. Jiang, R. Jelliffe, X. Wang

1. Introduction. This paper considers the solution of a large scale entropy maximization problem motivated by a pharmacokinetic application to optimal drug therapy. A newly developed "Multiple Model (MM)" method [1] of individualized drug therapy for patients is based on a control paradigm that uses a discrete joint density such as that given by a nonparametric population pharmacokinetic model. The model's parameter values are assumed to lie in a discrete set of support points that is initially defined from nonparametric population studies that take into consideration such factors as the individual's age, sex, height, weight, renal function, past dosage, and serum concentrations. The dosage regimen to optimally achieve the desired goal is developed as a function of the entire probability joint density of the parameters. As the patient receives the regimen, the entire distribution can then be adjusted based on feedback measurements (serum levels).

The MM control process is thus initialized with the selection of a discrete prior distribution. There are several methods for obtaining this prior, including parametric, semi-parametric and nonparametric population modeling approaches [2–5]. Of these, only the maximum likelihood nonparametric approaches in [4,5] develop the discrete probability distributions required by the multiple model control paradigm. In the event that a parametric or continuous prior is initially provided, it must first be converted into a discrete distribution that approximates the original continuous distribution in some fashion. This is necessary when one wishes to use the MM dosage paradigm on a well–known and useful parametric population model, where the original data has been lost or is not available for nonparametric modeling.

To perform the conversion from a continuous to a discrete distribution, we have taken the approach described below, based on replacing the given distribution with one that shares some of the characteristics of the original one. Specifically, a subset of the moment data can be matched, typically means and variances of the underlying random variables. This can be accomplished in the most "noninformative" or skeptical way, by finding the distribution having the *maximum entropy* that satisfies the moment constraints. Maximum entropy is a method for obtaining the most probable distribution, in a combinatorial sense, that fits the moment data. The concept of entropy arises in many diverse contexts including statistical mechanics [6], information theory [7], and various estimation problems [8].

The mathematical formulation of this optimization problem falls into the general class of convex optimization. The objective functional (the entropy function) is convex, while the moment constraints are linear. The focus of this paper is on developing algorithms to solve this optimization problem, and then using example data, demonstrating their use. A brief summary of the paper follows.

In Section 2 the maximum entropy moment matching problem with equality constraints is introduced. The dual problem is developed, and is shown to be an unconstrained convex programming problem. The important feature of the dual problem is that the number of dual variables is typically significantly less than the number of primal variables. The solutions to both the primal and dual problems are shown to exist and to be unique when the primal problem is feasible. In this case the Kuhn Tucker optimality conditions guarantee the existence of the solution to the dual problem. Eriksson [11] used a Newton scheme for solving for the stationary values of the dual functional. Standard local convergence results show that the iterates converge if the initial value is close enough to the solution. In Section 2 we take advantage of the convexity of the dual problem. We show that the dual problem also has a unique solution, and furthermore show that it is sufficiently well–behaved to admit globally convergent algorithms for its minimization, including descent based and second order trust region methods. The readily available gradient and Hessian of the dual problem are ideally suited for Levenberg–Marquardt type algorithms that utilize the second order information.

However, in the context of the pharmacokinetic problems that motivated our approach, the dual problem does not always have a finite solution. In fact, if the primal problem is not feasible, then the objective functional of the dual problem is unbounded from below. The next section of the paper addresses this situation.

In Section 3 an approach based on inequality constraints is introduced to relax the problem to allow for an underlying sample space that supports distributions which do not exactly satisfy the moment data. A new primal problem is formulated with a scalar quadratic inequality cosntraint, and the dual formulation is again used to solve the problem. Although approaches of this type have been applied to image reconstruction problems [13] to accommodate noise in data, an analysis of the convergence properties of these algorithms has not been conducted to our knowledge. We note that Powell [14] obtained a globally convergent algorithm for an interesting variant of this problem with linear inequality constraints. However his approach was based on exploiting the orthogonality properties of the Fourier matrix, which in turn required a one–to–one correspondence between the number of variables and constraints. The pharmacokinetic applications that motivated our work can easily lead to 10^5 or more primal variables, with relatively few constraints (e.g., less than 100). We show that the dual to the scalar quadratic inequality problem is also a convex programming problem, although this time with a single active bound constraint. The problem is then converted into an unconstrained problem, and global convergence properties of the Levenberg–Marquardt/trust region type algorithms are again proved.

The "curse of dimensionality" incurred by the number of parameter values motivates an "on the fly" implementation of the algorithms which circumvents the need for storing the very large matrices that appear. Section 4 briefly discusses this implementation.

Section 5 contains an example pharmacokinetic application involving developing discrete distributions for the parameters of a three compartment model of the drug digoxin. Even this relatively small model requires optimizing 100,000 variables. This particular problem is close to the limit of what is possible on a PC or workstation.

2. Moment Matching Problem. Let a discrete density p be defined on a discrete set $\Gamma \subset R^k$,

$$p(x) = \sum_{i=1}^{N} p_i \delta(x - x^i),$$

where x^i is the k–tuple,

$$x^i = (x_1^i, ..., x_k^i).$$

The entropy of the density p is defined as

$$H(p) = -\sum_i p_{i i}. \tag{2.1}$$

The entropy of a distribution is a measure of its randomness. For example, without any constraints (other than $p_i \geq 0$ and $\sum p_i = 1$) it is easily seen that H is maximized by taking $p_i = 1/N$, and is minimized by choosing $p_r = 1$ for some $1 \leq r \leq N$ with $p_i = 0$ for all $i \neq r$.

The problem we address here is maximizing the entropy of the distribution with some additional constraints on the moments of the distribution. In the example above, the only constraint was derived from requiring p to be a probability density. Now suppose we are also given a set of moments $\{m_{r_j}\}_{r,j}$ where m_{r_j} denotes an r^{th} moment

$$m_{r_j} = \sum_i x^i_{m_1} \cdots x^i_{m_r} p(i).$$

The problem we pose is to determine a density p^* with the property that

$$m_{r_i} = \sum_j x^j_{m_1} \cdots x^j_{m_r} p^*(j), \tag{2.2}$$

and

$$H(p^*) \geq H(q) \tag{2.3}$$

for any other density satisfying the moment constraints (2.2). To be precise about the constraints implicitly imposed by requiring p^* to be a density we have in addition to (2.2),

$$p^*_i \geq 0, \quad \text{for all} \quad i = 1, ..., N, \tag{2.4a}$$

and

$$\sum_i p^*_i = 1. \tag{2.4b}$$

Now we introduce the matrix T to represent the linear constraints (2.2) and (2.4b). To incorporate (2.4b) we require the first row T to consist of ones, i.e., $T(1, j) = 1, j = 1, ..., N$. Each constraint of the form (2.2) corresponds to a row of T, say row $r_i + 1$,

$$T(r_i + 1, \cdot) = [x^1_{m_1} \cdots x^1_{m_r} \quad x^2_{m_1} \cdots x^2_{m_r} \quad \cdots \quad x^N_{m_1} \cdots x^N_{m_r}].$$

The constraints (2.2), (2.4a), and (2.4b) can now be expressed as

$$Tp = \tilde{m}, \quad \tilde{m} = [1 \quad m^t]^t; \quad p_i \geq 0.$$

The maximum entropy optimization problem is formulated as

$$\max_p H(p) \quad \text{subject to} \quad Tp = \tilde{m}, \quad p_i \geq 0. \tag{2.5}$$

This problem has, as we shall see, some nice features that make it very amenable to solution, even for extremely large data sets.

The first observation in this regard is that $-H$ is a *strictly* convex functional. This follows from noting that $d^2f/dx^2 > 0$ on $x > 0$ where $f(x)$ is defined as $f(x) = xlogx$. Since $-H$ is the sum of such functions, it is strictly convex. Assuming that the problem (2.5) is feasible, the constraint set is also convex since $C_1 = \{p' : Tp' = \tilde{m}\}$ is affine and $C_2 = \{p : p_i \geq 0\}$ is trivially convex. Furthermore $C_1 \cap C_2$ is compact since $C_1 \cap C_2$ is clearly bounded ($\sum p_i = 1$ together with $p_i \geq 0$), and C_2 and C_1 are both closed ($C_1 = T^{-1}\{\tilde{m}\}$). Therefore (2.5) is guaranteed to have a solution. Also, because of strict convexity this solution is unique.

So far we have established that the primal problem (2.5) has a unique solution. This is still a problem with constraints, however. We will next show how these constraints can be removed by working with the dual to (2.5). We begin by introducing the Lagrangian

$$L(p, \lambda, \mu) = -H(p) - < \lambda, Tp - \tilde{m}) > - < \mu, p >,$$

where $\lambda \in R^d$, $\mu \in R^N$. Here d is the number of equality constraint, which equals one plus the number of moment constraints. Because the constraints are regular (they are in fact linear), the Kuhn–Tucker [9] necessary conditions for optimality are

$$\nabla_p L = 0, \tag{2.6a}$$

$$\mu_i p_i = 0, \quad i = 1, ..., N, \tag{2.6b}$$

together with the constraints

$$Tp = \tilde{m}, \quad p_i \geq 0, \quad i = 1, ..., N. \tag{2.6c}$$

And because the problem is convex, the K–T conditions above are also *sufficient.* Thus we can assert that this system has a unique solution p^*. With the condition that T has full rank we can also assert that the optimal multipliers λ^*, μ^* are also unique. To see this, examine more closely the vanishing of the gradient $\nabla_p L$:

$$\begin{aligned} \nabla_p L &= -\nabla_p H - < T^t\lambda, \cdot > - < \mu, \cdot > \\ &= < v, \cdot > - < T^t\lambda, \cdot > - < \mu, \cdot > . \end{aligned} \tag{2.7}$$

Thus $\nabla_p \ L = 0$ implies $v = T^t\lambda + \mu$ where v has coordinates $v_i = 1 + logp_i$. From this we see that $p_i^* > 0$ for all i since otherwise the multipliers λ, μ cannot be finite. This implies that the Kuhn–Tucker condition (2.6b) is $\mu_i = 0$. Since $\nabla_p \ L = 0$, taking exponentials yields the relationship

$$p_i^* = exp\{< e_i, T^t\lambda^* > -1\}, \tag{2.8}$$

where e_i denotes the vector with a 1 in the i^{th} coordinate and zero elsewhere. Now it is also clear that λ^* is unique because T has full rank and p^* is unique. (Note that since T has full rank, $\lambda_1 \neq \lambda_2$ implies $T^t\lambda_1 \neq T^t\lambda_2$. Thus for some index i, $< e_i, T^t\lambda_1 > \neq < e_i, T^t\lambda_2 >$. Hence, the uniqueness of p^* now implies that λ^* is unique.) This discussion is summarized in

Lemma 2.1. Suppose the primal problem (2.5) is feasible, and T has full rank. Then the primal problem has a unique solution $p^* > 0$, and the Kuhn–Tucker system of equations (2.6a)–(2.6c) has a unique solution.

Having established these facts we can move on to the dual problem. The dual result for the convex problem (2.5) after the analysis above is that (p^*, λ^*) solves the optimization problem

$$\max_{p,\lambda} L(p, \lambda), \tag{2.9a}$$

subject to the constraint

$$\nabla_p L = 0. \tag{2.9b}$$

(Recall that the inequality constraint $p_i \geq 0$ is inactive so that associated muliplier μ is zero.) Solving (2.9b) leads to relation (2.8) as established before. Next observe that

$$\begin{aligned} < \lambda, Tp > &= < T^t\lambda, p > \\ &= \sum_i < e_i, T^t\lambda > p_i \\ &= \sum_i < e_i, T^t\lambda > exp\{< e_i, T^t\lambda > -1\}. \end{aligned} \tag{2.10}$$

Now substituting (2.8) into (2.9a) and keeping (2.10) in mind gives

$$\begin{aligned} L(p,\lambda) &= \sum_i exp\{< e_i, T^t\lambda > -1\}[< e_i, T^t\lambda > -1] - < \lambda, Tp - \tilde{m} > \\ &= -\sum_i exp\{< e_i, T^t\lambda > -1\} + < \lambda, \tilde{m} > . \end{aligned}$$

Writing $h(\lambda) = L(p,\lambda)$ where p is given in (2.8), the dual problem (2.9) reduces to the unconstrained problem

$$\min_{\lambda \in R^d} h(\lambda) = \sum_i exp\{< e_i, T^t\lambda > -1\} - < \lambda, \tilde{m} > . \tag{2.11}$$

Note that not only have the constraints been removed from the problem, but in many cases the dimensionality has also been reduced. This reduction can be very significant when the number of support points of the density p (which is the cardinality of the set Γ) greatly exceeds the number of moment constraints. Although this will always be the case in the moment matching problem we solve, this condition also holds true for the image reconstruction problems generated from interferometric imaging, for example.

In general we cannot expect the dual problem to be convex. However, this is the case here, as we shall next establish. This claim is most easily verified by taking derivatives and showing that the Hessian is nonnegative definite. We will see that the gradient and Hessian of h have very simple forms.

First note that

$$\frac{\partial h}{\partial \lambda_s} = \sum_i T_{si} \exp\{\sum_j T_{ji}\lambda_j - 1\} - m_s. \tag{2.12}$$

From this it follows that

$$\frac{\partial^2 h}{\partial \lambda_s \partial \lambda_t} = \sum_i T_{si} T_{st} \exp\{\sum_j T_{ji}\lambda_j - 1\}. \tag{2.13}$$

Recalling that

$$p_i = \exp\{< e_i, T^t\lambda > -1\},$$

we arrive at succint forms for the gradient and Hessian of h:

$$\nabla_\lambda h = Tp - \tilde{m}, \tag{2.14}$$

and

$$\nabla^2_\lambda h = TXT^t, \tag{2.15}$$

where X is the diagonal matrix with $X_{ii} = p_i$. Thus $\nabla^2_\lambda h$ is nonnegative definite, and h is consequently convex. These simple forms for the gradient and Hessian were first derived in [11].

From (2.14) note that $\nabla_\lambda h$ vanishes precisely when the constraints from the *primal* problem are satisfied. And since the solution to the *primal* problem is unique, we can show that the solution to the *dual* problem must also be unique. To verify this, observe first from (2.8) and the assumption that T has full rank, that p and λ are in one–to–one correspondence. By convexity of the *dual* problem, the K–T conditions are both necessary and sufficient. Hence, if (2.14) vanishes for two values of λ, say λ_1 and λ_2, then the two associated densities, p_1 and p_2, are both solutions to the *primal* problem. By uniqueness $p_1 = p_2$. And because of the unique correspondence from (2.8), it follows that $\lambda_1 = \lambda_2$. Therefore we can state

Theorem 2.2. Under the hypotheses of Lemma 2.1, the dual problem (2.11) has a unique solution.

The next result has significance with respect to the implementation of algorithms for solving for the minimum of h in (2.11).

Proposition 2.3. If Problem (2.5) is feasible, then the set

$$L_0 = \{\lambda : h(\lambda) \leq h(\lambda_0)\}$$

is bounded for any λ_0.

Proof. Because (2.5) is feasible $h(\lambda)$ is bounded from below [9]. Thus the set

$$S = \{\lambda : \quad <\lambda, \tilde{m}> \quad > 0, \quad <Te_i, \lambda> \leq 0\}$$

is empty. By Farkas' lemma [10], there exist $\alpha_i \geq 0$ such that

$$\tilde{m} = \sum_i \alpha_i Te_i .$$

So now we may write

$$h(\lambda) = \sum_i \exp\{<Te_i, \lambda> -1\} - \sum_i \alpha_i <Te_i, \lambda> . \tag{2.16}$$

Let λ_N be any sequence with $|\lambda_N| \to \infty$, and let $z_N = \lambda_N / |\lambda_N|$. Hence,

$$h(\lambda_N) = \sum_i \exp\{|\lambda_N| <Te_i, z_N> -1\} - |\lambda_N| \sum_i \alpha_i <Te_i, z_N> . \tag{2.17}$$

Now z_N has a convergent subsequence $z_{N_k} \to z^*$ with $|z^*| = 1$. Since $z^* \neq 0$ and T has full rank, the two sets of indices, $I_+ = \{i : \quad <Te_i, z^*>> 0\}$ and $I_- = \{i : \quad <Te_i, z^*>< 0\}$ cannot be be simultaneously empty. Next define the functions $h_\pm$,

$$h_\pm = \sum_{i \in I} exp\{<Te_i, \lambda> -1\} - \sum_{i \in I} \alpha_i <Te_i, \lambda>, \tag{2.18}$$

and note that $h = h_+ + h_-$. From (2.17) and (2.18) it is clear that if I_+ (resp. I_-) is nonempty, then

$$\lim_k h_+(\lambda_{N_k}) = +\infty \quad (\text{resp.} \quad \lim_k h_-(\lambda_{N_k}) = +\infty)$$

Hence L is bounded.///

This result together with the strict convexity of (2.11) implies global convergence for several classes of algorithms. For example, from the form of the hessian in (2.15) it follows that $\nabla_\lambda h$ is uniformly continuous on L_0, thus establishing the classical Goldstein criteria for global convergence for the class of descent algorithms with line search. The proposition also establishes global convergence for a class of restricted stepsize second order methods.

However, it is generally not known *a priori* whether the initial problem is feasible. The next section introduces a modification to insure that a feasible problem is always defined.

3. The Inequality Constrained Problem. In the pharmacokinetic applications that motivated the problem formulation, it is seldom known *a priori* whether the primal problem is feasible. Typically the moment data is provided, but the underlying sample space is not. In this case it is necessary to define the sample space. An obvious way to do this is simply to assume a uniform grid for each random variable. This does not assure feasibility, and one alternative is to relax the problem to allow for this possibility. We remark that similar approaches have been implemented in image reconstruction problems where the constraint has the form

$$T * p - m = 0,$$

where now $*$ denotes (two–dimensional) convolution, p represents the source (which is of positive intensity), and m represents the measured image. Typically the image is corrupted by noise and the constraint is not satisfied, analogous to the moment matching problem [13, 14].

To circumvent this difficulty, an additional inequality constraint is introduced, converting (2.5) into

$$\min_{p} -H(p) = \sum p_i \log(p_i), \tag{3.1}$$

subject to the constraints

$$Tp - m + N = 0, \tag{3.2}$$

$$T_0 p - m_0 = 0, \tag{3.3}$$

$$p \geq 0, \tag{3.4}$$

$$\Omega - |N|^2 \geq 0, \tag{3.5}$$

where Ω is a positive scalar chosen to make the problem feasible and N is a slack variable. Here again $T_0 = [1 \cdots 1]$ and $m_0 = 1$. T_0 and m_0 may be chosen to normalize solutions in other ways; in image reconstruction applications m_0 represents the total flux, for example. Ω can be determined from the quadratic programming problem defined by minimizing $|Tp - m|^2$ subject to the constraints (3.3)–(3.4). Then any value of Ω greater than the minimum solution yields a feasible problem.

An alternative to this is to attempt to solve the equality constrained problem. By Farkas' lemma, if the primal problem has no solution, then in the course of solving the dual problem, a direction λ will be determined along which the cost function becomes unbounded ($h(\lambda) \to -\infty$). As soon as this direction is found, the solution grows very rapidly. However, before this event occurs, any $\nabla h = Tx - m'$ can be interpreted as a solution to a *neighboring* primal problem with m' replacing m. Choosing $\Omega = |m - m'|^2$ then leads to a feasible problem.

The new problem (3.1)–(3.5) no longer consists only of linear constraints because of the appearance of (3.5), but it is still convex. And for sufficiently large Ω it is feasible. If Ω is chosen too large, the constraint (3.5) may no longer be active, in which case the uniform distribution solves the problem.

We again look to the dual to solve the optimization problem. As before we begin by forming the Lagrangian $L(p, N, \lambda, \lambda_0, \mu, \mu_\Omega)$,

$$L = -H(p) - < \lambda, Tp - m + N > - < \lambda_0, T_0 p - m_0 > - < \mu, p > -\mu_\Omega[\Omega - |N|^2]. \tag{3.6}$$

Let

$$T_e = \begin{pmatrix} T \\ T_0 \end{pmatrix}, \quad \lambda_e = \begin{pmatrix} \lambda \\ \lambda_0 \end{pmatrix} \quad m_e = \begin{pmatrix} m \\ m_0 \end{pmatrix}. \tag{3.7}$$

Then the Kuhn–Tucker conditions are

$$\frac{\partial L}{\partial p} = 0, \tag{3.8a}$$

$$\frac{\partial L}{\partial N} = 0, \tag{3.8b}$$

$$\mu_i p_i = 0, \quad \mu_i \geq 0, \tag{3.8c}$$

$$\mu_\Omega(\Omega - |N|^2) = 0, \quad \mu_\Omega \geq 0. \tag{3.8d}$$

As before we can deduce that $\mu_i = 0$, so that the first Kuhn–Tucker condition becomes

$$v = T_e^T \lambda_e, \quad \text{with components} \quad v_i = 1 + logp_i. \tag{3.9}$$

Assuming (3.5) is active and strict complementarity holds, (3.8b) implies

$$N = \frac{1}{2\mu_\Omega}\lambda. \tag{3.10}$$

Note that if (3.5) is not active or if strict complementarity does not hold, then $\mu_\Omega = 0$, and hence, $\lambda = 0$ so that the uniform distribution solves (3.1)–(3.5).

Now we consider the dual formulation. We shall assume that $\mu_\Omega > 0$ in what follows, since the problem is trivial otherwise. The dual problem is

$$\max_{\mu_\Omega > 0, \lambda_e} L \quad \text{subject to} \quad \frac{\partial L}{\partial p} = 0 \quad \text{and} \quad \frac{\partial L}{\partial N} = 0. \tag{3.11}$$

Using the relationship for N in (3.10) we find that

$$L = -\sum \exp\{< e_i, T_e^T \lambda_e > -1\} + < \lambda_e, m_e > -\frac{|\lambda|^2}{4\mu_\Omega} - \mu_\Omega \Omega, \quad \mu_\Omega > 0. \tag{3.12}$$

Previously it was observed that when the primal problem was not feasible (i.e., no solution to $Tp = m$ with $p \geq 0$), then $h(\lambda) \to -\infty$ for some direction λ where the dual function h is defined in (2.11). We shall see shortly that the presence of the quadratic term $|\lambda|^2$ mitigates this possibility for Ω sufficiently large.

Converting (3.12) to a minimization problem we have

$$\min L = \sum \exp\{< e_i, T_e^T \lambda_e > -1\} - < \lambda_e, m_e > + \frac{|\lambda|^2}{4\mu_\Omega} + \mu_\Omega \Omega, \quad \mu_\Omega > 0. \tag{3.13}$$

One alternative to solving this comes from noting that $\nabla_{\mu_\Omega} L = 0$ implies

$$\mu_\Omega = \frac{|\lambda|}{2\sqrt{\Omega}}. \tag{3.14}$$

Hence, we might equivalently solve

$$\min_{\lambda_e} h(\lambda_e) + \sqrt{\Omega}|\lambda|, \tag{3.15}$$

as an alternative to (3.13). But (3.15) is undesirable because $|\cdot|$ is not smooth at the origin, and we will not pursue this possibility.

However, we note from (3.13) and (3.15) that

$$L(\lambda_e, \mu_\Omega) \geq h(\lambda_e) + \sqrt{\Omega}|\lambda|.$$

Now let

$$\sigma = \inf_{|\lambda_e|=1} \{| < \lambda_e, m_e > | : \quad < \lambda_e, Te_i > \leq 0 \quad \text{and} \quad < \lambda_e, m_e >> 0\}$$

Then for $\Omega \geq \sigma$, it follows that $\inf_{\lambda_e} h(\lambda_e) + \sqrt{\Omega}|\lambda| > -\infty$, so that the dual problem has a solution. However, since the constraints are not linear, solvability of the dual does not guarantee that the primal problem is feasible. We will proceed on the assumption that Ω has been chosen large enough so that the dual problem has a solution.

Theorem 3.1. The dual minimization problem (3.13) is a convex programming problem.

Proof. Recalling that $N = \lambda/2\mu_\Omega$, we have

$$\nabla_{\lambda_e} L = T_e p - m_e + \begin{pmatrix} 0 \\ N \end{pmatrix}. \tag{3.16}$$

Also, again using $N = \lambda/2\mu_\Omega$,

$$\nabla_{\mu_\Omega} L = \Omega - \mu_\Omega |N|^2. \tag{3.17}$$

From these we easily develop the Hessian of L as

$$\frac{\partial^2 L}{\partial \lambda_e^2} = T^T X T + D, \quad \text{where} \quad X = diag(p_1, ...) \quad D = diag(0, 1/2\mu_\Omega, 1/2\mu_\Omega ..., 1/2\mu_\Omega), \tag{3.18}$$

$$\frac{\partial^2 L}{\partial \lambda_e \partial \mu_\Omega} = \begin{pmatrix} 0 \\ -N/\mu_\Omega \end{pmatrix}, \tag{3.19}$$

$$\frac{\partial^2 L}{\partial \mu_\Omega^2} = \frac{2|N|^2}{\mu_\Omega}. \tag{3.20}$$

Thus we have

$$\nabla^2 L = \begin{pmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{pmatrix}$$

where L_{11} is given by (3.18), L_{12} is given by (3.19), $L_{21} = L_{12}^T$, and L_{22} is given by (3.20).

Next we will show that $\nabla^2 L$ is positive definite. We compute for a partitioned vector $(\alpha^t \quad \beta)$:

$$(\alpha^t \quad \beta) \nabla^2 L \begin{pmatrix} \alpha \\ \beta \end{pmatrix} =< L_{11}\alpha, \alpha > +2 < L_{12}\beta, \alpha > + L_{22}\beta^2. \tag{3.21}$$

This quadratic form is minimized when

$$\alpha = -L_{11}^{-1} L_{12} \beta. \tag{3.22}$$

Letting $Q(\beta)$ denote the value of (3.21) with the relationship in (3.22), we obtain

$$Q(\beta) = L_{22}\beta^2 - \beta^2 < L_{11}^{-1} L_{12}, L_{12} > . \tag{3.23}$$

If T has full rank, then as before $TXT^T > 0$. Hence, for some $\epsilon > 0$, $TXT^T > \epsilon I$ and

$$L_{11} > \begin{pmatrix} \epsilon I & 0 \\ 0 & D + \epsilon I \end{pmatrix}.$$

Hence,

$$L_{11}^{-1} < \begin{pmatrix} 1/\epsilon I & 0 \\ 0 & (D+\epsilon I)^{-1} \end{pmatrix}.$$

Therefore,

$$< L_{11}^{-1}N', N' >< \frac{2}{\mu_\Omega + \epsilon\mu_\Omega^2}|N|^2, \quad \text{where} \quad N' = \begin{pmatrix} 0 \\ N \end{pmatrix}. \tag{3.24}$$

Recalling the definitions of L_{12} and L_{22}, it follows that $Q(\beta) > 0$, thus establishing that $\nabla^2 L > 0$ and that L is convex for $\mu_\Omega > 0$. Hence, the dual problem is also a convex programming problem with the single constraint $\mu_\Omega > 0$. ///

The constraint $\mu_\Omega > 0$ can be removed by the change of variables

$$\mu_\Omega = \exp(v), \tag{3.25}$$

so that (3.13) becomes the unconstrained minimization problem

$$\min_{\lambda_e, v} H(\lambda_e, v) \quad \text{where} \quad H(\lambda, v) = h(\lambda_e) + \frac{|\lambda|^2}{4}\exp(-v) + \exp(v)\Omega. \tag{3.26}$$

Computing ∇H we have

$$\nabla_{\lambda_e} H = T_e p - m_e + \begin{pmatrix} 0 \\ N \end{pmatrix}, \tag{3.27}$$

and

$$\begin{aligned} \nabla_v H &= -\frac{|\lambda|^2}{4}\exp(-v) + \exp(v)\Omega \\ &= -\frac{|\lambda|^2}{4\mu_\Omega} + \mu_\Omega\Omega. \end{aligned} \tag{3.28}$$

Comparing (3.27)–(3.28) with (3.16)–(3.17) we see that $\nabla H = 0$ if and only if $\nabla L = 0$. If μ_Ω is nonzero (which is our working assumption to keep the problem nontrivial), then the vanishing of ∇H is a sufficient condition for a global minimum.

Although we have eliminated the constraint $\mu_\Omega > 0$, we have lost some of the convexity of the problem, as H is no longer convex. However, computing the Hessian of H we obtain

$$\frac{\partial^2 H}{\partial \lambda_e^2} = T^T X T + D, \quad X = diag(p_1, ...) \quad D = diag(0, 1/2\mu_\Omega, 1/2\mu_\Omega ..., 1/2\mu_\Omega), \tag{3.29a}$$

$$\frac{\partial^2 H}{\partial \lambda_e \partial v} = -N, \tag{3.29b}$$

$$\frac{\partial^2 H}{\partial v^2} = \mu_\Omega(|N|^2 + \Omega). \tag{3.29c}$$

And, arguing as before in (3.22)–(3.25) we can show that a sufficient condition for $\nabla^2 H$ to be non-negative definite is that $|N|^2 \leq \Omega$. Hence, $\nabla H = 0$ satisfies the second order sufficient conditions for a minimum.

From the discussion above, minimizing H is an effective means for solving (3.1)–(3.5). Having the gradient and Hessian of H from (3.27)–(3.29), leads to algorithms having the same form discussed in Section 2 for the equality constrained problem. To ensure global convergence, we next establish that the dual function H is bounded from below for appropriate values of Ω, even when the original primal problem (2.5) is not feasible.

Theorem 3.2. If the problem (3.1)–(3.5) is feasible with solution other than the uniform distribution, there exists (λ_e^*, v^*) such that the level set

$$L = \{(\lambda_e, v): \quad H(\lambda_e, v) \leq H(\lambda_e^*, v^*)\}$$

is bounded.

Proof. Again let

$$\sigma = \inf_{|\lambda_e|=1} \{| < \lambda_e, m_e > | : \quad < \lambda_e, Te_i > \leq 0 \quad \text{and} \quad < \lambda_e, m_e >> 0\}.$$

Then for $\Omega \geq \sigma$, it follows that $\inf_{\lambda_e} h(\lambda_e) + \sqrt{\Omega}|\lambda| > -\infty$, and H is bounded from below since

$$H(\lambda_e, v) \geq h(\lambda_e) + \sqrt{\Omega}|\lambda|.$$

If Ω^* leads to a feasible solution to (3.1)–(3.5), then $\Omega^* \geq \sigma$. Trivially if $|\lambda_e^N| \to \infty$, it follows from the inequality above that $H(\lambda_e^N, v) \to +\infty$. The only possibility for L to be unbounded is then if $v^N \to -\infty$ with $\lambda = 0$. Thus we assume that λ_e has the form

$$\lambda_e = \begin{pmatrix} \lambda_0 \\ 0 \end{pmatrix}$$

Minimizing $h(\lambda_e)$ over λ_e of the form above, we find that $\lambda_0 = \log(e/N)$. Hence,

$$\lim_{N\to\infty} H(\lambda_e, v_N) \geq h\begin{pmatrix} \lambda_0 \\ 0 \end{pmatrix} \quad \text{with} \quad \lambda_0 = \log(e/N).$$

Now for any unit vector y,

$$h(\begin{pmatrix} \lambda_0 \\ 0 \end{pmatrix} + ty) + t|\Pi y|\sqrt{\Omega^*} = h(\begin{pmatrix} \lambda_0 \\ 0 \end{pmatrix}) + t < \nabla h, y > +t|\Pi y|\sqrt{\Omega^*},$$

where Π denotes the projection $\Pi : (u_1, \cdots u_M) \to (u_2, \cdots u_M)$. But recall that

$$\nabla h = Tp - m_e,$$

with

$$\begin{aligned} p_i &= \exp\{< e_i, T^t \begin{pmatrix} \lambda_0 \\ 0 \end{pmatrix} > -1\} \\ &= 1/N. \end{aligned}$$

If $|Tp - m_e| \leq \sqrt{\Omega^*}$ the uniform distribution optimizes the problem. If not, by choosing $y = -(Tp - m_e)$ it follows that for ϵ sufficiently small

$$\lim_{N\to\infty} H(\lambda_e, v_N) > h(\begin{pmatrix} \lambda_0 \\ 0 \end{pmatrix} + \epsilon y) + \epsilon|y|\sqrt{\Omega^*}.$$

And we may choose

$$\lambda^* = \begin{pmatrix} \lambda_0 \\ 0 \end{pmatrix} - \epsilon(Tp - m_e), \quad v^* = \log(\frac{|Tp - m_e|}{2\sqrt{\Omega^*}})$$

in the statement of the theorem.///

The significance of this result is that it establishes convergence of gradient and trust region algorithms for the dual problem whenever the primal problem (3.1)–(3.5) is feasible.

Another way of obtaining solutions when the underlying sample is not defined is to construct a sample space to guarantee that the moment constraints can be satisfied. This construction of problems in which the means and variances are prescribed was developed in [12].

4. Algorithms. The algorithms that have been implemented are based on a modification of the fundamental Newton iteration. It will be useful to write here the Newton iteration for problem (2.11). For brevity we shall write $g = \nabla_\lambda h$ and $G = \nabla_\lambda^2 h$. Using a Newton scheme to solve for the necessary (and sufficient) condition $g = 0$ leads to the iteration

$$\lambda^{(k+1)} = \lambda^{(k)} - G_k^{-1}(g_k), \tag{4.1}$$

where $g_k = g(\lambda^{(k)})$ and $G_k = G(\lambda^{(k)})$. (Observe that G^{-1} can fail to exist only if $R(T^t) \cap N(X)$ is not empty. So for example G will be invertible if $p_i \neq 0$ for all i.) Returning to the definition of the associated iterate $p_i^{(k)}$ defined from (2.13) and (2.8) we have

$$\begin{aligned} p_i^{(k)} &= \exp\{\sum_j T_{ji}\lambda_j^{(k)} - 1\} \\ &= \exp\{\sum_j T_{ji}[\lambda_j^{(k-1)} - G_{k-1}^{-1}(g_{k-1})] - 1\} \\ &= p_i^{k-1} \exp\{-\sum_j T_{ji}G_{k-1}^{-1}(g_{k-1})\} \\ &= p_i^{k-1} \exp\{-\sum_j T_{ji}G_{k-1}^{-1}[Tp^{(k-1)} - \tilde{m}]\}. \end{aligned} \tag{4.2}$$

Note that although we are solving the *dual* problem, the update is written entirely in terms of the *primal* variables. From (2.14) and (2.15) the gradient and Hessian are also developed in terms of the *primal* variables.

The Newton iteration (4.1)–(4.2) was derived in [11]. Note that convergence is achieved when $Tp^* = \tilde{m}$, in accordance with Theorem 2.2 (cf (2.14)). The Newton iteration converges quadratically so long as the initial value is sufficiently close to the solution. The trust region modification of the Newton algorithm coupled with Proposition 2.3 insures global convergence [9]. A Levenberg–Marquardt variation of the trust region algorithm has been implemented in our pharmacokinetic applications.

The iteration for solving the inequality constrained problem (3.1)–(3.5) is based on the same iteration as (4.1)–(4.2) above, but using the gradient and Hessian developed in (3.27)–(3.29).

The main bottleneck in implementing these algorithms arises from computing the matrix product

$$G = TXT^T. \tag{4.3}$$

Here, $T = [T_{ij}]$ is a "fat" matrix of real numbers of size $m \times n$ with $m \ll n$, and $X = \text{diag}\{p_1, ..., p_n\}$ is a diagonal matrix of real numbers of size $n \times n$. The resulting matrix has size $m \times m$, corresponding to the total number of constraints in the problem and can be considered as a relatively small matrix.

The key computational issue is how to handle the constraint matrix T. Let q be the number of problem parameters and g be the number of computational grid points for each parameter. The number n of columns of T is equal to g^q, while the number m of rows of T is the number of constraints prescribed for the problem. The development in the sequel focusses on the case in which the means and cross correlations of the parameters have been defined, leading to $m = (q^2+q)/2+1$. Other moment constraints can be handled similarly.

Ideally, elements of the constraint matrix are calculated once and stored in an array so that they can be used subsequently without re-calculation. However, due to the size of the matrix, storing its elements requires a substantial amount of memory. Table 4.1 below lists the memory requirement M (in MegaBytes) for the matrix T for examples of different numbers of parameters. M is calculated according to the following formula,

$$M = s * n * m/10^6 = s * g^q * m/10^6$$

where s is the number of bytes that each floating point number takes. In the two tables, s is assumed to be 4 and g is 10.

Clearly, as the number of parameters q becomes greater than 6, the memory requirement M alone exceeds capabilities that a single state-of-the-art PC or workstation can offer. Indeed, 7000 to 20000 PC nodes are needed to make up the required memory space for the q = 10 case. Therefore, generating elements of the matrix on-the-fly is critical to solving problems having more than 6 parameters.

In the table below

- q= number of parameters
- $m = (q^2 + q)/2 + q + 1$ is the number of rows
- $n = g^q$ is the number of columns
- M = megaBytes of memory

q	4	5	6	7	8	9	10
m	5	6	7	8	9	10	11
n	10^4	10^5	10^6	10^7	10^8	10^9	10^{10}
M	0.36	4.4	52	600	6,800	76,000	840,000

Table 4.1. Memory requirements for constraint matrix

The matrix T depends on the grid sampled in the parameter space. For each $k = 0, 1, ..., q-1$, let $X_k = \{x_k^0, ..., x_k^{g-1}\} \subset R$ be the set of supporting points (real numbers) for the k-th parameter. The grid formed by these sets X_k is

$$\Omega = X_0 \times X_1 \times \cdots \times X_{q-1} \subset R^q,$$

and the grid points in $\Omega = \{x^0, ..., x^{n-1}\}$ can be enumerated as

$$x^j = (x_0^{j_0}, x_1^{j_1}, ..., x_{q-1}^{j_{q\ 1}}),$$

where $j = 0, 1, ..., n-1$, and $(j_0, ..., j_{q-1})$ is the q-dimensional representation of j with the base g; that is,

$$j = j_0 + j_1 g + \cdots + j_{q-1} g^{q-1}.$$

It is very important to note that the enumeration above establishes a one-to-one relationship between any grid point and its parameter coordinates. With this relationship, the constraint matrix T of size $m \times n$ can be defined as follows.

- For $i = 0$ and $j = 0, 1, ..., n-1$,

$$T(i, j) = 1.$$

- For $i = 0, ..., q-1$ and $j = 0, 1, ..., n-1$,

$$T(i+1, j) = x_i^{j_i}.$$

- For $i = 0, ..., q(q-1)/2 - 1$ and $j = 0, 1, ..., n-1$,

$$T(i + 2 * q + 1, j) = x_{i_0}^{j_{i_0}} x_{i_1}^{j_{i_1}}.$$

Here, for each $i = 0, 1, ..., q(q-1)/2 - 1$, (i_0, i_1) with $i_0 > i_1$ is the 2-dimensional representation of i with base q; that is,

$$i = i_0 + i_1 * q.$$

In this case, the number of rows is $m = q * (q-1)/2 + 2 * q + 1 = q * (q+1)/2 + q + 1$.

For example, when $q = 2$, $g = 3$ and

$$\begin{aligned} X_0 &= \{x_0^0, x_0^1, x_0^2\} \\ X_1 &= \{x_1^0, x_1^1, x_1^2\}, \end{aligned}$$

the grid generated from X_0 and X_1 is

$$\begin{aligned} \Omega = \{ & x^0 = (x_0^0, x_1^0), x^1 = (x_0^1, x_1^0), x^2 = (x_0^2, x_1^0), \\ & x^3 = (x_0^0, x_1^1), x^4 = (x_0^1, x_1^1), x^5 = (x_0^2, x_1^1), \\ & x^6 = (x_0^0, x_1^2), x^7 = (x_0^1, x_1^2), x^8 = (x_0^2, x_1^2).\} \end{aligned}$$

The matrix T has the form

$$T = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ x_0^0 & x_0^1 & x_0^2 & x_0^0 & x_0^1 & x_0^2 & x_0^0 & x_0^1 & x_0^2 \\ x_1^0 & x_1^0 & x_1^0 & x_1^1 & x_1^1 & x_1^1 & x_1^2 & x_1^2 & x_1^2 \\ x_0^0 x_0^0 & x_0^1 x_0^1 & x_0^2 x_0^2 & x_0^0 x_0^0 & x_0^1 x_0^1 & x_0^2 x_0^2 & x_0^0 x_0^0 & x_0^1 x_0^1 & x_0^2 x_0^2 \\ x_1^0 x_1^0 & x_1^0 x_1^0 & x_1^0 x_1^0 & x_1^1 x_1^1 & x_1^1 x_1^1 & x_1^1 x_1^1 & x_1^2 x_1^2 & x_1^2 x_1^2 & x_1^2 x_1^2 \\ x_0^0 x_1^0 & x_0^1 x_1^0 & x_0^2 x_1^0 & x_0^0 x_1^1 & x_0^1 x_1^1 & x_0^2 x_1^1 & x_0^0 x_1^2 & x_0^1 x_1^2 & x_0^2 x_1^2 \end{pmatrix}$$

A simple algorithm for computing the entries $T(i, j)$ is a straightforward implementation of the definition above. The problem with this approach is its inefficiency when it is required to compute all the entries of the matrix. This is because each entry needs to perform rather time-consuming decompositions to compute $j's$ coordinates $(j_0, ..., j_{q-1})$ and $i's$ coordinates (i_0, i_1).

When the constraint matrix is used in the maximum entropy computation, all of its entries will be used in a row-by-row manner. For example, when computing the multiplication

$$G = TXT^T$$

where X is a diagonal matrix, a double loop is used. Because computing j's coordinates $(j_0, ..., j_{q-1})$ is time-consuming, it is very desirable to compute $T(s, j)$ and $T(t, j)$ incrementally.

A careful examination on the definition of the constraint matrix T and an incremental relation of $(j + 1)_i$ with j_i leads to an efficient incremental algorithm for generating the entries of T [17].

Computing the Diagonal Matrix D. Let y be a vector of real numbers of length m, and let $b = T^T y$ be the product vector of T^T and y. Then, the definition of the diagonal entries of the diagonal matrix D is

$$D_{kk} = \exp\{b_k - 1\}.$$

To compute the b_k's in an incremental manner, it is necessary to enumerate entries of the matrix T column-wise, as

$$b_k = \sum_{i=1}^{m} T_{ik} y_i.$$

An algorithm for accomplishing this task is found in [17].

Since it is symmetric, the matrix G can be computed from the following code.

```
for s = 0 to m-1 do
   for t = s+1 to m-1 do
      B(s, t) = 0.0
        for j = 0 to n-1 do
          G(s, t) = T(s, j)*X(j, j)*T(t, j)
        end do
        G(t,s)=G(s,t)
      end do
    end do
```

5. Examples. The following examples illustrate the maximum entropy/moment matching method developed herein. The initial data for these problems are contained in the table below. The parameters are derived from presumably normally distributed population data of the drug digoxin [15, 16].

Parameter Number	Range	Mean	Std. Dev
V	0 – 5.0	1.57	3.14e-1
Ks	0 – 1.5e-3	4.51e-4	1.00e-4
Kcp	0 – 1.5	5.60e-1	3.26e-1
Kpc	0 – 5.0e-1	1.50e-1	3.00e-2
Ka	0 – 1.5	6.09e-1	1.22e-1

Table 5.1. Digoxin Population Data

The pharmacokinetic model for digoxin is shown in the block diagram below.

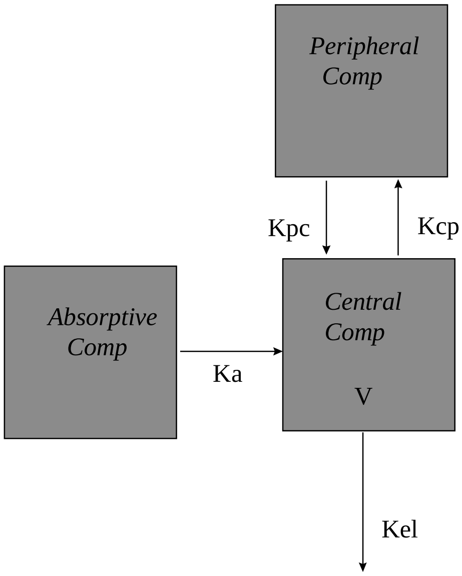

Fig. 5.1 Digoxin Pharmacokinetic Model

The parameters are V, which is the apparent volume of distribution of the central compartment, Kcp the rate constant from the central to peripheral compartment, Kpc the rate constant from the peripheral to central compartment, Ka the absorption rate constant, and Ks, which is the linear term in the renal component of the elimination rate constant Kel; $Kel = Kint + Ks * CCr$ (CCr=creatinine clearance). This three compartment model corresponds to the linear ordinary differential equation:

$$\dot{x} = Ax + Bu,$$

where

$$A = \begin{pmatrix} -(Kel + Kcp) & Kpc & Ka \\ Kcp & -Kpc & 0 \\ 0 & 0 & -Ka \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

The therapeutic objective is often to maintain a specified serum level of the drug, z, in the central compartment by administration of the drug via the control term u. This objective is defined as

$$z = [1/V \quad 0 \quad 0]x.$$

Each possible value of the 5–tuple of parameters defines a model. The probability distribution defines the likelihood of that model. In the multiple model approach it is assumed that one of these models describes the patient. The maximum entropy algorithm is used to initialize the probability distribution based on the population pharmacokinetic model existing at the beginning of therapy.

A uniform grid consisting of 10 points per variable was used to define the underlying sample space. The primal problem in this case consisted of 10^5 variables and 11 constraints using the means and variances from Table 5.1. The algorithm converged for values of $\Omega = 10^{-6}$ and $\Omega = 10^{-10}$ (cf.

(3.5)) and did so in fewer than 25 iterations for each of these values of Ω. The figures below show the convergence history. Note how the convergence in each of these examples is initially linear, but ultimately becomes quadratic, as predicted from theory [9].

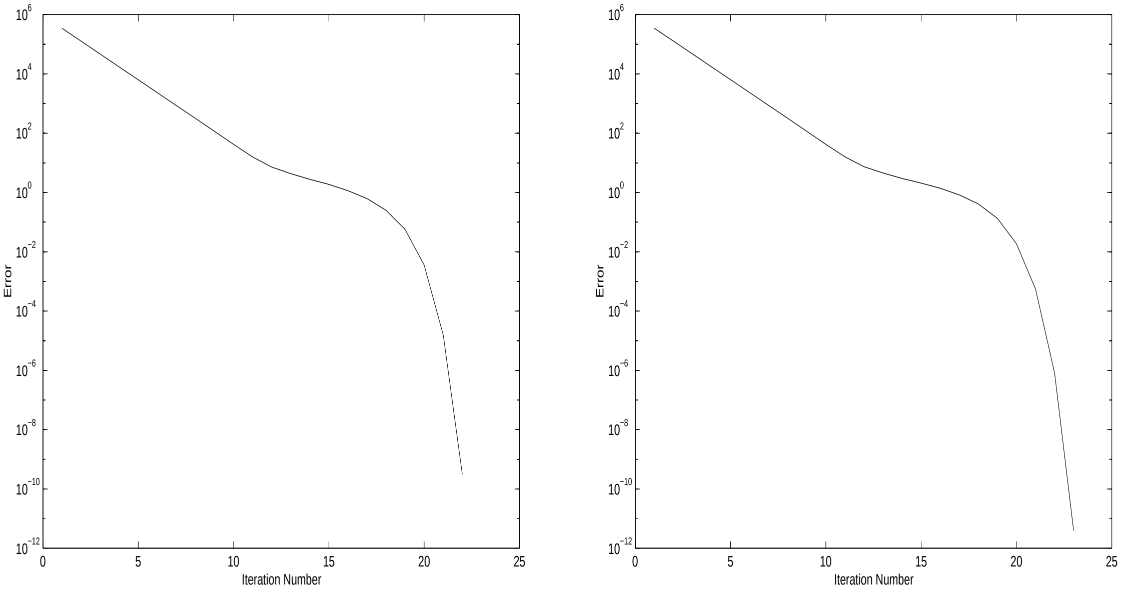

Fig. 5.2: Error Convergence: $\Omega = 10^{-6}$ *(left)* $\Omega = 10^{-10}$ *(right)*

The comparison between the *a priori* estimates and the moments obtained in the distribution generated by the maximum entropy method is given in the table below. The first five rows contain the mean values of the parameters V, Ks, Kcp, Kpc, and Ka, respectively, while the last five rows contain the values of their second moments.

Table 5.2 Moment Comparison

original moment	estimated moment Ω=1.0e-06	estimated moment Ω=1.0e-10
1.5714e+00	1.5713e+00	1.5714
4.5100e-04	7.4362e-04	4.6768e-04
5.6000e-01	5.5964e-01	5.6000e-01
1.5000e-01	1.4952e-01	1.5000e-01
6.0930e-01	6.0897e-01	6.0930e-01
2.5681e+00	2.5681e+00	2.5681e+00
2.1340e-07	7.7982e-07	3.9542e-07
3.2614e-01	3.2646e-01	3.2614e-01
2.3400e-02	2.54004e-02	2.3407e-02
3.8611e-01	3.8638e-01	3.8611e-01

The errors in the final moments are within the bounds defined by Ω; which would yield maximum errors on the order of 10^{-3} for each moment when Ω =1.0e-06, and errors on the order of 10^{-5} when Ω =1.0e-10. Of course it would be wise to scale all of the variables to approximately unity to achieve a more uniform distribution of error. The histories of the entropy values as a function of the iteration number is shown for these two solutions in the figures below. Note that the final entropy is greater, as expected, when Ω is larger.

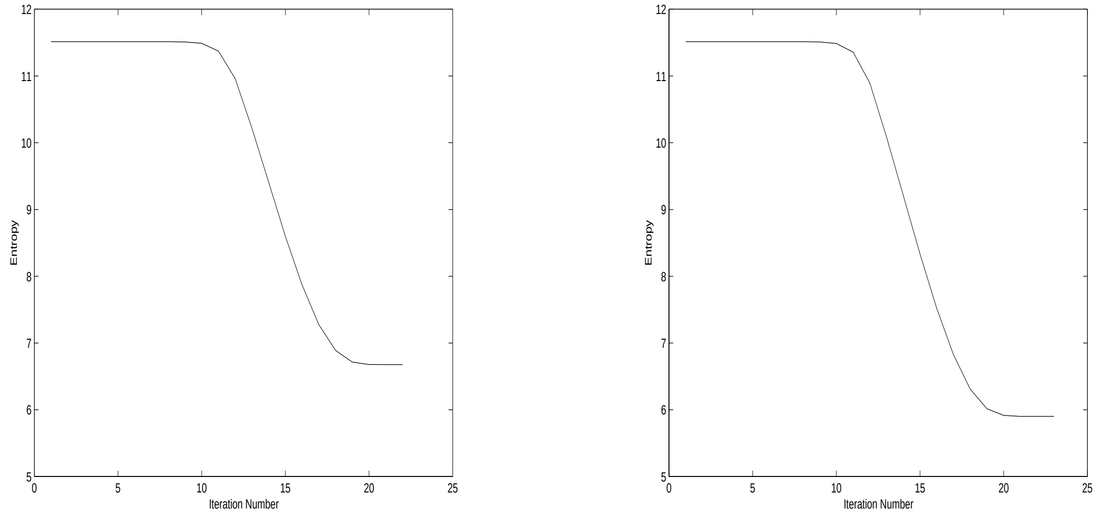

Fig. 5.3: Convergence of Entropy: $\Omega = 10^{-6}$ *(left)* $\Omega = 10^{-10}$ *(right)*

6. Concluding Remarks. Maximum entropy/moment matching methods provide a powerful means for constructing discrete probability distributions from continuous parametric data. This construction has important applications to pharmacokinetic control problems that rely on discrete probability distributions for their implementation such as multiple model adaptive controllers [1]. Although the algorithms developed in this paper were targeted for this application, they are generally applicable to other entropy maximization problems with linear constraints as well, such as those that arise in image reconstruction.

Algorithms based on the theory in Sections 2–3 have been implemented in software programs [12]. The user can select either the equality or inequality constrained options. For the equality constrained option, the user may either define the sample space, or the sample space can be automatically generated via a method in [12] which guarantees feasibility of the problem. If the user wishes to define the sample space, the inequality constrained problem is a better selection of algorithm, since it is generally not known *a priori* whether a solution within a prescribed tolerance can be attained for the equality constrained problem. The software implementation of the inequality constrained algorithm adjusts the value of Ω in (3.5) via a bisection algorithm until a feasible problem is formulated. These algorithms has been successfully applied to problems with as many as 100,000 primal variables, as demonstrated in Section 5.

Current work focuses on extending these programs so that they can accommodate eight or more parameter problems with 10^8 or greater primal variables. The cpu times required to find the maximum entropy distribution for the 5 parameter problem presented in Section 5 were typically on the order of 500 seconds on a Sun Ultra Sparc 1 workstation. Larger problems are prohibitive on workstations, and this motivated the methods described in Section 4 for reducing memory requirements and improving execution times. Further speedup is anticipated by parallelizing the matrix computations in the implementation of the algorithms.

References

[1] D. S. Bayard, M. H. Milman, and A. Schumitzky, Design of dosage regimens: A multiple model stochastic control approach, Int. J. of Bio–Medical Computing, 36, (1994), pp. 103–115.

[2] S. L. Beal and L. B. Sheiner, Estimating population kinetics, CRC Critical Reviews, Bioengineering, 8, (1982), pp. 195–222.

[3] M. Davidian and A. R. Gallant, The nonlinear mixed effects model with a smooth random effects density, Inst. of Statistics Mimeo Ser. No. 2206, North Carolina State Univ., Raleigh, North Carolina.

[4] A. Schumitizky, A nonparametric maximum likelikhood approach to pharmacokinetic population analysis, Proc. 1993 Western Simulation Multiconference–Simulation for Health Care, Soc. of Comp. Sim., San Diego, CA, 95–100.

[5] A. Mallet, A maximum likelihood estimation method for random coefficient regression models, Biometrika, 73, (1986), pp. 645–656.

[6] E. T. Jaynes, Phys. Rev., 108, (1957) pp. 171–190.

[7] C. E. Shannon, Bell System Tech. J., (1948), pp. 379–423.

[8] J. P. Burg, "Maximum Entropy Spectral Analysis", Ph.D. Thesis, Dept. of Geophysics, Stanford University, Stanford, CA (1975).

[9] R. Fletcher, "Practical Methods of Optimization", John Wiley and Sons, New York, (1987).

[10] D. Luenberger, "Optimization by Vector Space Methods", John Wiley, New York, (1967).

[11] J. Eriksson, "A note on solution of large sparse maximum entropy problems with linear equality constraints", Math. Programming, 18, (1980), pp. 146–154.

[12] M. Milman, R. Jelliffe and F. Jiang,"Moment Matching/Maximum Entropy Optimization", University of Southern California School of Medicine, Laboratory of Applied Pharmacokinetics Tech. Report, 1995.

[13] T. Oscarsson, Dual principles in maximum entropy reconstruction of the wave distribution function, J. Comp. Physics, 110, (1994), pp. 221–233.

[14] M. J. D. Powell, An algorithm for maximizing entropy subject to simple bounds, Math. Prog. 42, (1988) pp. 171–180.

[15] R. H. Ruening, A. S. Sams, and R. B. Notari, Role of Pharmacokinetics in Drug Dosage Adjustment. 1. Pharmacological effects, kinetics, and apparent volume of distribution of digoxin, J. Clin. Pharmacol. 13: 127–141, 1973.

[16] R. Jelliffe, A. Schumitzky, M. Van Guilder, and F. Jiang, User Manual, Version 10.7 of the USC*PACK Collection of PC Programs, Laboratory of Applied Pharmacokinetics, USC School of Medicine, Los Angeles, CA, December 1, 1995.

[17] X. Wang, Parallel Implementation for Large Scale Entropy Maximization Problems, Laboratory of Applied Pharmacokinetics Tech. Report, July, 1997.