Optimization problem from last time

MAXIMIZE: $n_0 \cdot n_1 \cdot n_2 \cdots n_q$

MINIMIZE: $-n_0 \cdot n_1 \cdot n_2 \cdots n_q$

or $\boxed{-\ln(n_0 \cdot n_1 \cdot n_2 \cdots n_q + 1)}$

recall:

df = diff. in func. values

compute:

$e^{-df/k}$

MAGIC SQUARE:

An $n \times n$ grid filled with the integers $1, 2, 3, \ldots, n^2$ such that all row, column, and diagonal sums are the same.

not a magic square:

	1	7	6	14
	8	5	3	16
	4	2	9	15
15	13	14	18	16

sum of k consecutive integers:

$$\begin{array}{rl} & 1 + 2 + 3 + \cdots + (k-1) + k = \tfrac{1}{2}k(k+1) \\ + & k + (k-1) + (k-2) + \cdots + 2 + 1 \\ \hline & (k+1) + (k+1) + (k+1) + \cdots + (k+1) + (k+1) \leftarrow k \text{ pairs} \end{array}$$

So the sum is $k(k+1)$

let $k = n^2$:

$$1+2+3+\cdots+n^2 = \underline{\tfrac{1}{2}n^2(n^2+1)}$$

sum of all n^2 numbers in the magic square

If we have a magic square, then each row, column, and diagonal must add up to:

$$\frac{1}{n}\left(\frac{1}{2}n^2(n^2+1)\right) = \frac{1}{2}n(n^2+1)$$

If $n=3$: sums are $\frac{1}{2}(3)(3^2+1) = \frac{1}{2}(3)(10) = 15$.

Using simulated annealing to find magic squares:

1. What are the states?
2. How do we move between states?
3. What function do we minimize?