- **Variants of Gradient descent:**

  Let's assume that we are minimizing a loss function (f) while training a model. The update using Gradient descent is given by:

$$\theta^{t+1} = \theta^t - \eta \sum_{i=1}^{n} \frac{\partial f(x_i)}{\partial \theta}$$

  We use all the data points for one update, which leads to a high computation time if our dataset is very large.

  So we have some variants of Gradient descent that can help us with the problem.

  1. **Batch Gradient descent** calculates the partial derivative using only a few data points from our data set randomly while performing many updates. i.e.

$$\theta^{t+1} = \theta^t - \eta \sum_{i \in B} \frac{\partial f(x_i)}{\partial \theta}$$

     where B is a **random sample** of our data points.
     We get very high-speed improvement while training our model with almost a similar accuracy.

  2. **Stochastic Gradient descent** updates the parameters for each training example one by one.

     i.e. $$\theta^{t+1} = \theta^t - \eta . \frac{\partial f(x_k)}{\partial \theta}$$

     where k is a random number from 1 to n.

     It is comparatively faster than Batch GD but the number of updates needed to reach the minima is large.

Constrained Optimization Problem

- For a **constrained optimization** problem, we have an objective function that we are trying to optimize (say, $min_{x,y}$ f(x, y)) and this objective function will be subjected to some constraints.

The constraint may be an **equality constraint** ($g(x, y) = 0$) or we can also have **inequality constraints** like $g(x, y) < c$.

- The **method of Lagrange multipliers** is a method of finding the local minima or local maxima of a function subject to equality or inequality constraints.

  We want to solve the problem $x^*, y^* = min_{x,y} f(x, y)$
  subjected to the constraint $g(x, y) = c$

  To solve the above problem, we **combine** both the constraint and the objective function.
  We can write the constraint as $g(x, y) - c$ and then rewrite our problem as:

  $x^* y^* = min_{x,y} f(x, y) + \lambda(g(x, y) - c) = L(x, y, \lambda)$

  Here λ is called a **Lagrange multiplier** ($\lambda \geq 0$) and the function $L(x, y, \lambda)$ is called the **Lagrangian function.**

  Example: $min_{x,y} \sum_{i=1}^{n} - y_i(w^T xi + w0)$, subjected to the constraint $||w||^2 = 1$

  We can rewrite the constraint as $||w||^2 - 1 = 0$.

  Using the Lagrange multiplier, we can convert it into an unconstrained optimization problem.

  i.e. $L = min_{x,y} \sum_{i=1}^{n} - y_i(w^T xi + w0) + \lambda(||w||^2 - 1), \quad \lambda \geq 0$

  We can solve for the optimal value using the Gradient Descent algorithm.