Optimization: Algorithms, Complexity & Approximations

Anastasios Kyrillidis [*]

[*]Instructor, Computer Science at Rice University

Contributors: Nick Sapoval, Carlos Quintero Pena, Delaram Pirhayatifard, McKell Stauffer, Mohammad Taha Toghani, Senthil Rajasekaran, Gaurav Gupta, Pranay Mittal

Chapter 10

In the previous lectures, we studied non-convex optimization in the context of sparse feature selection and low rank recovery, where non-convexity is introduced by the constraints. We considered low-rank model selection in data science application and went beyond hard thresholding methods to discuss the non-convex path. We will now discuss the landscape of non-convex optimization problems in general, including the types of stationary points including saddle points and conditions that would allow escaping from these saddle points.

Saddle points, Matrix sensing

Non-convex optimization problems are NP-hard in general, although some specific cases can be solved in polynomial time. Neural networks, the classical example of non-convex optimization, cannot be even solved to global optimality without a fine grid search over the space of initial points. To further illustrate the difficulty of non-convex optimization, consider the example of homogeneous quartics.

Homogeneous Quartics. Homogeneous quartics are functions of the form

$$f(x) = \sum_{i,j=1}^{p} Q_{ij} x_i^2 x_j^2$$

If $Q \succeq 0$, then $f(x) \geq 0$, and $x = 0$ is the global minimum. However, if Q is arbitrary, $\nabla f(x)$ at zero is zero but zero can be a minimum, a maximum, or a saddle point. Checking if 0 is a global minimizer is equivalent to checking if there is a point that leads to a negative objective. Using a change of variable $u_i = x_i^2$, we transform the original objective function into $f(u) = u^\top Qu$. Looking for a non-negative u such that $u^\top Qu < 0$ is equivalent to checking if Q is co-positive, which is an NP-hard problem.

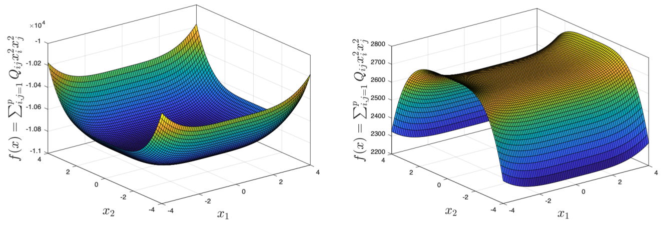

Fig. 1. For $Q \succeq 0$, 0 is the global minimum, but for any arbitrary Q, 0 can be a minimum, a maximum, or a saddle point

Previously when we studied the Newton's method, we have turned to the Hessian for information about the local curvature. However, for homogeneous quartics, $\nabla^2 f(0) = 0$, the Hessian provides no useful insights. We could use even higher-order information such as third or fourth-order derivatives, but that would propel the problem into the realm of NP-hardness. Hence, we see that in non-convex optimization, determining the identity of a stationary point is a difficult task in and of itself. Even if we were at the global minimum, proving that our solution is indeed globally minimum is NP-hard. This challenge is not only found in homogeneous quartics but in many other non-convex problems: quadratic combinatorial optimization (QCOP), matrix completion and sensing, tensor decomposition, etc.

Local minima: the next best thing to global minimum. Recall that a critical or stationary point, $x^\star$ where $\nabla f(x^\star) = 0$, can be one of the following

- Global minima: all directions go upwards and $f(x^\star) \leq f(x), \forall x$
- Local minima: all directions go upwards and maybe $f(x^\star) \geq f(x), \exists x$
- Saddle points: there are upwards, downwards, and/or flat directions

Having seen convex optimization algorithms and studied their convergence to globally optimal solutions, we may be averse to accepting local minima as solutions to non-convex problems. However, for larger models like neural networks, local minima tend to yield similar loss values as the global minimum. While poor local minima exist, it has been shown that the probability of convergence to a poor local minimum is near zero for some models. This is consistent with the fact that, in practice, training a neural network with different random seeds often leads to models that perform similarly well.

Motivation for escaping saddle points. With the knowledge that good local minima exist for some non-convex optimization problems, convergence to local minima can still be a challenging process. Saddle points can stall the convergence to a good quality local minimum. In the optimization landscape, saddle points can be large plateaus or flat regions where the slope is very slow. Saddle points can dramatically slow down learning, giving the illusion that we have reached a local minimum. Recall that for a generic smooth function, the update according to gradient descent is given by $x_{t+1} = x_t - \frac{1}{L} \nabla f(x_t)$. As t increases, gradient descent converges to the points where the gradient has zero energy.

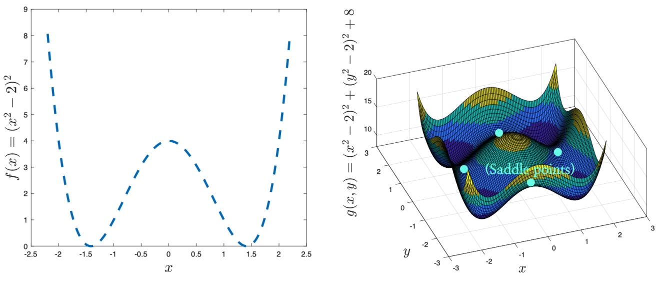

Fig. 2. Saddle points emerge and increase in number with higher dimensionality

Saddle points can be a particularly ubiquitous issue for optimization in high dimensions, as saddle points emerge and their numbers may even increase exponentially with increasing dimensionality. To illustrate this, consider the example $f(x) = (x^2-2)^2$, which has two local/global minima, one local maximum, and no saddle points.

Extending the same function from 1D to 2D, $f(x,y) = (x^2-2)^2 + (y^2-2)^2 + 8$, saddle points emerge. The 2D function's landscape resembles an egg holder, and there are 4 saddle points, one between each "slot". From 1D to 2D, the number of saddle points has increased from 0 to 4. In fact, the number of saddle points will continue to increase with higher dimensions. For the same function in 3D, we will get 8 saddle points.

Escaping saddle points: Second-order derivative test. Consider the Hessian, $\nabla^2 f(x) \in \mathbb{R}^{|\times|}$, at a critical point x. The Hessian is square and symmetric, which means that we can compute its eigendecomposition and characterize the critical point based on the signs of its eigenvalues. If $\nabla^2 f(x)$ has only positive eigenvalues, then the critical point x is a local minimum. To prove this, consider the second-order Taylor's expansion and the fact that $\nabla f(x) = 0$ at the critical point.

$$\begin{aligned} f(x+\eta u) &= f(x) + \eta\langle\nabla f(x), u\rangle + \frac{\eta^2}{2}\langle\nabla^2 f(x)u, u\rangle \\ &= f(x) + \frac{\eta^2}{2}\langle\nabla^2 f(x)u, u\rangle > f(x) \end{aligned}$$

Hence, when $\nabla^2 f(x)$ has only positive eigenvalues, all directions go upwards, and the critical point is a local minimum.

Based on similar reasoning, we can devise the following rules for characterizing a critical point.

- Only positive eigenvalues: local minimum
- Only negative eigenvalues: local maximum
- Only positive and negative eigenvalues: strict saddle point
- Positive, negative, and zero eigenvalues: general saddle point

At a saddle point, strict or general, the objective function decreases in the direction of the eigenvector that corresponds to a negative eigenvalue. By following the direction of this eigenvector, we can escape a saddle point.

Strict saddle property. A function $f(x)$ satisfies the strict saddle property, if all points x in its domain satisfies the at least one of the following:

- The gradient is large, i.e. $\|\nabla f(x)\|_2 \geq \alpha$
- The Hessian has at least one negative eigenvalue, bounded away from zero, i.e. $\lambda_{\min}(\nabla^2 f(x)) \leq -\gamma$
- x is near a local minimum

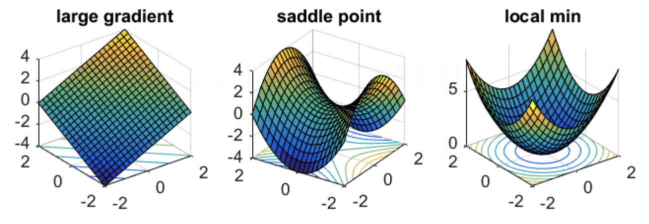

Fig. 3. Strict saddle property

For a function that satisfies this property, the minimum eigenvalue of $\nabla^2 f(x)$ is bounded by a negative value. Therefore, there is always an escape route from a saddle point of a function that satisfies this property. However, finding the minimum eigenvalue requires computing the eigendecomposition of the Hessian, which has $\mathcal{O}(p^3)$ complexity. Some existing methods such as cubic regularization and trust-region methods do not compute the full eigendecomposition but are nonetheless time consuming in practice, as they require second-order information from the Hessian.

Noisy gradient descent. The good news is that it is possible to escape from saddle points using first-order methods such as gradient descent. Although gradient at saddle points is null, strict saddle points are quite unstable. At a strict saddle point where the Hessian has no zero eigenvalue, if we perturb the our location even by just a little bit, we will fall and escape from the saddle point. We can incorporate this perturbation in the form of noise into the gradient descent step.

$$x_{t+1} = x_t - \eta\nabla f(x_t) + \epsilon, \epsilon \sim \eta \cdot \mathcal{S}^{p-1}$$

Alternatively, an even easier approach is simply using the stochastic gradient descent, which naturally has noise incorporated into each step.

$$x_{t+1} = x_t - \eta\nabla f_{i_t}(x_t) = x_t - \eta\nabla f(x_t) + \epsilon, \epsilon = \eta(\nabla f(x_t) - \nabla f_{i_t}(x_t))$$

It has been proven that noisy gradient descent finds a local minimum of an objective function that satisfies the strict saddle property in polynomial time, up to $\mathcal{O}(\frac{1}{\epsilon^4})$ iterations. To put this result in perspective, convergence of gradient descent to a critical point (not necessarily a local minimum) has a running time of $\mathcal{O}(\frac{1}{\epsilon^2})$, the noisy gradient descent converges to a local minimum but at the cost of more iterations.

A different perspective on saddle points. We have seen that strict saddle points are highly unstable, and we can escape from them with a little perturbation. Another important perspective to consider is that convergence to saddle points depends strongly on initialization. Consider the 2D example $f(x,y) = \frac{1}{2}x^2 + \frac{1}{4}y^4 - \frac{1}{2}y^2$. The only saddle point is $(0,0)$, and to converge to this saddle point, initialization has to be of the form $(x,0)$, which in the case of random initialization, occurs with a probability of 0.

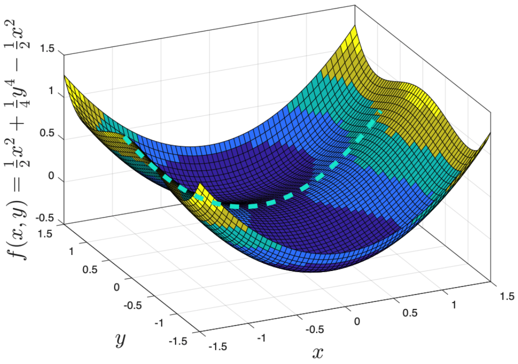

Fig. 4. Initialization has to be along the dotted line to converge to the saddle point $(0, 0)$