Hit and Run and Stuff

Brian Cohn May Szedlák

March 10, 2015

Abstract

1 Author Summary

2 Introduction

3 Results

4 Discussion

5 Materials

6 Hit-and-Run

The Hit-and-Run algorithm used for sampling in a convex body K, was introduced by Smith in 1984 [5]. The mixing time is known to be $\mathcal{O}^*(n^2R^2/r^2)$, where R and r are the radii of the inscribed and cicumscribed ball of K respectively [1, 3]. In the case of the muscles of a limb, we are interested in the polygon P that is given by the set of all possible activations $\mathbf{a} \in \mathbb{R}^n$ that satisfy

$$\mathbf{f} = A\mathbf{a}, \mathbf{a} \in [0,1]^n,$$

where $\mathbf{f} \in \mathbb{R}^m$ is a fixed force vector and $A = J^{-T}RF_m \in \mathbb{R}^{m\times n}$. P is bounded by the unit n-cube since all variables a_i, $i \in [n]$ are bounded by 0 and 1 from below, above respectively.

Consider the following 1×3 example.

$$\begin{aligned} 1 &= \frac{10}{3}a_1 - \frac{53}{15}a_2 + 2a_3 \\ &a_1, a_2, a_3 \in [0,1], \end{aligned}$$

the set of feasible activations is given by the shaded set in figure ??.

FIGURE 1

The Hit-and-Run walk on P is defined as follows (it works analogously for any convex body).

1. Find a given starting point $\mathbf{p}$ of P .

2. Generate a random direction through $\mathbf{p}$ (uniformly at random over all directions).

3. Choose the next point of the sampling algorithm uniformly at random from the segment of the line in P.

4. Repeat from (b) the above steps with the new point as the starting point.

FIGURE 2: *Figures in 2D or 3D? Better in 3D to have the overview.*

1. figure inner point

2. *figure choice of direction*

3. *figure line segment*

4. *figure choice of new point*

5. *figure of distribution after some sampling*

The implementation of this algorithm is straight forward except for the choice of the random direction. How do we sample uniformly at random (u.a.r.) from all directions in P? Suppose that $\mathbf{q}$ is a direction in P and $p \in P$. Then by definition of P, $\mathbf{q}$ must satisfy $\mathbf{f} = A(\mathbf{p} + \mathbf{q})$. Since $\mathbf{p} \in P$, we know that $\mathbf{f} = A\mathbf{p}$ and therefore

$$\mathbf{f} = A(\mathbf{p} + \mathbf{q}) = \mathbf{f} + A\mathbf{q}$$

and hence

$$A\mathbf{q} = 0.$$

FIGURE 3: *q must be parallel to the plane given by* $1 = \frac{10}{3}a_1 - \frac{53}{15}a_2 + 2a_3$.

We therefore need to choose directions uniformly at random from all directions in the vectorspace

$$V = \{\mathbf{q} \in \mathbb{R}^n | A\mathbf{q} = 0\}.$$

As shown by Marsaglia this can be done as follows [4].

1. Find an orthonormal basis $b_1, \dots, b_r \in \mathbb{R}^n$ of $A\mathbf{q} = 0$.

2. Choose $(\lambda_1, \dots, \lambda_r) \in \mathcal{N}(0,1)^n$ (from the Gaussian distribution).

3. $\sum_{i=1}^{r} \lambda_i b_i$ is a u.a.r. direction.

A basis of a vectorspace V is a minimal set of vectors that generate V, and it is orthonormal if the vectors are pairwise orthogonal (perpendicular) and have unit length. Using basic linear algebra one can find a basis for $V = \{A\mathbf{q} = 0\}$ and orthogonalize it with the well known Gram-Schmidt method (for details see e.g. [2]). Note that in order to get the desired u.a.r. distribution the basis needs to be orthonormal. For the limb case we can safely assume that the rows of A are linearly independent and hence the number of basis vectors is $n - m$.

FIGURE 4

1. *Some basis*

2. *orthonormal basis*

7 Acknowledgments

Author Summary, Introduction, Results, Discussion, Materials and Methods, Acknowledgments, References, and Supporting Information Captions. Figure Legends and Tables

References

[1] M. Dyer, A. Frieze, and R. Kannan. A random polynomial time algorithm for approximating the volume of convex bodies. *Proc. of the 21st annual ACM Symposium of Theory of Computing*, pages 375–381, 1989.

[2] T. S. Blyth E. F. Robertson. *Basic Linear Algebra.* Springer, 2002.

[3] L. Lovász. Hit-and-run mixes fast. *Math. Prog.*, 86:443–461, 1998.

[4] G. Marsaglia. Choosing a point from the surface of a sphere. *Ann. Math. Statist.*, 43:645–646, 1972.

[5] R.L. Smith. Efficient monte-carlo procedures for generating points uniformly distributed over bounded regions. *Operations Res.*, 32:1296–1308, 1984.