\[ f = x^3 - y^2 - 3x + y + 2 = 0 \]
\[ S = x^2 + y^2 - 4 = 0 \]

Higher dimensions \( \Rightarrow \) matrix approach

Given \( N \) functional relations to be zeroed, involving variables \( X_i, i = 1, 2, \ldots, N \)

\[ f_i(x_1, x_2, \ldots, x_N) = 0 \quad i = 1, 2, \ldots, N \]

Let \( \mathbf{X} \) denote entire vector
\textbf{Implicit Plot \{\{f = 0, g = 0\}, \\
\{x, -2, 2\}, \{y, -2, 2\}\}}
of values, \( X_i \), and expanding \( f_i \) each in Taylor series (as before)

\[
f_i(X + \delta X) = f_i(X) + \sum_{j=1}^{N} \frac{\partial f_i}{\partial x_j} \delta x_j + \mathcal{O}(\delta X^2)
\]

Neglecting term \( \mathcal{O}(\delta X^2) \)

\[
\Rightarrow \text{linear system of equations}
\]

\[
\sum_{j=1}^{N} \frac{\partial f_i}{\partial x_j} \delta x_j = -f_i
\]
Finding multi-dimensional roots is more difficult than multi-dimensional minimization.
(Because minimization has additional integrability conditions, additional constraints yield a one-dimensional downhillness.)

$$x_i \prod_{j=1}^n x_j = \beta_i,$$
Format problem, revisited.

Given: \( X \) values, \( y_0, y_3 = 3 \)

Unknowns: \( \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \)

\( x_1 = 1, \ x_2 = 2, \ x_3 = 3 \)

\( x_0 = 0, \ y_0 = 0 \)
\[
\text{time} = \left[ (y_1 - y_0)^2 + (x_1 - x_0)^2 \right]^{\frac{1}{2}} + \frac{n}{c} \left[ (y_2 - y_1)^2 + (x_2 - x_1)^2 \right]^{\frac{1}{2}} + \frac{1}{c} \left[ (y_3 - y_2)^2 + (x_3 - x_2)^2 \right]^{\frac{1}{2}}
\]

Take \( \delta t = 0 \)

Answer: \( y_1 = 1.201632 \)
\( y_2 = 1.798367 \)

Also, satisfies Snell's law

Using Mathematica FindMinimum
Must choose between techniques that need evaluations of derivatives and methods that require derivatives are more powerful but not always better than just evaluating functions.
- Methods generally require either \(N^2\) or \(N\) storage (not so much of a problem today).

- Method of Nelder ; Mead: slow, but steadily crawls downhill; easy to program (Numer. Recipes)

- within gradient techniques
  1) conjugate gradient (Fletcher- Reeves)
  \(N\) - storage
2) Newton-like methods

Storage

Method of steepest descent

Start at a point $P_0$. As many times needed, move from $P_i$ to $P_{i+1}$ by minimizing along the line from $P_i$ in the direction of local downhill gradient, $-\nabla f(P_i)$.

"Reasonable" approach, but this can many steps to get the bottom of a long valley.
Combinatorial Minimization or Simulated Annealing.

“How do you find the minimum of a function that depends on many parameters”

Traveling salesman problem.

Given $N$ cities to visit, start at any city, but end at that city, visiting each city exactly once, minimize distance traveled.