Notes on Nelder-Mead algorithm for maximizing a log likelihood \( L(\theta) \), with parameter \( \theta \in \mathbb{R}^p \).

The algorithm produces a sequence of sets \( S_1, S_2, \ldots, S_m, \ldots \) with any \( S = \{ \theta_0, \ldots, \theta_{p+1} \} \), \( p+1 \) points in \( \mathbb{R}^p \).

Assume that we start with \( S \) not a hyperplane. Update until stopping condition:
1. \( \theta_j \)'s sufficiently close within \( S \)
2. \( L(\theta_j) \)'s sufficiently close
3. Reach maximum number of iterations

Updates proceed by trying to improve the \( L(\theta) \) value of points in current \( S_m = \emptyset \).

Reflect step: First identify three points in \( S \).

\[ \theta_0 = \text{value with smallest } L(\theta) \]
\[ \theta_b = \text{best (largest) } L(\theta) \]
\[ \theta_{b+1} = \text{2nd smallest } L(\theta) \]

and form
\[ \theta_o = \frac{1}{p} \sum_{j \neq b} \theta_j \quad \text{[mean of non-out]} \]
Construct \( q_r = (1 + \alpha) q_o - \alpha q_o \)

for some \( \alpha > 0 \)

\[ q_0 = \frac{1}{p} \sum_{j \neq 0} q_j \]

Expansion Step:

If \( l(q_r) > l(q_o) \)

[reflection improves things]

expand further to

\[ q_{o^e} = q_o + \gamma (q_r - q_o) \]

for some \( \gamma > 1 \)

if \( l(q_{o^e}) > l(q_r) \)

Return \( S_{m+1} = \{ S_m \mid q_{o^e} \} + f[q_{o^e}] \) \[ i.e. \text{drop } q_o \text{ and include } q_e \]

if \( l(q_{o^e}) \leq l(q_r) \)

Return \( S_{m+1} = \{ S_m \mid q_{o^e} \} + f[q_{o^e}] \)

i.e., if reflection improves things, try further improvement and take the better \( q_r, q_{o^e} \).
Intermediate Case


\[ l(Q_{01}) \leq l(o_c) < l(Q_3) \]

then set

\[ S_{m1} = (S_m \setminus Q_2) + \{o_c\} \]

[reflection not too bad]

Contraction Case

1. External \[ l(Q_4) \leq l(o_c) < l(Q_{0m}) \]

form

\[ o_c = Q_0 + \beta (Q_0 - Q_0) \]

for some \( \beta \in (0,1) \)

if \( l(o_c) > l(Q_0) \)

\[ S_{m1} = (S_m \setminus Q_2) + \{o_c\} \]

else "shrink" [see below]

2. Internal \( l(o_c) < l(Q_0) \) [reflected point very close]

form

\[ o_c = Q_0 + \beta (Q_0 - Q_0) \]
If \( l(\theta_{cc}) > l(\theta_d) \)

\[ S_{m+1} = (S_m \theta_d)^2 + l(\theta_{cc}) \]

else "shrink"

Shrink step: if no luck above, and directed here,

\[ S_{m+1} = \{ \theta_b, \forall j \neq b, \theta_b + s(\theta_j - \theta_b) \} \]

for some \( s \in (0, 1) \)

Note: in constrained parameter spaces, some adjustment is required to keep all points \( \theta_1, \theta_2, \theta_0, \theta_r, \theta_{cc} \) in the parameter space.