Chapter 2:
Convex Optimization

2.1 Convex Functions

Definition: Convex Set *A set $\mathcal{X} \subset \mathbb{R}^d$ is called convex if and only if the line segment connecting any two points in $\mathcal{X}$ entirely lies within $\mathcal{X}$, that is,*

$$\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}, \quad \forall \theta \in [0,1] : (1-\theta)\mathbf{x}_1 + \theta \mathbf{x}_2 \in \mathcal{X}.$$

Example: Hyperplanes are convex sets
Let $H = \{\mathbf{x} \in \mathbb{R}^d : \mathbf{w}^T\mathbf{x} + b = 0\}$ be a hyperplane. We will now show that hyperplanes are convex.

Let $\mathbf{x_1}, \mathbf{x_2} \in H$ and $\theta \in [0,1]$. Then $\mathbf{w}^T\mathbf{x_1} = -b$, $\mathbf{w}^T\mathbf{x_2} = -b$ and

$$\mathbf{w}^T((1-\theta)\mathbf{x}_1 + \theta\mathbf{x}_2) + b = (1-\theta)\mathbf{w}^T\mathbf{x_1} + \theta\mathbf{w}^T\mathbf{x_2} + b = (1-\theta)(-b) - \theta b + b = 0.$$

Definition: Convex functions *A function $f : \mathbb{R}^d \to \mathbb{R}$ is convex if and only if the set above the graph is convex or equivalently,*

$$\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^d, \forall \theta \in [0,1] : f\left((1-\theta)\mathbf{x}_1 + \theta\mathbf{x}_2\right) \leq (1-\theta) f\left(\mathbf{x}_1\right) + \theta f\left(\mathbf{x}_2\right)$$

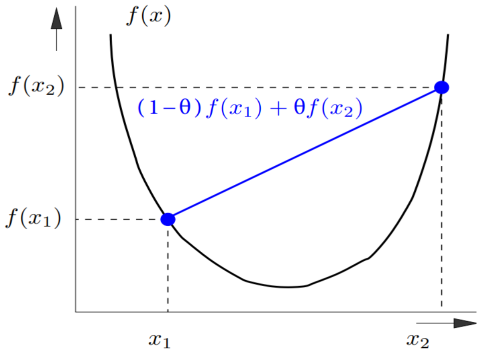

Figure 1: A convex function

Definition: Concave function *A function f is concave if and only if $-f$ is convex.*

Recall from Chapter 0 that f is convex if and only if the Hessian matrix $H_f(\mathbf{x})$ is positive semi-definite for all $\mathbf{x} \in \mathbb{R}^d$.

Theorem 2.1.1 The following statements regarding a symmetric matrix $M \in \mathbb{R}^{d\times d}$ are equivalent:

- M is positive semi-definite (we write $M \succeq 0$)
- $\mathbf{x}^\top M\mathbf{x} \geq 0$ for all $\mathbf{x} \in \mathbb{R}^d$
- All eigenvalues of M are non-negative.
- All principal minors of M are positive

Example: Convex Function
Define f as follows :

$$\begin{aligned} f : \mathbb{R}^2 &\to \mathbb{R} \\ (x_1, x_2)^T &\mapsto x_1^2 + x_2^2 + 8. \end{aligned}$$

Note that

$$\nabla f \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 2x_1 \\ 2x_2 \end{pmatrix} \text{ and } H_f(\mathbf{x}) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}.$$

Observe that $H_f(\mathbf{x})$ is a diagonal matrix and we can read the eigenvalues from the diagonal.
As the eigenvalues are non negative it follows that $H_f(\mathbf{x})$ is positive semi-definite and that f is convex.

2.2 Convex Optimization Problems

Let $f_0, f_1, \ldots, f_n, g_1, \ldots, g_m : \mathcal{X} \to \mathbb{R}$.

Definition: Optimization Problem
An optimization problem (OP) is of the form:

$$\begin{array}{rll} \min_{\mathbf{x}\in\mathbb{R}^d} & f_0(\mathbf{x}) & \\ \text{s.t.} & f_i(\mathbf{x}) \leq 0, & i = 1, \ldots, n \\ & g_j(\mathbf{x}) = 0, & j = 1, \ldots, m \end{array}$$

where we call:

- f_0 the **objective function.**
- $f_i \quad \forall i = 1 \ldots n$ the **inequality constraints.**
- $g_j \quad \forall j = 1 \ldots m$ the **equality constraints.**.

We observe that this special form is not really a restriction. As for a set S we have that

$$\max_{x\in S} x = -\min_{x\in S} -x.$$

The inequality constraints can be brought in this form by bringing everything to one side and multiplying by -1 if necessary.

Definition: Convex Optimization Problem *An OP is called convex if and only if the functions $f_0, f_1, \ldots, f_n$ are convex and $g_1, \ldots, g_m$ are linear.*

Example: Convex Optimization Problem
Recall the hard margin SVM.

$$\begin{aligned} \max_{\gamma, b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d \setminus \{0\}} & \gamma \\ \text{s.t.} \quad & y_i \left(\mathbf{w}^\top \mathbf{x}_i + b \right) \geq \|\mathbf{w}\| \gamma \quad \forall i = 1 \ldots n \end{aligned}$$

Let us check if this is a convex OP. By the definition above we need to check whether the following is convex for each i:

$$f(\gamma, b, \mathbf{w}) = \|\mathbf{w}\| \gamma \; - y_i \left(\mathbf{w}^\top \mathbf{x}_i + b \right).$$

We can show that even for the case $b = 0$ and $\mathbf{w} > 0 \in \mathbb{R}_+$ the function is not convex. Explicitly for the case:

$$f(\gamma, w) = w\gamma - y_i(wx_i)$$

Let us calculate the gradient:

$$\nabla f \begin{pmatrix} \gamma \\ w \end{pmatrix} = \begin{pmatrix} w \\ \gamma - y_i x_i \end{pmatrix}$$

Then the hessian matrix

$$H_f(\mathbf{x}) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

is not positive semi-definite as $\det(H_f(x)) = -1$ which shows that the hard margin SVM is not a convex OP.

2.3 Making the SVM convex

In the previous example we saw that the hard margin SVM is not a convex OP but as we will show, we can make it convex.

Theorem 2.3.1 The linear hard margin SVM that is,

$$\max_{\gamma, b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d \setminus \{0\}} \gamma \text{ s.t. } y_i \left(\mathbf{w}^\top \mathbf{x}_i + b \right) \geq \|\mathbf{w}\| \gamma$$

can be equivalently rewritten in convex form as given below:

$$\begin{aligned} \min_{\boldsymbol{b} \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d} \quad & \tfrac{1}{2} \|\mathbf{w}\|^2 \\ \text{s.t.} \quad & 1 - y_i \left(\mathbf{w}^\top \mathbf{x}_i + b \right) \leq 0 \quad \forall i = 1, \ldots, n \end{aligned}$$

Proof:
We show the proof, for data that is linearly separable, to avoid special cases. The theorem also holds when $\gamma = 0$ is the only solution.
Our SVM will give us parameters $\mathbf{w}$ and b which we will use to define the classifier

$$f(\mathbf{x}) = \operatorname{sign}\left(\mathbf{w}^\top \mathbf{x} + b\right).$$

Notice that these parameters are not unique as

$$\forall \lambda > 0 : \operatorname{sign}\left(\mathbf{w}^\top \mathbf{x} + b\right) = \operatorname{sign}(\underbrace{\lambda \mathbf{w}^T}_{=:\mathbf{w}_\lambda} \mathbf{x} + \underbrace{\lambda b}_{=:b_\lambda}). \tag{$\star$}$$

Apparently any $\mathbf{w}_\lambda$ will work. The idea is to choose a $\mathbf{w}_\lambda$ with an appropriate norm. So we start with:

$$\max_{\gamma, b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d \setminus \{0\}} \gamma \text{ s.t. } y_i \left(\mathbf{w}^\top \mathbf{x}_i + b\right) \geq \|\mathbf{w}\| \gamma. \tag{1}$$

Observe that for arbitrary $\mathbf{w}$,b$=-\mathbf{w}^T \mathbf{x_i}$,$\gamma = 0$ satisfies the constraints and has objective function value 0. Apparently 0 is a lowerbound. So we can write (1) in the form:

$$\max_{\gamma \geq 0, b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d \setminus \{0\}} \gamma \text{ s.t. } y_i \left(\mathbf{w}^\top \mathbf{x}_i + b\right) \geq \|\mathbf{w}\| \gamma. \tag{2}$$

We assume our data to be linearly separable, so $\gamma = 0$ will not be an optimal solution. So it suffices to find a convex OP for

$$\max_{\gamma > 0, b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d \setminus \{0\}} \gamma \text{ s.t. } y_i \left(\mathbf{w}^\top \mathbf{x}_i + b\right) \geq \|\mathbf{w}\| \gamma. \tag{3}$$

Now observe that $\max \gamma$ has the same behavior as $\min \dfrac{1}{2\gamma^2}$. Decreasing/increasing γ has the same effect on both. So we can write (3) as

$$\min_{\gamma > 0, b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d \setminus \{0\}} \frac{1}{2\gamma^2} \text{ s.t. } y_i \left(\mathbf{w}^\top \mathbf{x}_i + b\right) \geq \|\mathbf{w}\| \gamma. \tag{4}$$

According to ($\star$) we get that (4) is equivalent to

$$\min_{\gamma > 0, b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d \setminus \{0\}} \quad \frac{1}{2\gamma^2} \quad \text{s.t.} \quad y_i \left(\lambda \mathbf{w}^\top \mathbf{x}_i + \lambda b\right) \geq \lambda \|\mathbf{w}\| \gamma. \tag{5}$$

We can rename λb as b as both are any vector in $\mathbb{R}^d$. So (5) is equivalent to

$$\min_{\gamma > 0, b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d \setminus \{0\}} \quad \frac{1}{2\gamma^2} \quad \text{s.t.} \quad y_i \left(\lambda \mathbf{w}^\top \mathbf{x}_i + b\right) \geq \lambda \|\mathbf{w}\| \gamma. \tag{6}$$

Choose $\lambda := \dfrac{1}{\|\mathbf{w}\| \, \gamma} > 0$ to write (6) in the form

$$\min_{\gamma > 0, b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d \setminus \{0\}} \frac{1}{2\gamma^2} \text{ s.t. } y_i \left(\frac{\mathbf{w}^\top \mathbf{x}_i}{\|\mathbf{w}\| \gamma} + b\right) \geq 1 \tag{7}$$

Substituting $\tilde{\mathbf{w}} := \dfrac{\mathbf{w}}{\|\mathbf{w}\|\gamma}$ and by rewriting $\|\tilde{\mathbf{w}}\| = \left\| \dfrac{\mathbf{w}}{\|\mathbf{w}\|\gamma} \right\| = \dfrac{\|\mathbf{w}\|}{\|\mathbf{w}\|\gamma} = 1/\gamma$ we get

$$\min_{b\in\mathbb{R},\tilde{w}\in\mathbb{R}^d\setminus\{0\}} \frac{1}{2}\|\tilde{\mathbf{w}}\|^2 \quad \text{s.t.} \quad y_i\left(\tilde{\mathbf{w}}^\top \mathbf{x}_i + b\right) \geq 1 \tag{8}$$

Assuming the set of data points contains at least 1 point from each class ($\exists i, j : (\mathbf{x}_i, 1), (\mathbf{x}_j, -1)$ is in our dataset), we can now allow $\tilde{\mathbf{w}} = 0$, since $y_i\left(\tilde{\mathbf{w}}^\top \mathbf{x}_i + b\right) = y_i b \geq 1$ cannot be satisfied for $\tilde{\mathbf{w}} = 0$. As such $\mathbf{w} = 0$ will never be in the set we optimize over anyway. Finally we get (8) into

$$\min_{b\in\mathbb{R},\mathbf{w}\in\mathbb{R}^d} \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{s.t.} \quad 1 - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) \leq 0. \quad \square \tag{9}$$

It is easy to verify that (9) is indeed convex. Analogously one can prove the following theorem for soft margin SVM.

Theorem 2.3.2 We can formulate the linear soft margin SVM that is

$$\begin{aligned} \max_{\gamma,b\in\mathbb{R},\mathbf{w}\in\mathbb{R}^d\setminus\{0\},\xi_1,\ldots,\xi_n\geq 0} \quad & \gamma - C\sum_{i=1}^n \xi_i \\ \text{s.t.} \quad & y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) \geq \|\mathbf{w}\|\gamma - \xi_i, \quad && \forall i = 1, \ldots n \\ & \xi_i \geq 0 && \forall i = 1, \ldots n \end{aligned}$$

as a convex OP of the form :

$$\begin{array}{cl} \displaystyle\min_{\mathbf{b}\in\mathbb{R},\mathbf{w}\in\mathbb{R}^d,\boldsymbol{\xi}\in\mathbb{R}^n} & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^n \xi_i \\ \text{s.t.} & 1 - \xi_i - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) \leq 0, \quad -\xi_i \leq 0 \quad \forall i = 1, \ldots n. \end{array}$$

2.4 Solving convex optimization problems

We saw that we can rewrite our soft/hard margin SVM as convex optimization problems. Now it remains to find an algorithm to solve them. We will use the following useful theorem.

Theorem 2.4.1 *Every locally optimal point of a convex optimization problem is also globally optimal.*

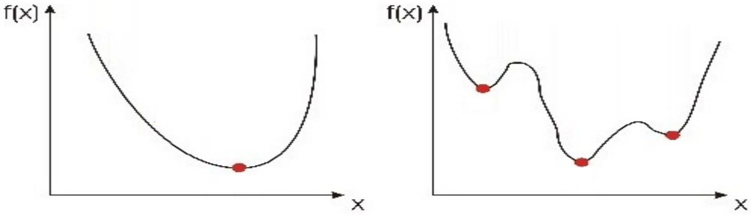

Figure 2: Left we see a convex function, to the right a nonconvex function

Apparently it suffices to only search for local optimum to find a global one in convex functions. The algorithm we will use will be gradient descent. The idea is to start at a random point and continuously go in the direction of steepest descent. The direction of steepest descent is the negative of the direction of steepest ascent, which is the direction of the gradient.

Algorithm Gradient Descent

Input: A convex function f, point $\mathbf{x_1}$ (e.g chosen randomly)

```
function GRADDESCENT($f, \mathbf{x_1}$)
    for $t \leftarrow 1$ to $T$ do
        $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \lambda_t \nabla f_0 (\mathbf{x}_t)$
    end for
    return $\mathbf{x_T}$
end function
```

Definition: Step size *We call λ_t the **step size** or learning rate.*
Observe that finding a good step size is important. If the step size is too small we need a higher T or longer to reach a global minimum. If the step size is too big we can overshoot the minimum or the algorithm might not find a minimum at all. A typical choice would be $\lambda_t := \frac{1}{t}$.

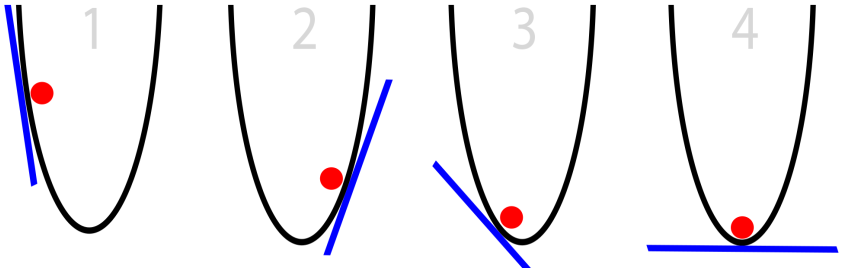

Figure 3: Shows the behavior of the gradient descent algorithm.

Theorem 2.4.2 Let $f_0 : \mathbb{R}^d \to \mathbb{R}$ be an arbitrary (possibly non-convex) objective function. Then, under some assumptions, [1]. we have:

- Gradient descent converges towards a stationary point(maximum, minimum, saddle point) which is in practical purposes usually a minimum.
- For ideal choice of the learning rate, the convergence rate is at least as good as:
$$f_0\left(\mathbf{x}_t\right) - f_0\left(\mathbf{x}^*_{\text{local}}\right) \leq O(1/t).$$

2.5 Applying gradient descent on the convex SVM

Let us recall the convex soft margin SVM

$$\begin{array}{cl} \min\limits_{\mathbf{b}\in\mathbb{R},\mathbf{w}\in\mathbb{R}^d,\boldsymbol{\xi}\in\mathbb{R}^n} & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^n \xi_i \\ \text{s.t.} & 1 - \xi_i - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) \leq 0, \quad -\xi_i \leq 0 \quad \forall i = 1, \ldots n. \end{array}$$

If we look at the gradient descent algorithm, we notice that the input is a convex function. But our OP has one objective function and two inequality constraints for each datapoint.
A quick fix would be to include the two inequality constraints in the objective function. Let us look at the inequality constraints

$$1 - \xi_i - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) \leq 0 \iff 1 - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) \leq \xi_i$$

and

$$-\xi_i \leq 0 \iff 0 \leq \xi_i.$$

Together we get that

$$\max\left\{1 - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right), 0\right\} \leq \xi_i$$

[1] The theorem assumes Lipschitz-continuous gradients with a uniformly bounded Lipschitz constant $\exists L : \|\nabla f(\mathbf{x}) - \nabla f(\tilde{\mathbf{x}})\| \leq L\|\mathbf{x} - \tilde{\mathbf{x}}\|$. Convergence is guaranteed for all learning rate schedules satisfying $\sum_{t=1}^\infty \lambda_t = \infty$ and $\lambda_t \frac{-\cdots}{t\to\infty} 0$, but with varying rates of convergence. The favorable $O(1/t)$ rate is achieved using the minimzation rule: $\lambda_t := \arg\min_\lambda f_0\left(\mathbf{x}_t - \lambda\nabla f_0\left(\mathbf{x}_t\right)\right)$

which we can safely plug in as an equality into our objective function as we are minimizing, resulting in the following Proposition:

Proposition 2.5.1 *The convex soft margin SVM can be rewritten as*

$$\min_{b\in\mathbb{R},\mathbf{w}\in\mathbb{R}^d} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \max\left(0, 1 - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right)\right)$$

Our new unconstrained objective becomes

$$f_0(\mathbf{w}, b) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \max\left(0, 1 - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right)\right).$$

Let us look at the part

$$\max\left(1 - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right), 0\right).$$

This part of the function is not differentiable. We can see this by replacing $y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right)$ (a linear growth function) by z resulting in the so called **hinge function**

$$\max\left(1 - z, 0\right).$$

which has two tangent lines at $z = 1$, one with slope -1 and the other with slope 0. A potential solution would be to just choose any of the derivatives at this point, so we consider the subgradient :

$$\nabla \max\left(1 - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right), 0\right) := \begin{cases} \nabla(1 - y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right)) & \text{if } y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) < 1 \\ 0 & \text{else} \end{cases}.$$

Finally we use the subgradient descent algorithm, which is the normal gradient descent algorithm but using the subgradient in place of the gradient.
An alternative solution could be to replace this function with something differentiable that has similar behavior.
Let us look at the **logistic function**

$$l(z) = \ln(1 + \exp(-z)).$$

Definition: Logistic Regression *Replacing the hinge function in the unconstrained convex soft margin SVM by the logistic function we get the* ***Logistic Regression***

$$f_0(\mathbf{w}, b) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \ln(1 + \exp(y_i(\mathbf{w}^T \mathbf{x_i} + b))).$$

As the Logistic Regression is differentiable, we can use our normal gradient descent algorithm.

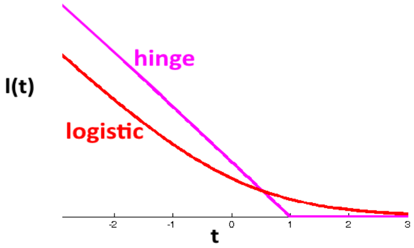

Figure 4: We see that the logistic function is a differentiable approximation to the hinge function

Let us discuss a final problem. Notice we need to iterate through all datapoints to calculate the (sub)gradient. That would mean n iterations, where n can be large for big data.
A solution would be to try and approximate the value of the function, leading us to the following algorithm:

Algorithm Stochastic Subgradient Descent (SGD) SVM

Input: A starting point $(\mathbf{w}, b)_1$

```
1: for $t \leftarrow 1$ to $T$ do
2:     Randomly select $B$ datapoints
3:     Denote their indices by $I \subseteq \{1 \dots n\}$
4:     $(\mathbf{w}, b)_{t+1} \leftarrow (\mathbf{w}, b)_t - \lambda_t \nabla \left( \frac{1}{2}\|\mathbf{w}\|^2 + \frac{Cn}{B} \sum_{i \in I} \max\left(0, 1 - y_i \left(\mathbf{w}^\top \mathbf{x}_i + b\right)\right) \right)$
5: end for
6: return $\mathbf{x_T}$
```

Observation: We are approximating

$$C \sum_{i=1}^{n} \max\left(0, 1 - y_i \left(\mathbf{w}^\top \mathbf{x}_i + b\right)\right)$$

by

$$\frac{Cn}{B} \sum_{i \in I} \max\left(0, 1 - y_i \left(\mathbf{w}^\top \mathbf{x}_i + b\right)\right)$$

The scalar $\frac{n}{B}$ is necessary to rescale the value up as we only calculated the value for B points. The value of $B \in [1 \dots N]$ called the **batch size** needs to be chosen beforehand. We can use the exact same algorithm also for the logistic regression, by just substituting it in step 4 of the algorithm.

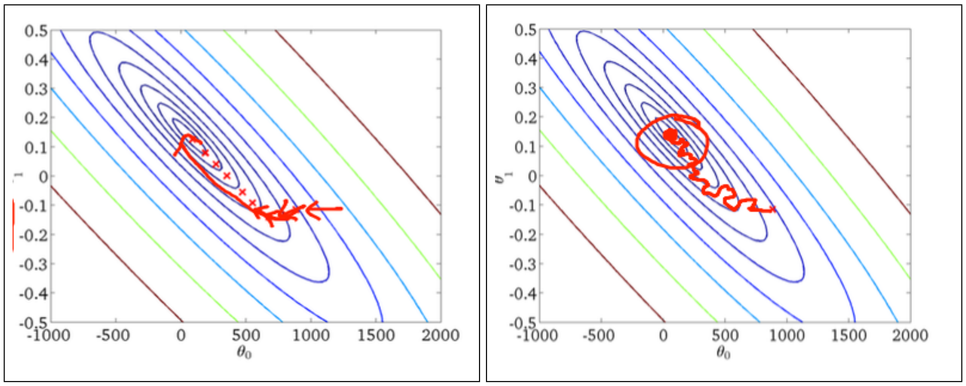

Figure 5: Visualization of the gradient descent algorithms. To the left we have the classical gradient descent algorithm and to the right the stochastic gradient descent. Notice how on the right we are not always perfectly going towards the steepest descent as the gradient is only approximated.

Theorem 2.5.2 Consider SGD using the learning rate $\lambda_t := 1/t$. Then, under mild assumptions, SGD converges with high probability to a stationary point which is usually a minimum with rate:

$$f_0\left(\mathbf{x}_t\right) - f_0\left(\mathbf{x}_{\text{local}}^*\right) \leq O(1/t)$$