Lecture 11: Nelder-Mead method

Ciaran Evans

Recap: optimization

Definition: *Optimization* is the problem of finding values that minimize or maximize some function.

Example:

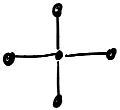
$$RSS(\beta_0, \beta_1) = \sum_{i=1}^{n} (\text{Weight}_i - \beta_0 - \beta_1 \text{WingLength}_i)^2$$

- $RSS(\beta_0, \beta_1)$ is a function of β_0 and β_1
- We want to find the values of β_0 and β_1 that *minimize* this function

Last time: Compass search overview (in 2 dimensions)

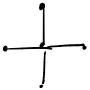
To minimize some function $f(\beta_0, \beta_1)$:

1. Choose an initial guess $(\beta_0^{(0)}, \beta_1^{(0)})$ and initial step size Δ_0
2. Evaluate f at the points
   - $(\beta_0^{(0)}, \beta_1^{(0)})$
   - $(\beta_0^{(0)}, \beta_1^{(0)} \pm \Delta_0)$
   - $(\beta_0^{(0)} \pm \Delta_0, \beta_1^{(0)})$



3. If f is smaller at one of the new points: move to the smallest value, update to $(\beta_0^{(1)}, \beta_1^{(1)})$
4. Otherwise: $\Delta_{k+1} = 0.5\Delta_k$ (shrink step size and try again)
5. Repeat

Downsides of compass search



Downsides of the compass search algorithm given on the previous slide:

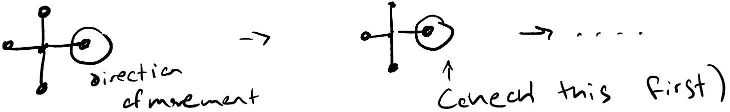
- Can require many steps
- Only considers specific search directions
- Step size only shrinks; if we find a promising direction, can't take bigger steps
- Choosing a direction requires many evaluations of f
  - In 2 dimensions (e.g. β_0 and β_1), requires 4 evaluations of f
  - In d dimensions ($\beta_0, \beta_1, ..., \beta_{d-1}$), requires $2d$ evaluations of f

Question: How could you modify the algorithm to address some of the issues here?

Compass search modifications

- Try the most recent direction of movement first (*dynamic ordering*)

- Try directions one at a time, and move as soon as an improvement on current position is found – don't have to try *all* directions and find the *best* direction (*opportunistic*)



Alternative: Nelder-Mead method

In R, standard function for performing optimization is `optim`:

```
?optim
```

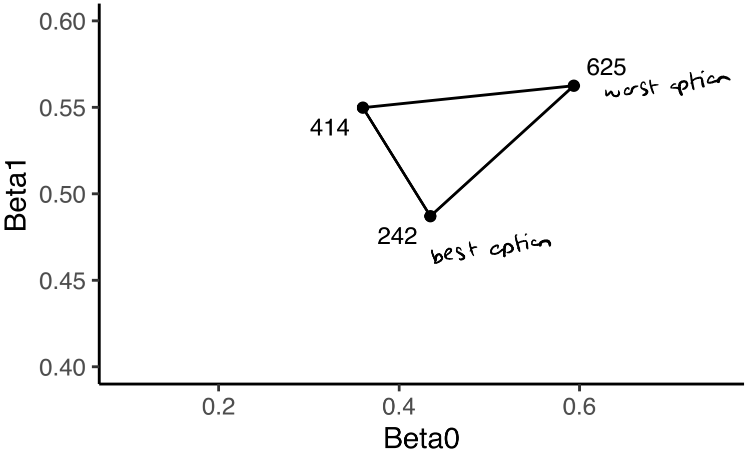
```
optim(par, fn, gr = NULL, ...,
      method = c("Nelder-Mead", "BFGS", "CG", "L-BFGS-B", '
                 "Brent"),
      lower = -Inf, upper = Inf,
      control = list(), hessian = FALSE)
```

- **Nelder-Mead:** another derivative-free optimization method
- **Very** widely used
  - Original 1965 paper has 40,000+ citations
  - Can find many examples of use in biology, medicine, physics, engingeering, etc.

Nelder-Mead method (in 2 dimensions)

want to minimize

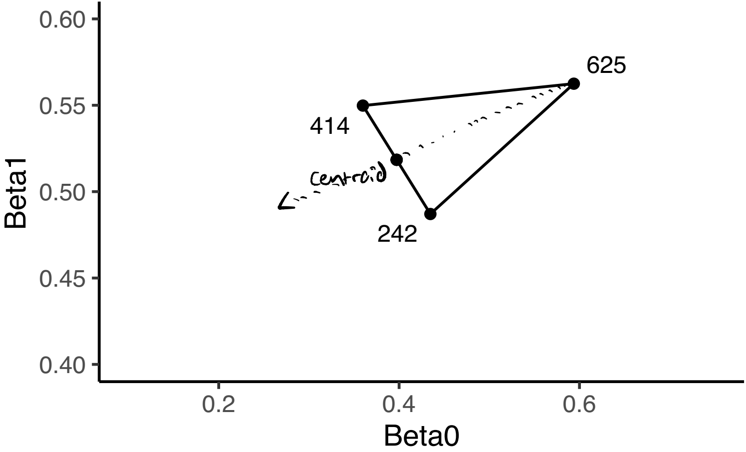
Start with 3 initial points, evaluate function f at each point:



Question: Where should I search next?

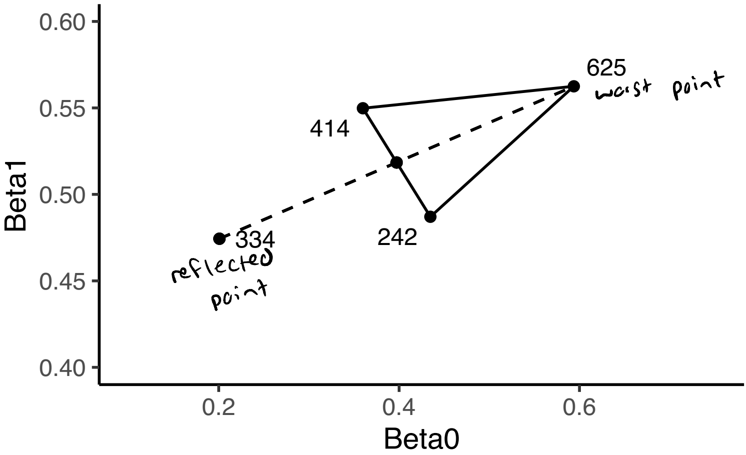
Nelder-Mead method (in 2 dimensions)

First, calculate centroid of the vertices (except the worst one)



Reflection

Now **reflect** the worst vertex over the centroid:



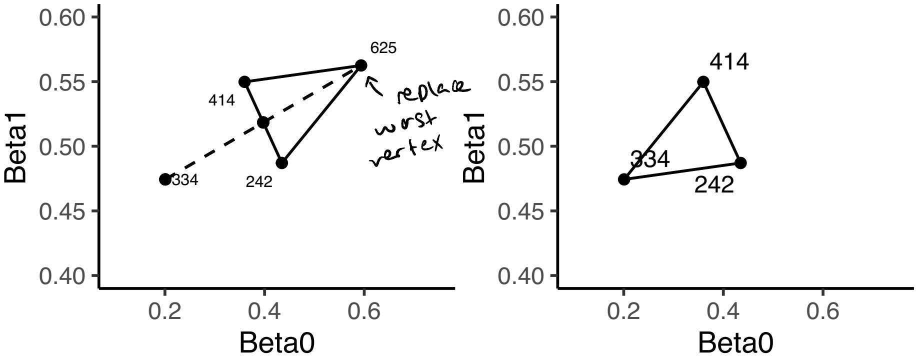
Question: Is the reflected point an improvement?

Reflection

242 $\leq$ 334 $\leq$ 414

best point   reflected point   second-worst point

If the reflected point is *better than the second-worst point* but *worse than the best point*, update the triangle:



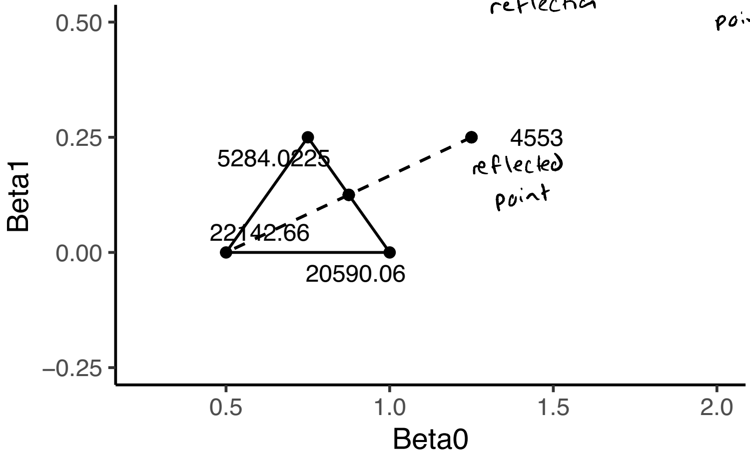
Another scenario

Sometimes the reflected point is better than *all* of the current vertices:

4553 $\leq$ 5284

reflection

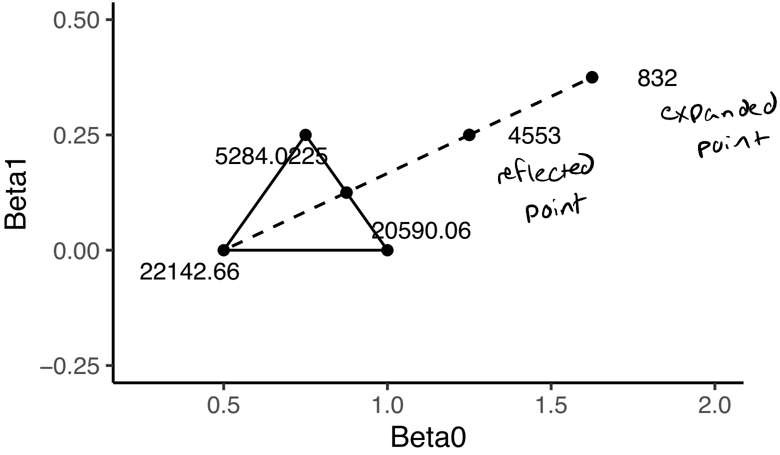
best point



Question: Where do you think we should move here?

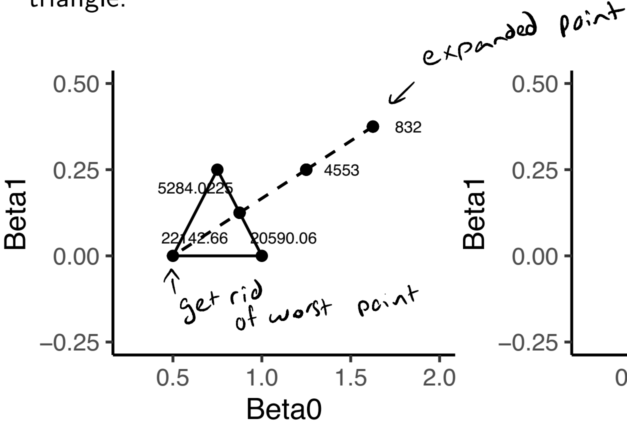
Expansion

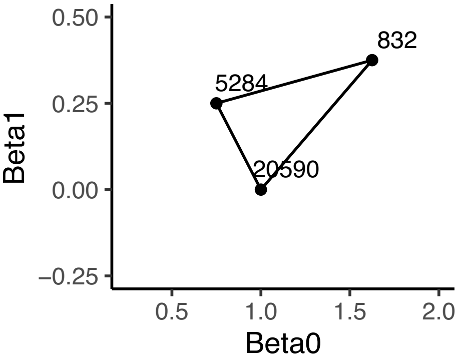
When the reflected point is better than all the current vertices, try **expanding** in that direction:



Expansion

If the expanded point is *better than the reflected point*, update the triangle:



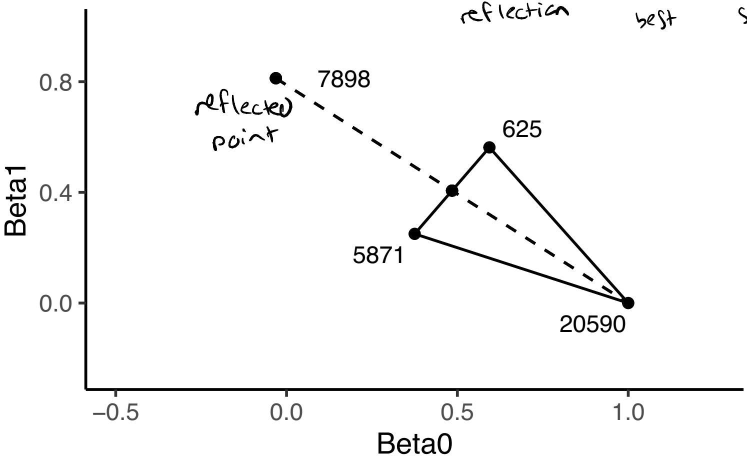


Another scenario

Sometimes, the reflected point is *worse* than two of the existing vertices, but *better* than the worse vertex:

7898 $\leq$ 625 $\leq$ 5871

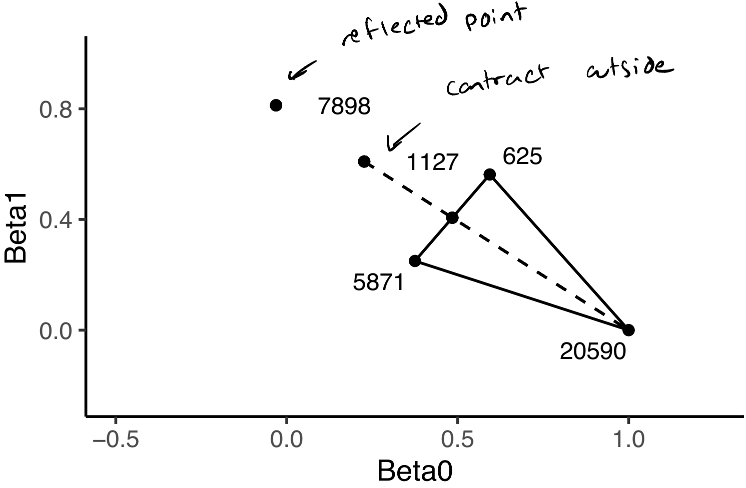
reflection best second-worst



Question: What should we do here?

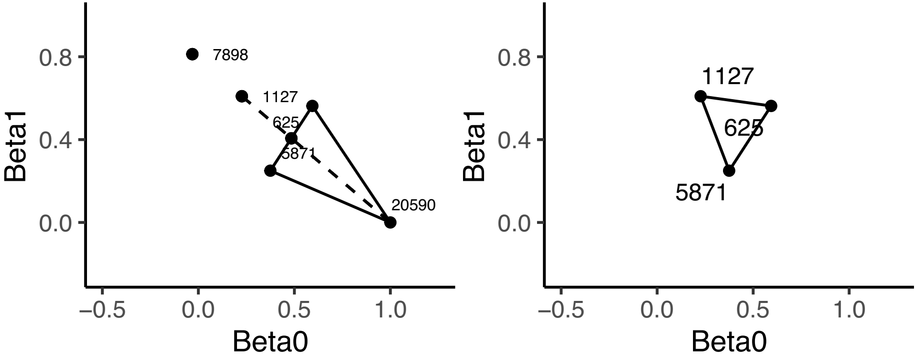
Contract outside

Move in the direction of the reflected point, but *not as far*:



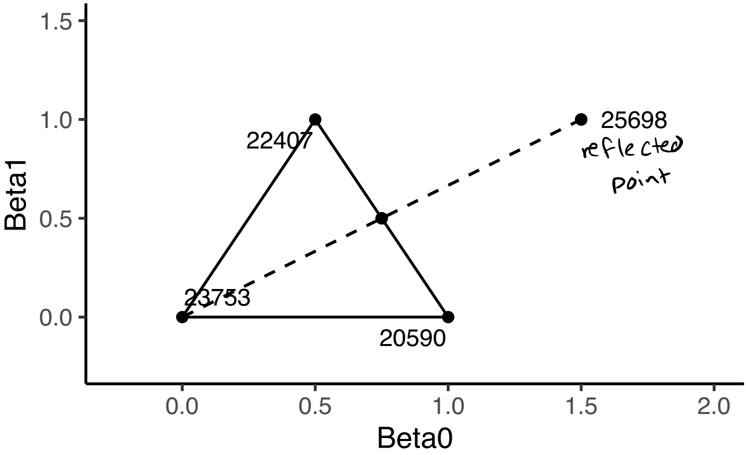
Contract outside

Move in the direction of the reflected point, but *not as far*:



Another scenario

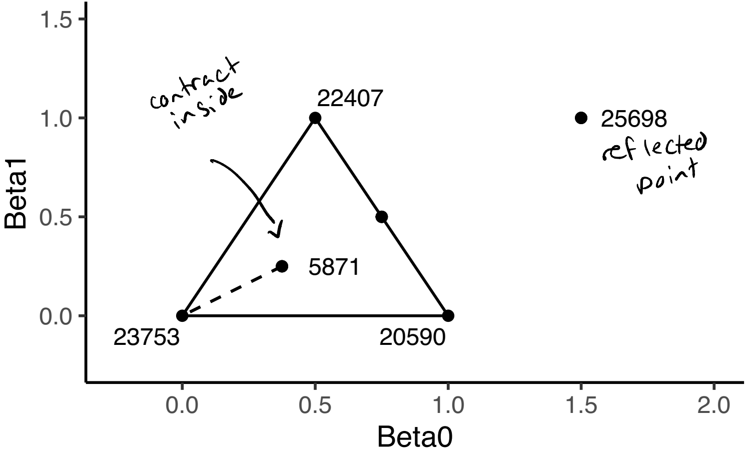
And sometimes, the reflected point is worse than *any* of the current vertices:



Question: What do you think we should do here?

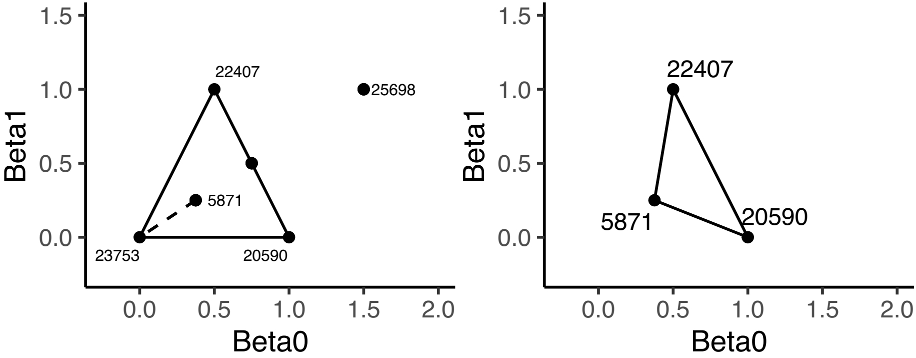
Contract inside

Move *away* from the worst point, but stay *inside* the triangle:



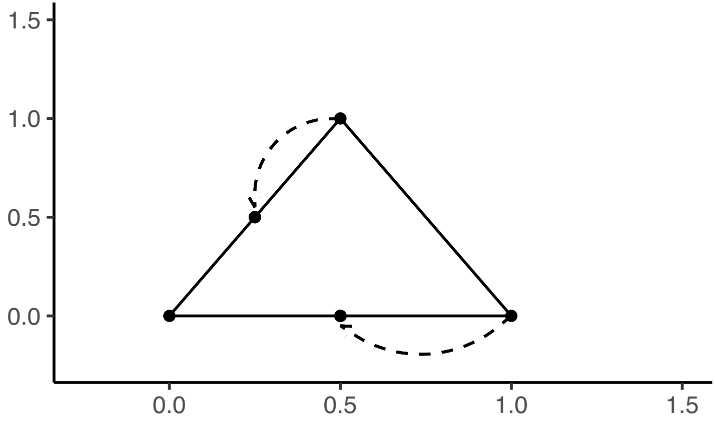
Contract inside

Move *away* from the worst point, but stay *inside* the triangle:



Shrinking

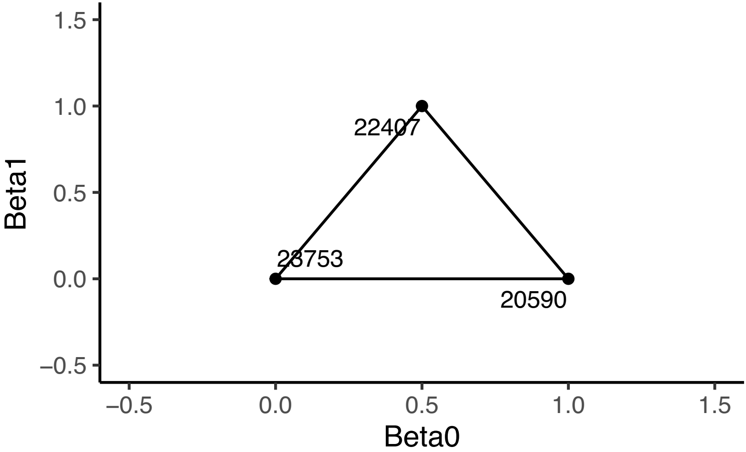
Sometimes, none of the other transformations improve the triangle. In that case, we *shrink* towards the best current vertex



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

Initial points:

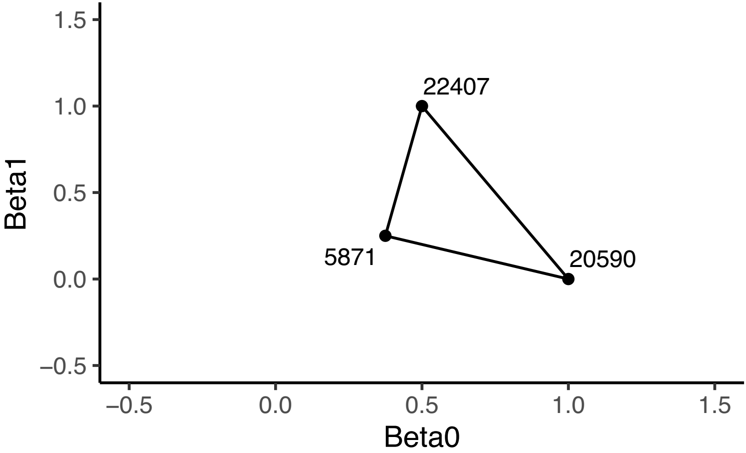


Nelder-Mead approach: search for a minimum through transformations of the triangle

Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

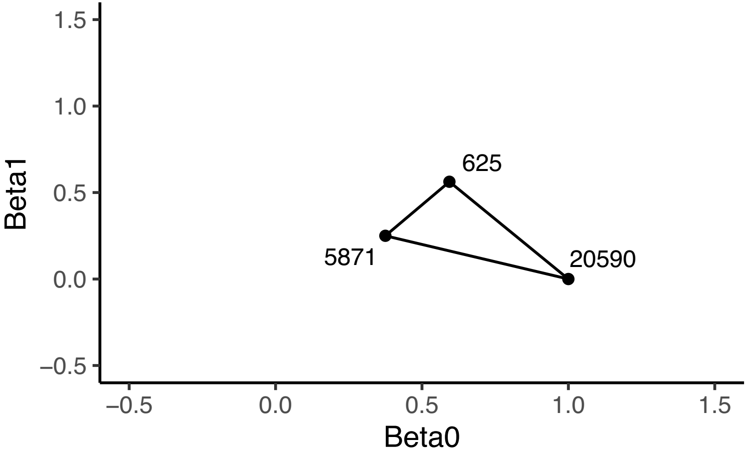
```
## [1] "contract inside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

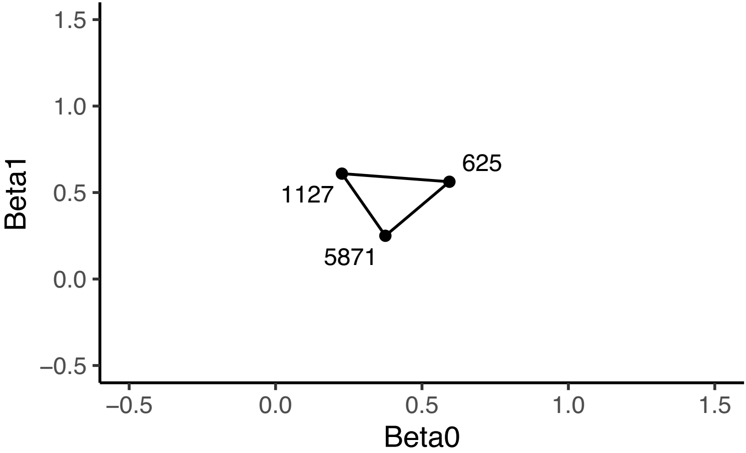
```
## [1] "contract inside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

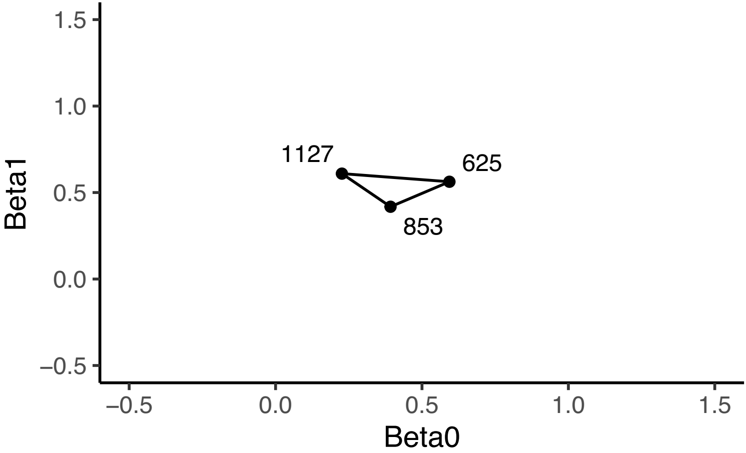
```
## [1] "contract outside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

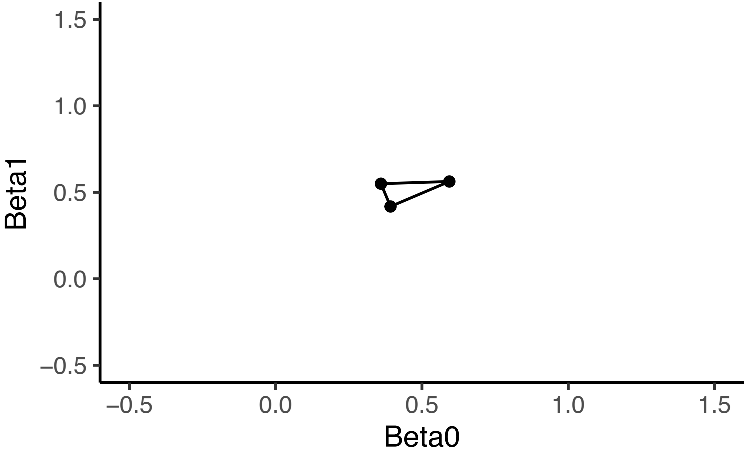
```
## [1] "contract inside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

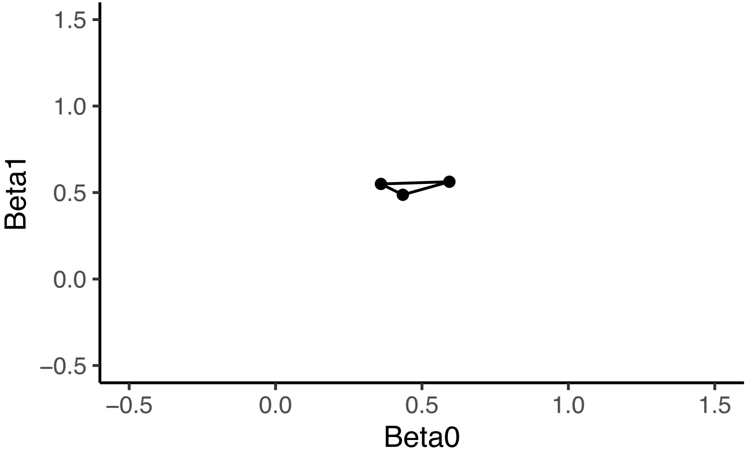
```
## [1] "contract inside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

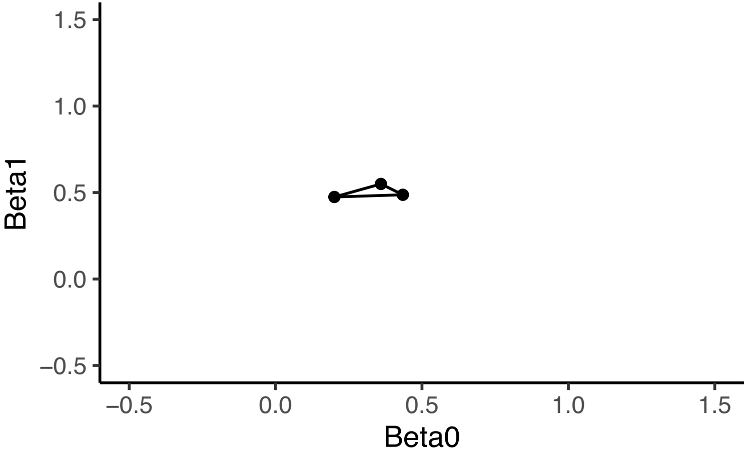
```
## [1] "contract inside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

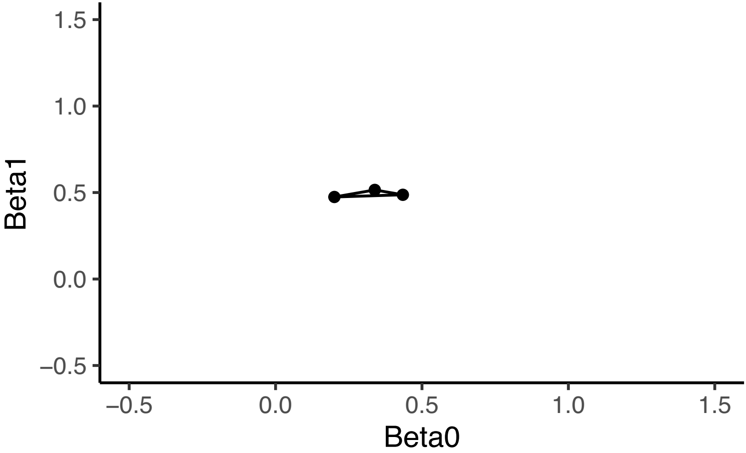
```
## [1] "reflection"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

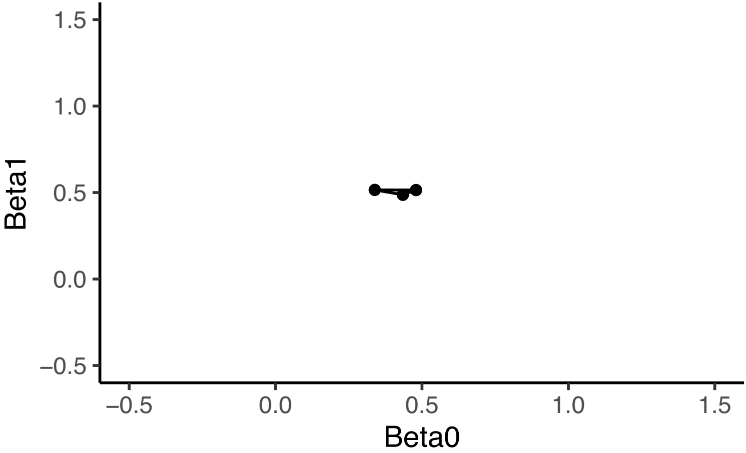
```
## [1] "contract inside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

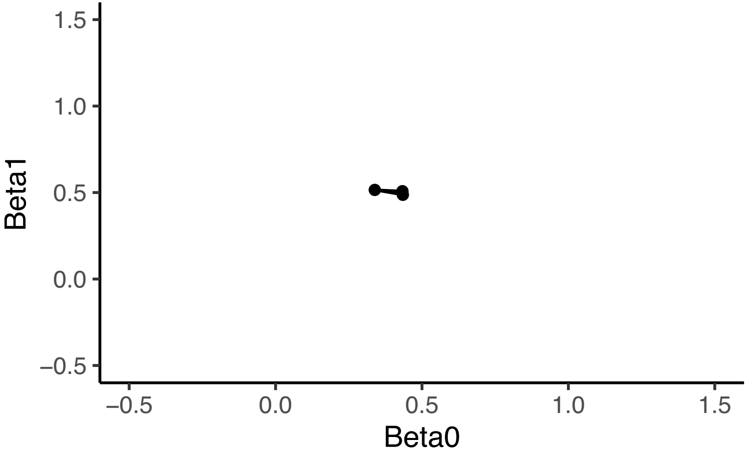
```
## [1] "contract outside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

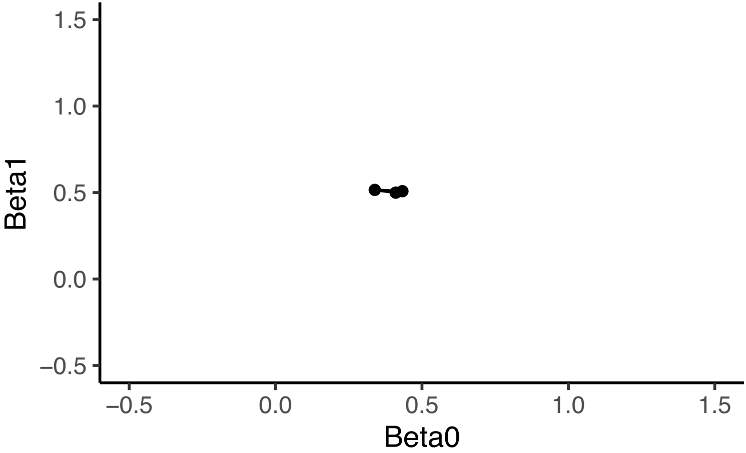
```
## [1] "contract inside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

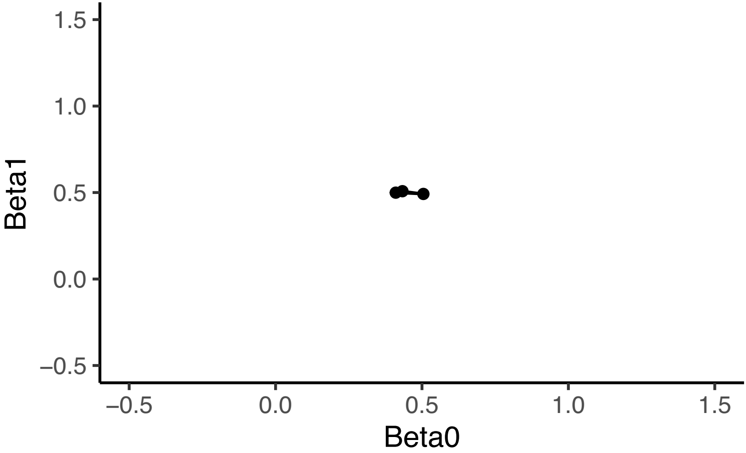
```
## [1] "contract inside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

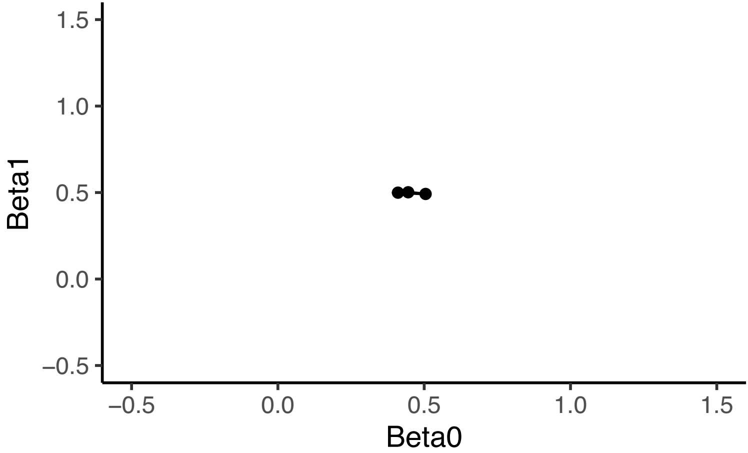
```
## [1] "reflection"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

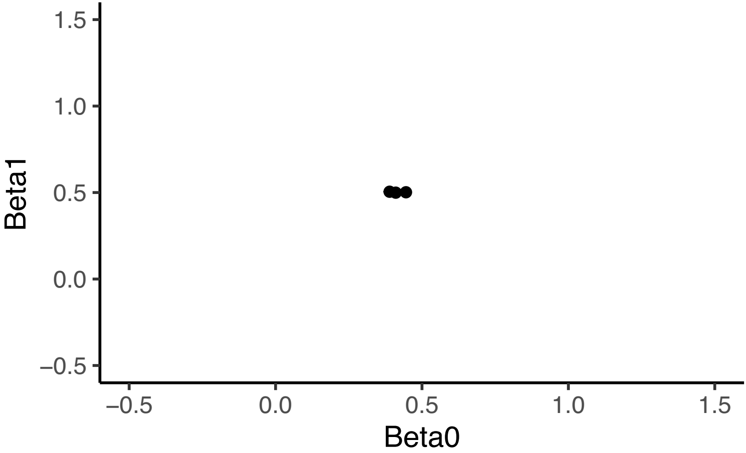
```
## [1] "contract inside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

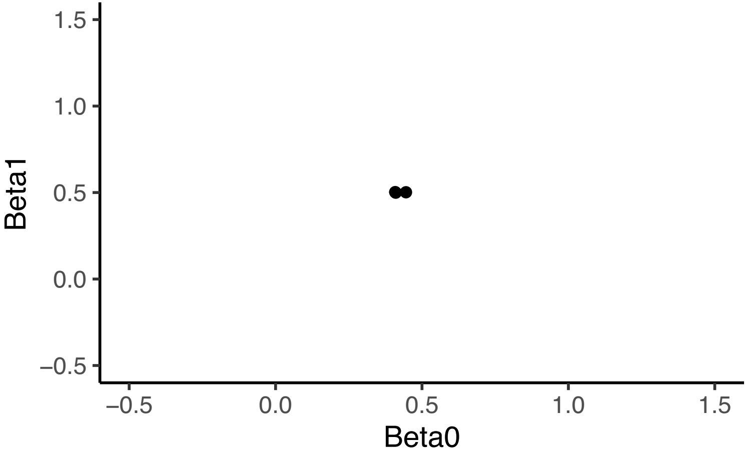
```
## [1] "contract outside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

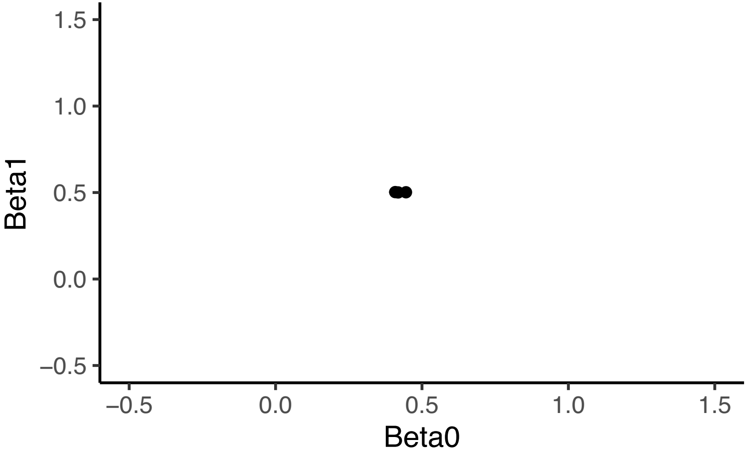
```
## [1] "contract inside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

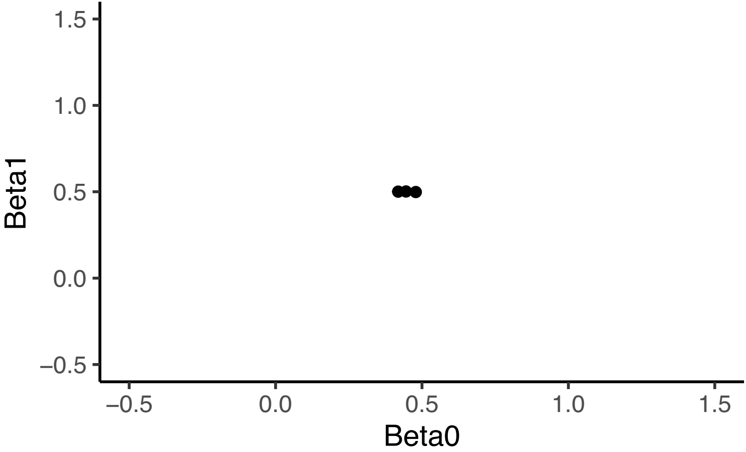
```
## [1] "contract inside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

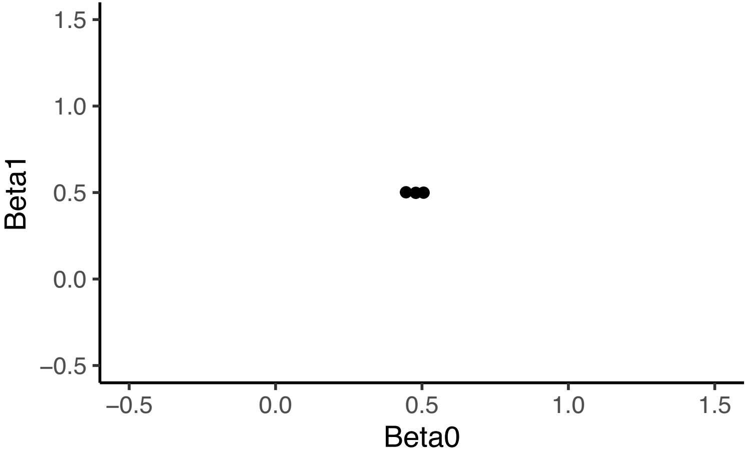
```
## [1] "expansion"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

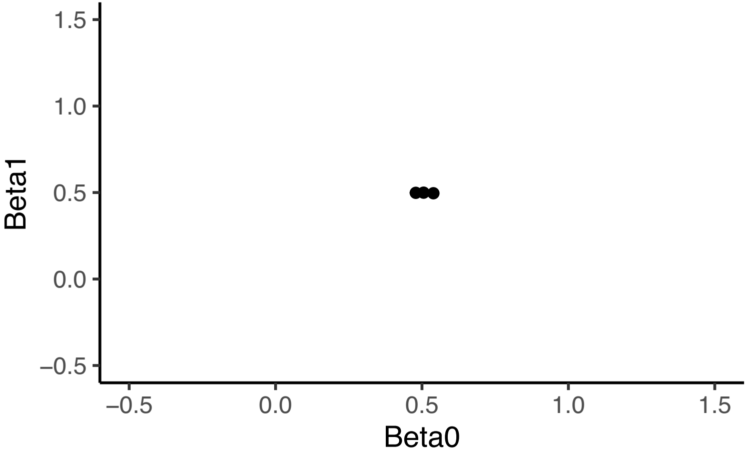
```
## [1] "reflection"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

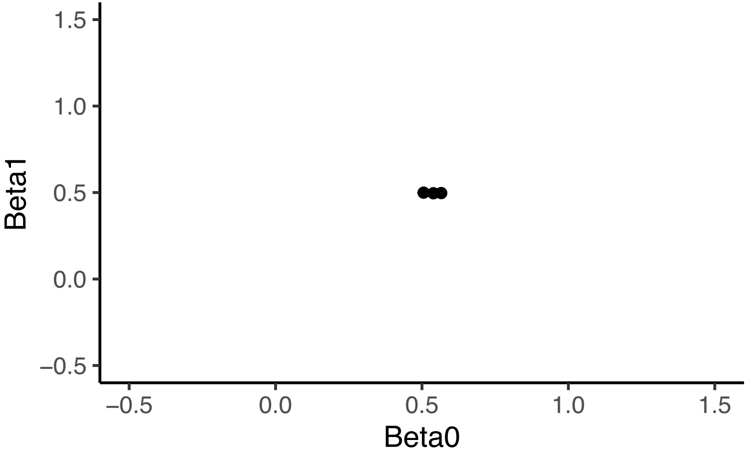
```
## [1] "reflection"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

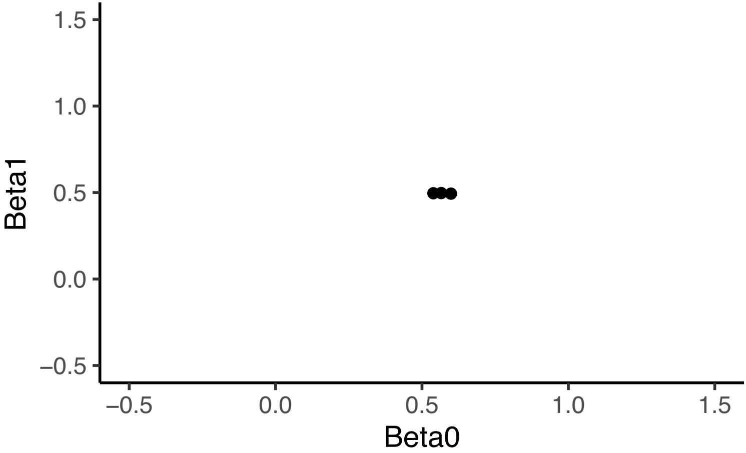
```
## [1] "reflection"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

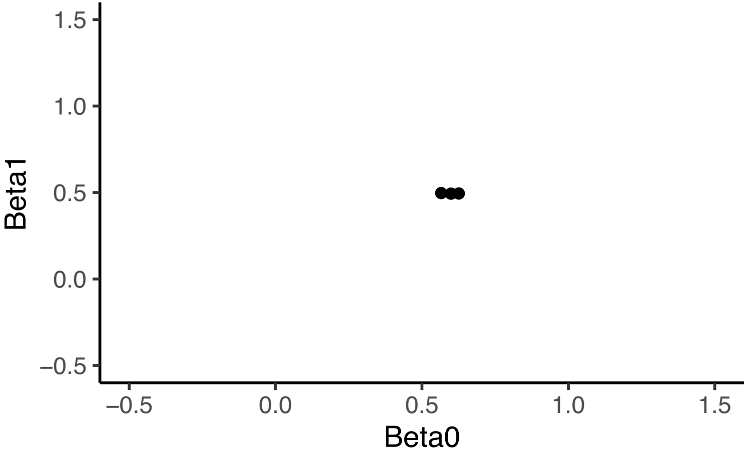
```
## [1] "reflection"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

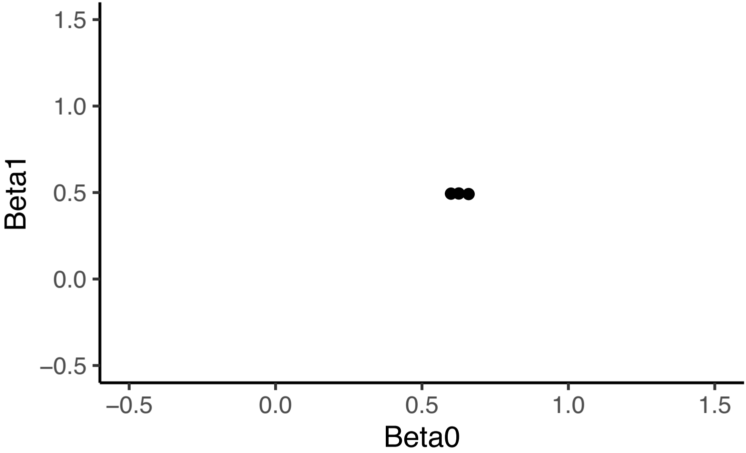
```
## [1] "reflection"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

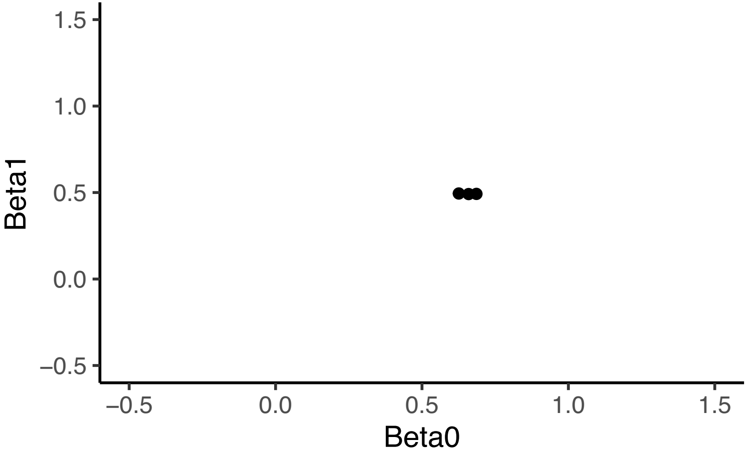
```
## [1] "reflection"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

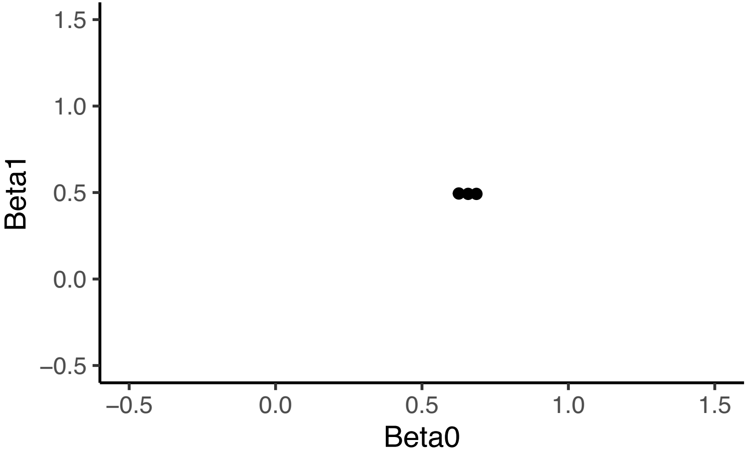
```
## [1] "reflection"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

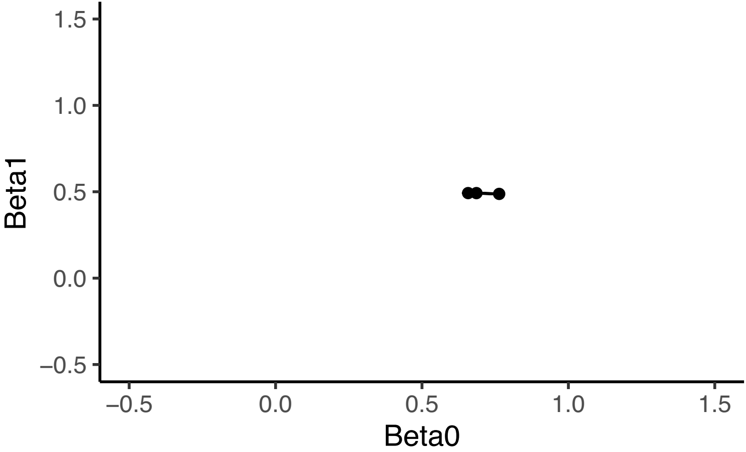
```
## [1] "contract inside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

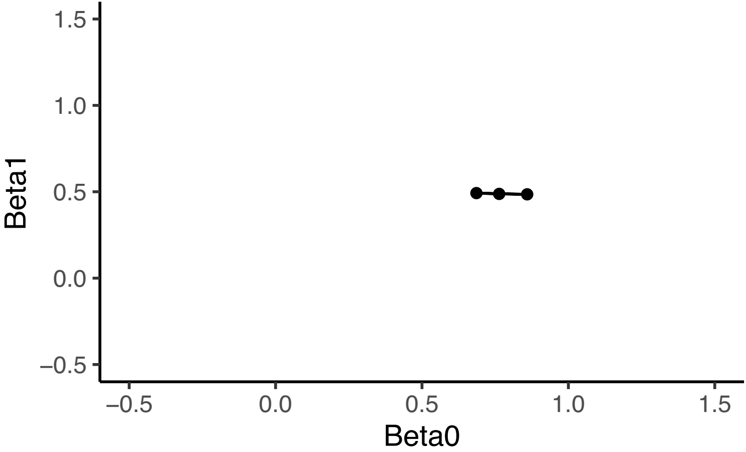
```
## [1] "expansion"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

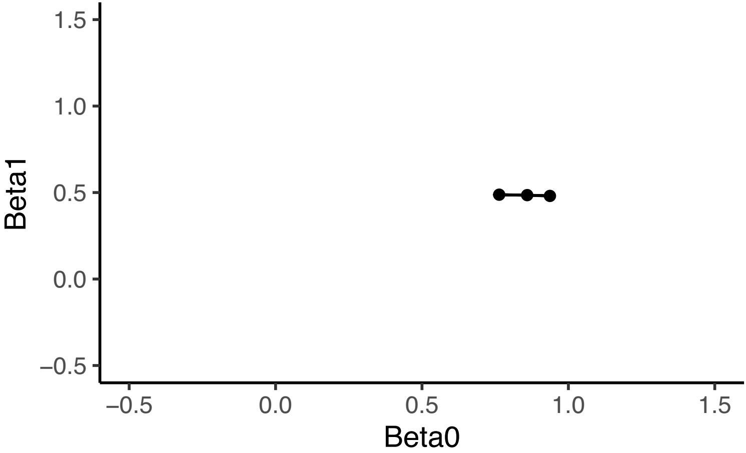
```
## [1] "expansion"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

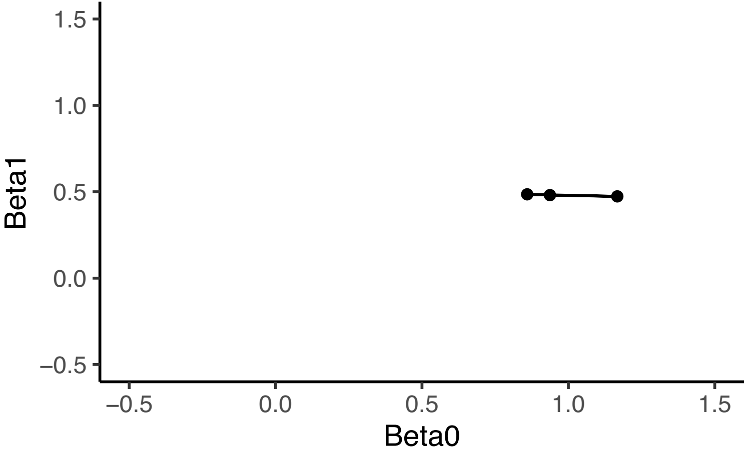
```
## [1] "reflection"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

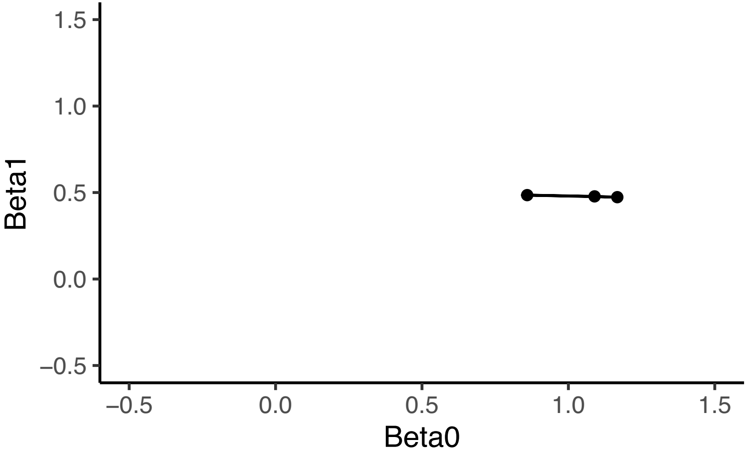
```
## [1] "expansion"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

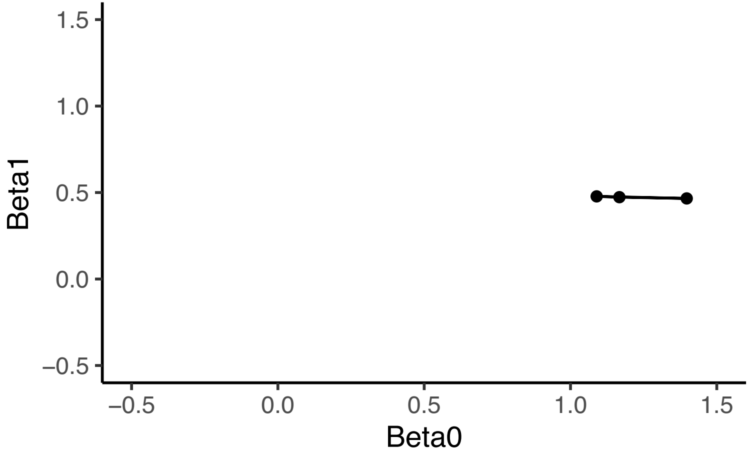
```
## [1] "reflection"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

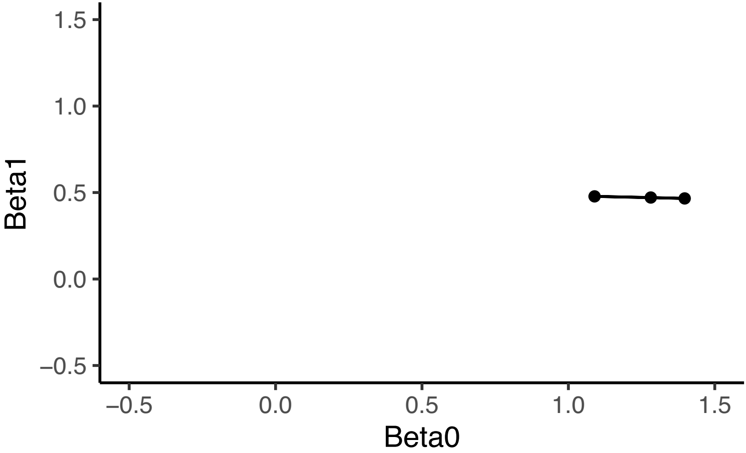
```
## [1] "reflection"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

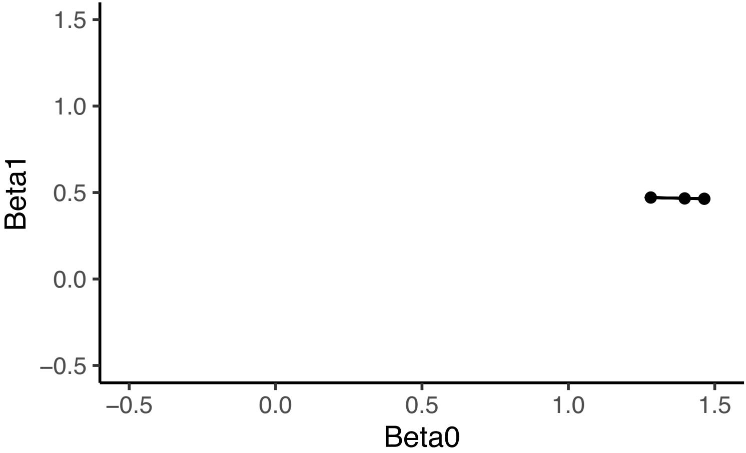
```
## [1] "contract outside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

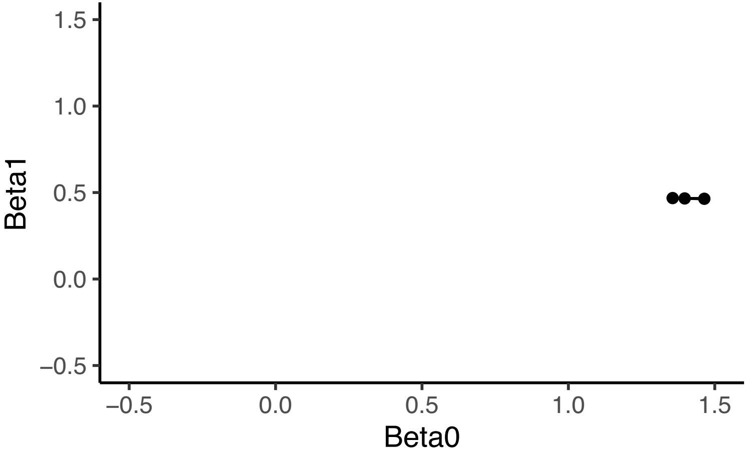
```
## [1] "contract outside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

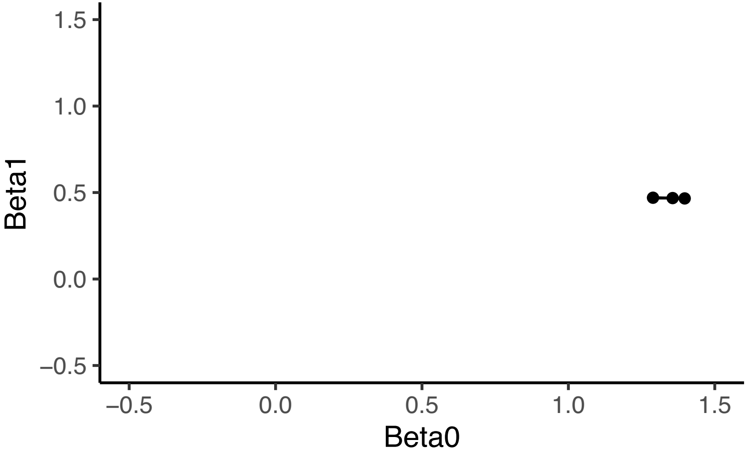
```
## [1] "contract inside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

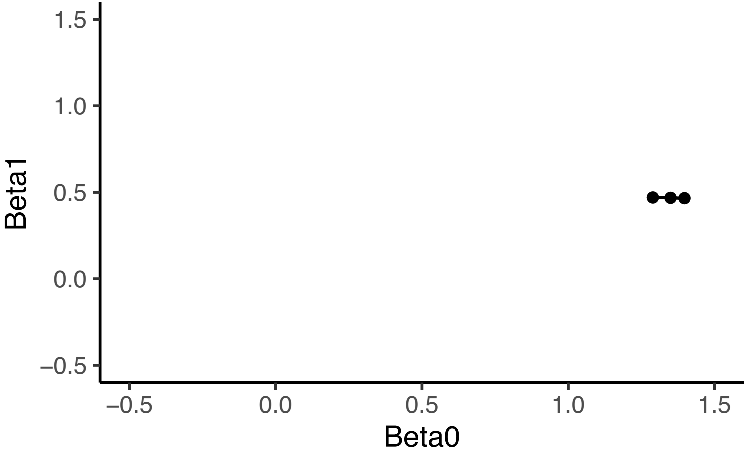
```
## [1] "reflection"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

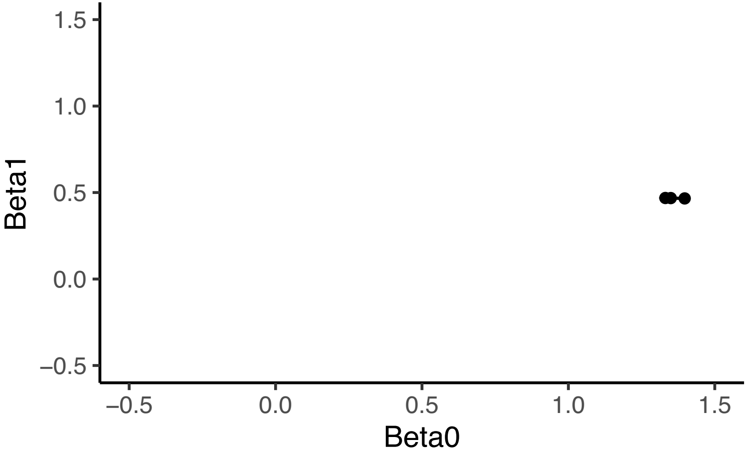
```
## [1] "contract inside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

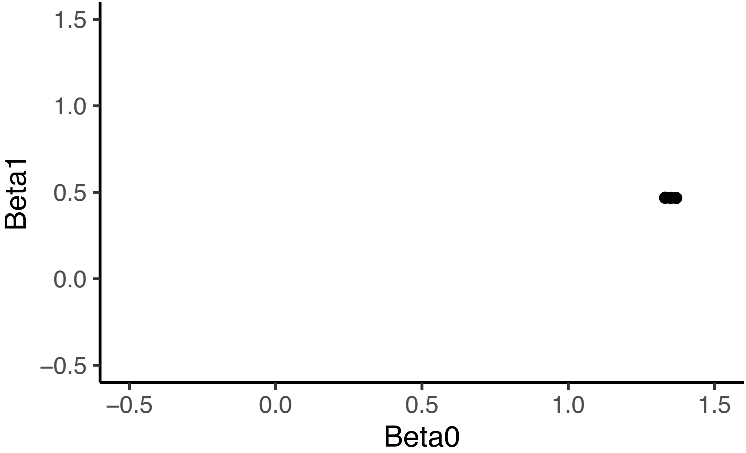
```
## [1] "contract inside"
```



Nelder-Mead in action

$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$

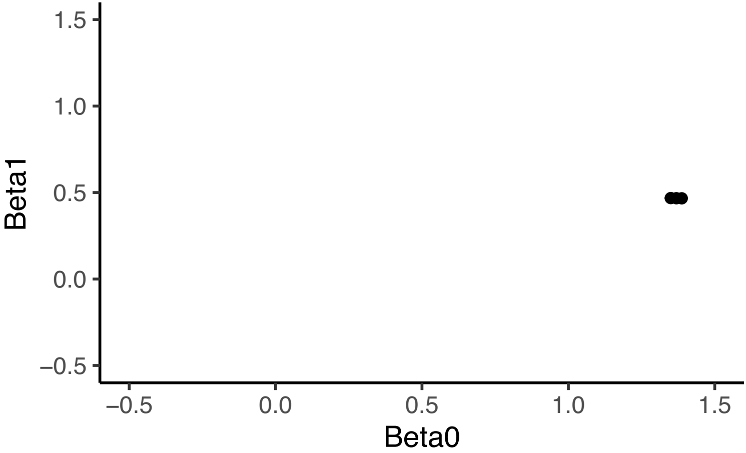
```
## [1] "contract inside"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

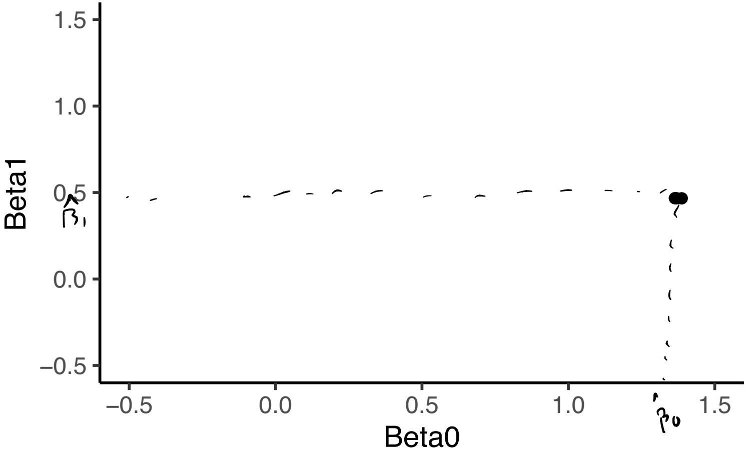
```
## [1] "reflection"
```



Nelder-Mead in action

$$\widehat{\text{Weight}}_i = 1.3655 + 0.4674\text{WingLength}_i$$

```
## [1] "contract inside"
```



Nelder-Mead summary

- Explores *away* from the current worst point
  - *Reflection* tries a point in the new direction
- *Expansion* allows us to increase the “step size” if the new direction is particularly promising
- *Contraction* and *shrinking* allows us to decrease the “step size” if we are moving too far

Your turn

Try out the Nelder-Mead transformations for minimizing a function on an interactive example (link also on course website):

https://alexdowad.github.io/visualizing-nelder-mead/