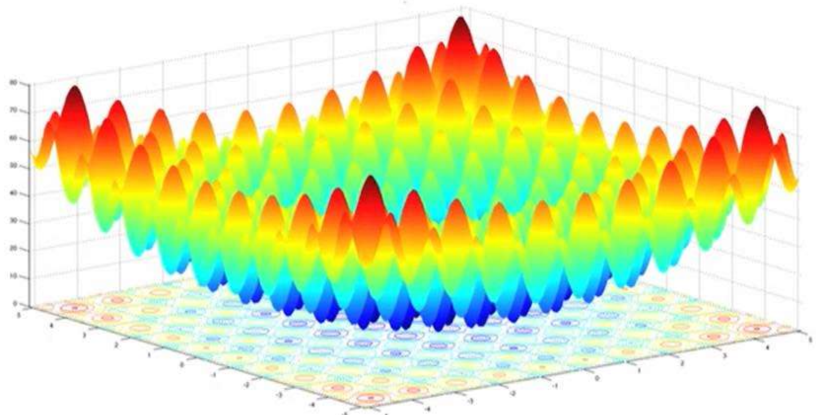
Multivariate optimization – Local and global optimum

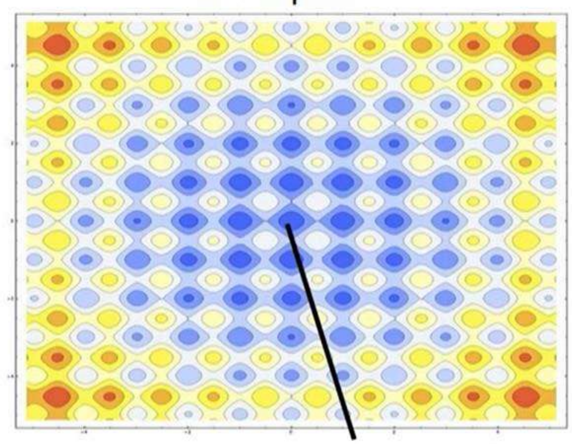
Multivariate optimization

Rastrigin function

$$f(x_1, x_2) = 20 + \sum_{i=1}^{2}[x_i^2 - 10\cos(2\pi x_i)]$$



Contour plot



Global minimum at [0,0]

Multivariate optimization

$$z = f(x_1, x_2 .... x_n)$$

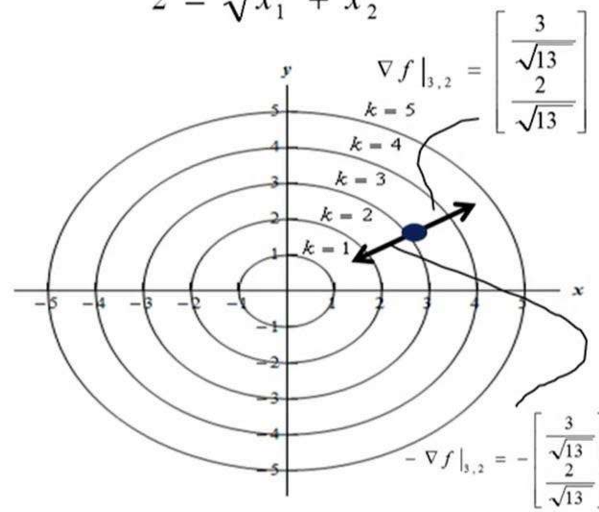
$$z = \sqrt{x_1^2 + x_2^2}$$

Gradient

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \cdots \\ \cdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$$

Hessian

$$\nabla^2 f = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$



- Gradient of a function at a point is orthogonal to the contours
- Gradient points in the direction of greatest increase of the function
- Negative gradient points in the direction of the greatest decrease of the function
- Hessian is a symmetric matrix

Overall Summary – Univariate and multivariate local optimum conditions

Multivariate optimization

$$\boxed{\begin{array}{c}\min\limits_{x} \; f(x) \\ x \in R\end{array}}$$

$$\boxed{\begin{array}{c}\min\limits_{\underline{x}} \; f(\overline{x}) \\ \overline{x} \in R^{n}\end{array}}$$

Necessary condition for x^* to be the minimizer

$$f'(x^*) = 0$$

Sufficient condition

$$f''(x^*) > 0$$

Necessary condition for $\overline{x^*}$ to be the minimizer

$$\nabla f(\bar{x}^*) = 0$$

Sufficient condition

$\nabla^2 f(\overline{x^*})$ has to be positive definite

Multivariate optimization – Numerical example

Multivariate optimization

$$\min_{x_1, x_2} \quad x_1 + 2x_2 + 4x_1^2 - x_1 x_2 + 2x_2^2$$

First order condition

$$\nabla f = \begin{bmatrix} \dfrac{\partial f}{\partial x_1} \\ \dfrac{\partial f}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 1 + 8x_1 - x_2 \\ 2 - x_1 + 4x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\Big\downarrow \text{solving}$$

$$\begin{bmatrix} x_1^* \\ x_2^* \end{bmatrix} = \begin{bmatrix} -0.19 \\ -0.54 \end{bmatrix}$$

Second order condition

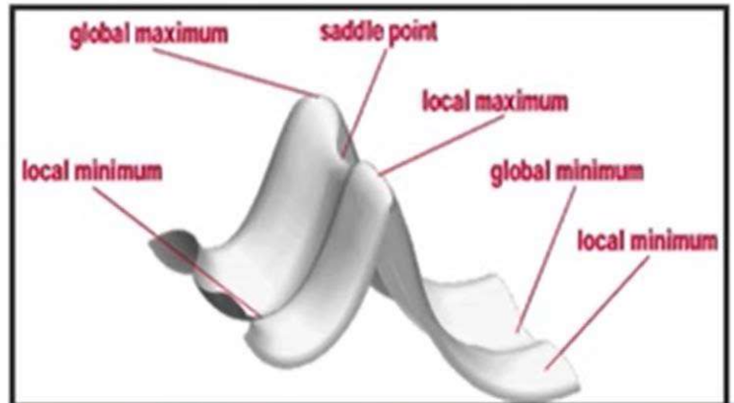
$$\nabla^2 f = \begin{bmatrix} \dfrac{\partial^2 f}{\partial x_1^2} & \dfrac{\partial^2 f}{\partial x_1 x_2} \\ \dfrac{\partial^2 f}{\partial x_2 x_1} & \dfrac{\partial^2 f}{\partial x_2^2} \end{bmatrix} = \begin{bmatrix} 8 & -1 \\ -1 & 4 \end{bmatrix}$$

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 3.76 \\ 8.23 \end{bmatrix}$$

$$\begin{bmatrix} x_1^* \\ x_2^* \end{bmatrix} = \begin{bmatrix} -0.19 \\ -0.54 \end{bmatrix}$$

Unconstrained multivariate optimization - Directional search

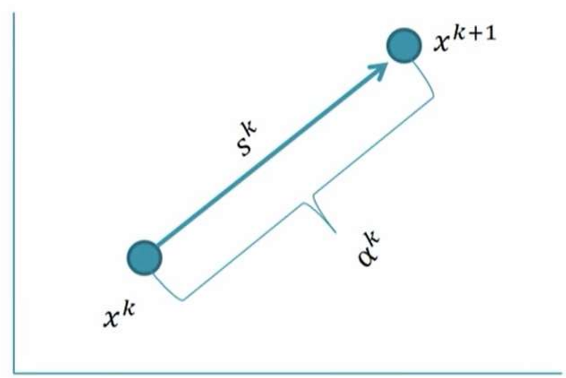
- Aim is to reach the bottom most region
- Directions of descent
- Steepest descent
- Sometimes we might even want to climb the mountain for better prospects to get down further

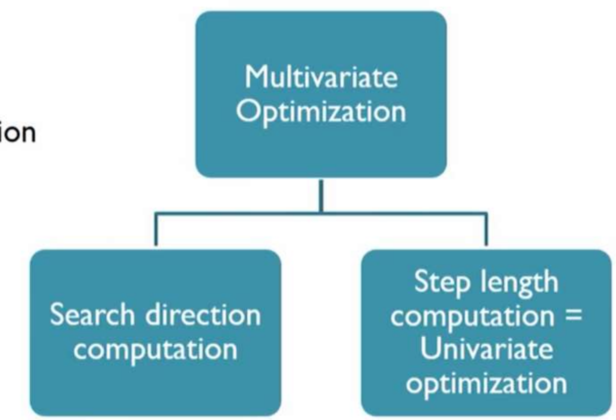


Unconstrained multivariate optimization - Descent direction and movement

- Iterative

$$x^{k+1} = \underbrace{x^k}_{\text{Starting point}} + \underbrace{\alpha^k}_{\text{Step length}} \underbrace{s^k}_{\text{Search direction}}$$

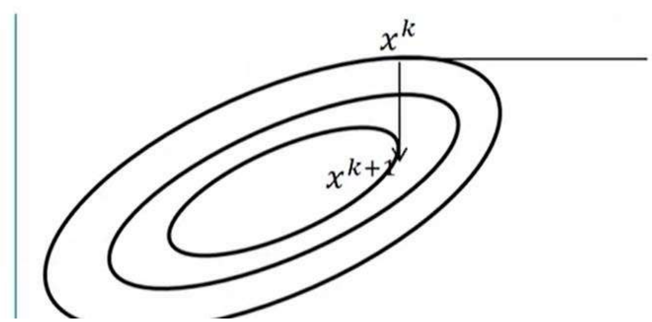




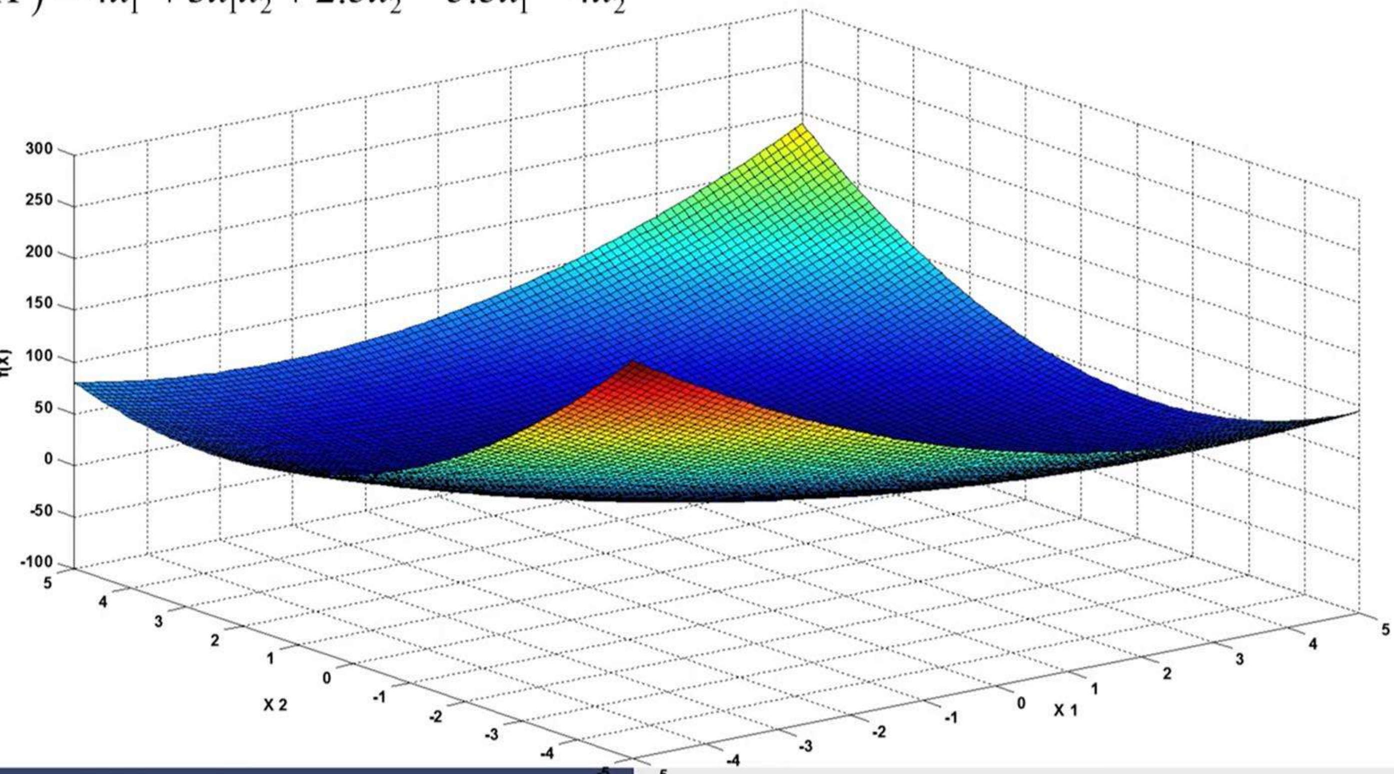
- In ML techniques, this is called as the learning rule
- In neural networks
  - Back-propagation algorithm
  - Same gradient descent with application of chain rule
- In clustering
  - Minimization of an Euclidean distance norm

Steepest descent and optimum step size

- Minimize $f(x_1, x_2, \ldots x_n) = f(x)$
- **Steepest descent**
  - At iteration k starting point is x^k
  - Search direction s^k = Negative of gradient of $f(x) = -\nabla f(x^k)$
  - New point is $x^{k+1} = x^k + \alpha^k s^k$ where α^k is the value of α for which $f(x^{k+1}) = f(\alpha)$ = is a minimum (univariate minimization)



$$f\left(X\right)=4x_1^2+3x_1x_2+2.5x_2^2-5.5x_1-4x_2$$



$$f'(X) = \begin{bmatrix} 8x_1 + 3x_2 - 5.5 \\ 3x_1 + 5x_2 - 4 \end{bmatrix}$$

Learning parameter (α) = 0.135

Initial guess $(X_0) = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ $\quad f(X_0) = 19$

Gradient Descent (or) Learning Rule in ML

Step 1: $X_1 = X_0 - \alpha f'(X_0)$

$$X_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} - 0.135 \begin{bmatrix} 8x_{0,1} + 3x_{0,2} - 5.5 \\ 3x_{0,1} + 5x_{0,2} - 4 \end{bmatrix}$$

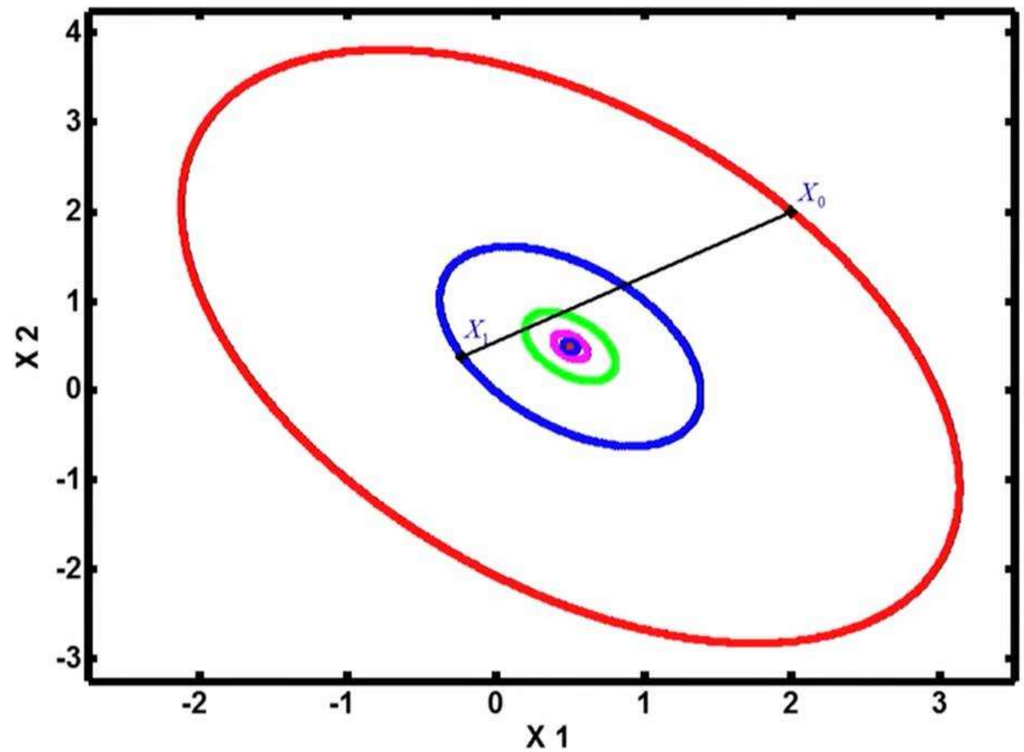
$$X_1 = \begin{bmatrix} 2 \\ 2 \end{bmatrix} - 0.135 \begin{bmatrix} 8(2) + 3(2) - 5.5 \\ 3(2) + 5(2) - 4 \end{bmatrix}$$

$$X_1 = \begin{bmatrix} -0.2275 \\ 0.3800 \end{bmatrix} \quad f(X_1) = 0.0399$$

Constant objective function contour plots

$$f(X) = 4x_1^2 + 3x_1x_2 + 2.5x_2^2 - 5.5x_1 - 4x_2 = K$$

Quadratic in this case - ellipse



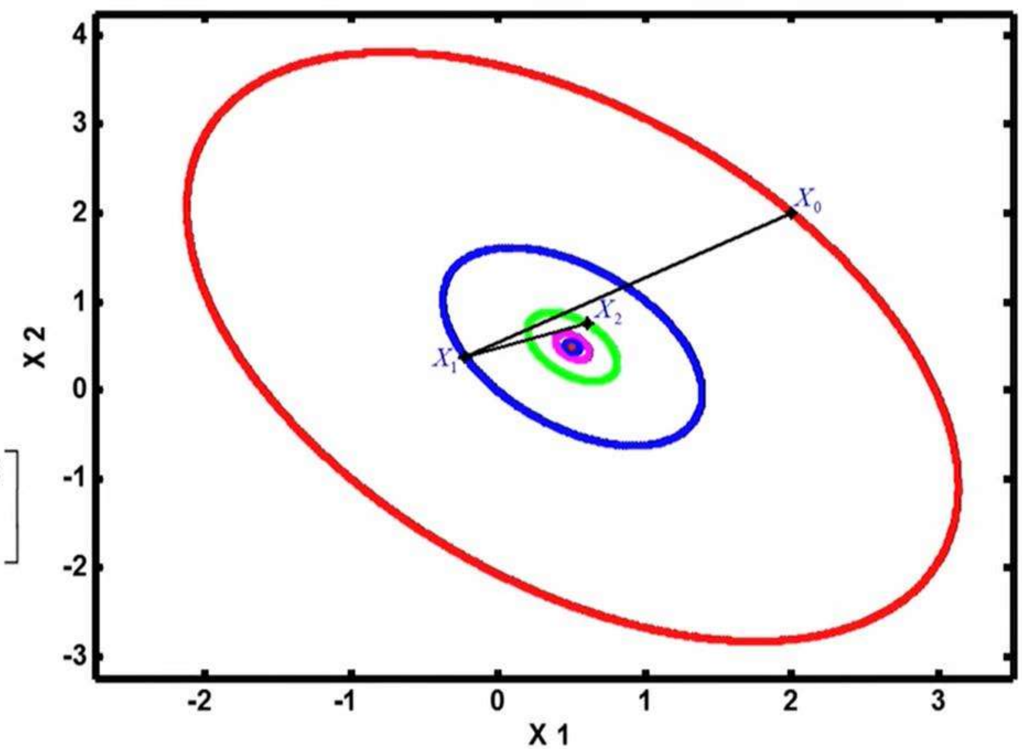
First iteration $(X_1) = \begin{bmatrix} -0.2275 \\ 0.3800 \end{bmatrix}$

Step 2: $X_2 = X_1 - \alpha f'(X_1)$

$$X_2 = \begin{bmatrix} -0.2275 \\ 0.3800 \end{bmatrix} - 0.135 \begin{bmatrix} 8x_{1,1} + 3x_{1,2} - 5.5 \\ 3x_{1,1} + 5x_{1,2} - 4 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} -0.2275 \\ 0.3800 \end{bmatrix} - 0.135 \begin{bmatrix} 8(-0.2275) + 3(0.3800) - 5.5 \\ 3(-0.2275) + 5(0.3800) - 4 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} 0.6068 \\ 0.7556 \end{bmatrix} \qquad f(X_2) = -2.0841$$



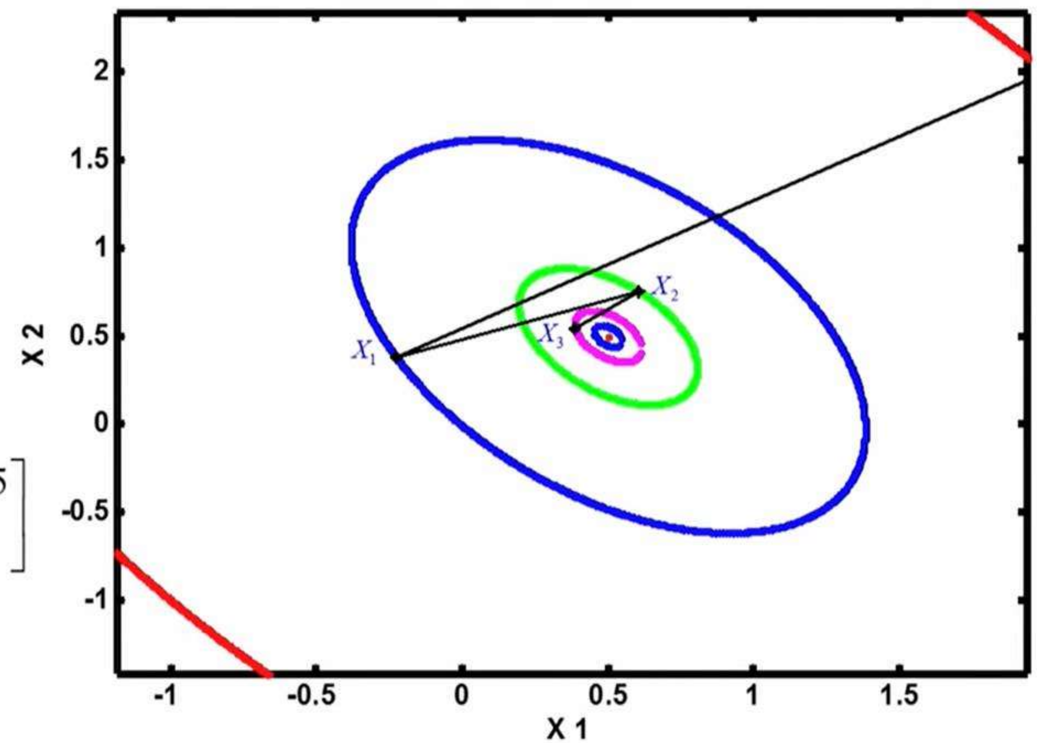
Second iteration $(X_2) = \begin{bmatrix} 0.6068 \\ 0.7556 \end{bmatrix}$

Step 3: $X_3 = X_2 - \alpha f'(X_2)$

$$X_3 = \begin{bmatrix} 0.6068 \\ 0.7556 \end{bmatrix} - 0.135 \begin{bmatrix} 8x_{2,1} + 3x_{2,2} - 5.5 \\ 3x_{2,1} + 5x_{2,2} - 4 \end{bmatrix}$$

$$X_3 = \begin{bmatrix} 0.6068 \\ 0.7556 \end{bmatrix} - 0.135 \begin{bmatrix} 8(0.6068) + 3(0.7556) - 5.5 \\ 3(0.6068) + 5(0.7556) - 4 \end{bmatrix}$$

$$X_3 = \begin{bmatrix} 0.3879 \\ 0.5398 \end{bmatrix} \qquad f(X_3) = -2.3342$$



Third iteration $\left(X_3\right)=\begin{bmatrix} 0.3879 \\ 0.5398 \end{bmatrix}$

Step 4: $X_4 = X_3 - \alpha f'\left(X_3\right)$

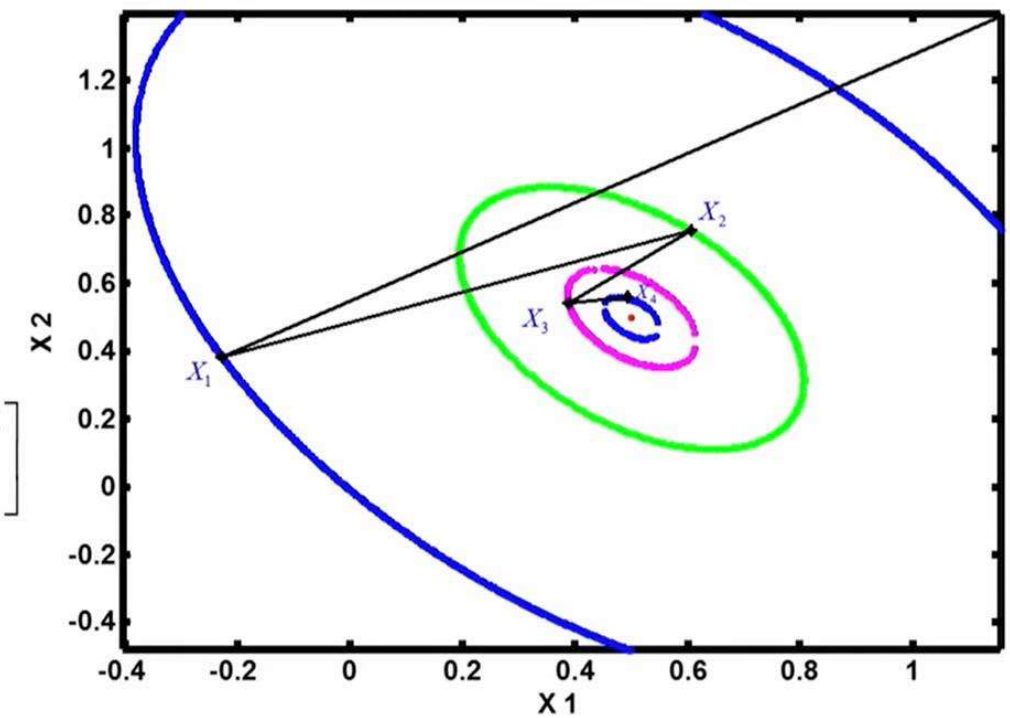
$$X_4 = \begin{bmatrix} 0.3879 \\ 0.5398 \end{bmatrix} - 0.135 \begin{bmatrix} 8x_{3,1} + 3x_{3,2} - 5.5 \\ 3x_{3,1} + 5x_{3,2} - 4 \end{bmatrix}$$

$$X_4 = \begin{bmatrix} 0.3879 \\ 0.5398 \end{bmatrix} - 0.135 \begin{bmatrix} 8(0.3879) + 3(0.5398) - 5.5 \\ 3(0.3879) + 5(0.5398) - 4 \end{bmatrix}$$

$$X_4 = \begin{bmatrix} 0.4928 \\ 0.5583 \end{bmatrix} \qquad f\left(X_4\right) = -2.3675$$

Optimal solution $\left(X_{opti}\right)=\begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$ $\quad f\left(X_{opti}\right) = -2.3750$

Gradient is zero at the optimum point



Third iteration $(X_3) = \begin{bmatrix} 0.3879 \\ 0.5398 \end{bmatrix}$

$$X^{k+1} \leftarrow X^{k}$$
$$f(X^{k+1}) \leftarrow f(X^{k})$$
$$\nabla f(X^{k+1}) \leftarrow \nabla f(X^{k})$$

Step 4: $X_4 = X_3 - \alpha f'(X_3)$

$$X_4 = \begin{bmatrix} 0.3879 \\ 0.5398 \end{bmatrix} - 0.135 \begin{bmatrix} 8x_{3,1} + 3x_{3,2} - 5.5 \\ 3x_{3,1} + 5x_{3,2} - 4 \end{bmatrix}$$

$$X_4 = \begin{bmatrix} 0.3879 \\ 0.5398 \end{bmatrix} - 0.135 \begin{bmatrix} 8(0.3879) + 3(0.5398) - 5.5 \\ 3(0.3879) + 5(0.5398) - 4 \end{bmatrix}$$

$$X_4 = \begin{bmatrix} 0.4928 \\ 0.5583 \end{bmatrix} \qquad f(X_4) = -2.3675$$

Optimal solution $(X_{opti}) = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$ $\quad f(X_{opti}) = -2.3750$

Gradient is zero at the optimum point