COMP 414/514: Optimization – Algorithms, Complexity and Approximations

Lecture 10

Overview

- In the previous lecture, we:
  - Considered **low-rank model selection** in Data Science applications
  - Followed the **non-convex path,** beyond hard thresholding methods
  - Discussed some global convergence guarantees (under proper initialization assumptions) and mentioned some open questions

Overview

- In the previous lecture, we:
  - Considered **low-rank model selection** in Data Science applications
  - Followed the **non-convex path,** beyond hard thresholding methods
  - Discussed some global convergence guarantees (under proper initialization assumptions) and mentioned some open questions
- For the next 2-3 lectures, we will worry about the **landscape** of such non-convex scenaria:
  - We will discuss about **types of stationary points**, focus on **saddle points** and study some of their properties
  - We will introduce conditions that allow **escaping from saddle points**
  - We will study matrix sensing as a test case, and how to prove ``no spurious local minima`` arguments

NP-hardness

- Non-convex continuous optimization = NP-hard in general

(Specifically can be polynomially solvable)

NP−hardness

– Non−convex continuous optimization = NP−hard in general

(Specifically can be polynomially solvable)

– Example: Homogeneous quartics

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

NP−hardness

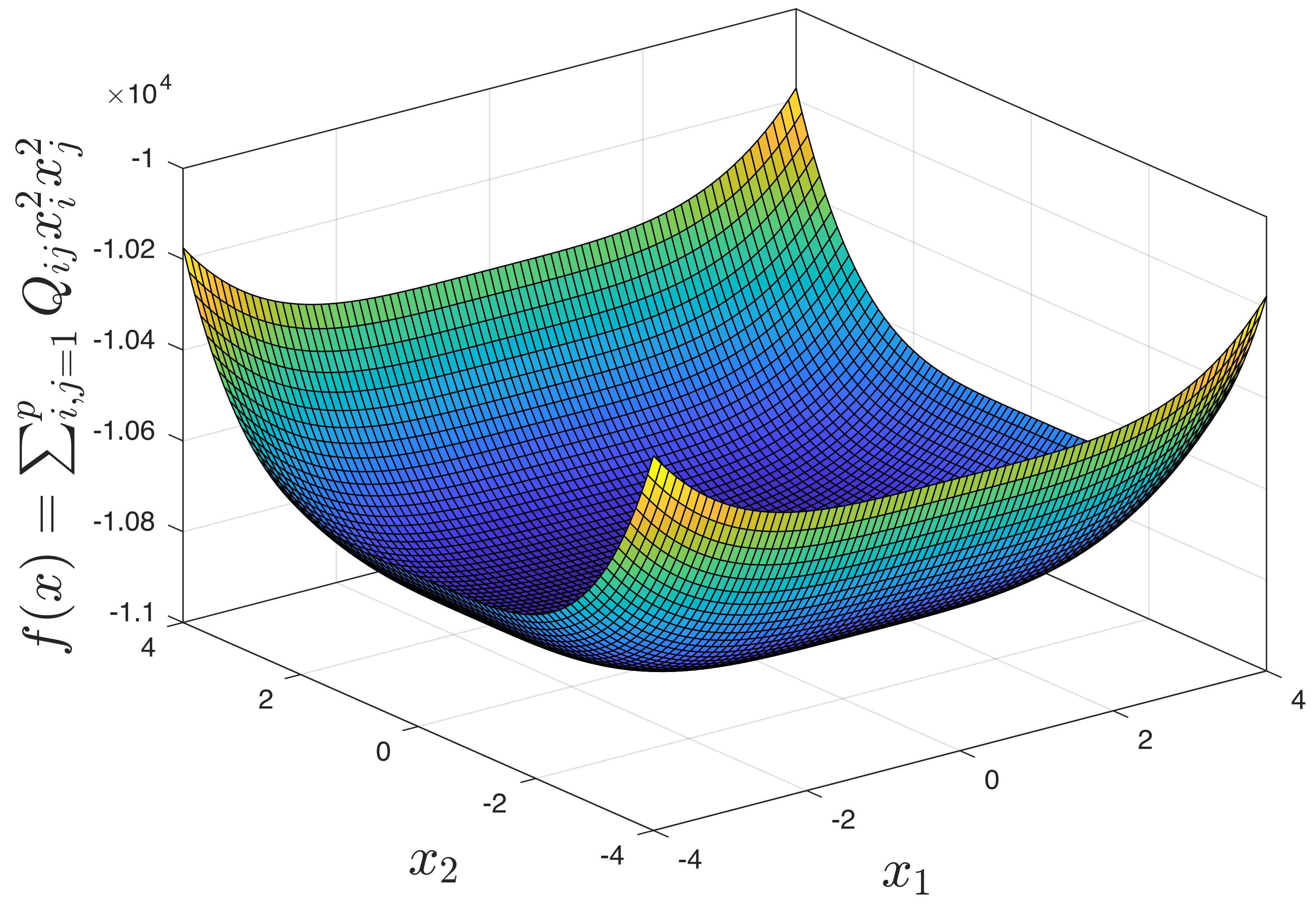
– Non−convex continuous optimization = NP−hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– $p = 2,\ Q \succeq 0$



NP−hardness

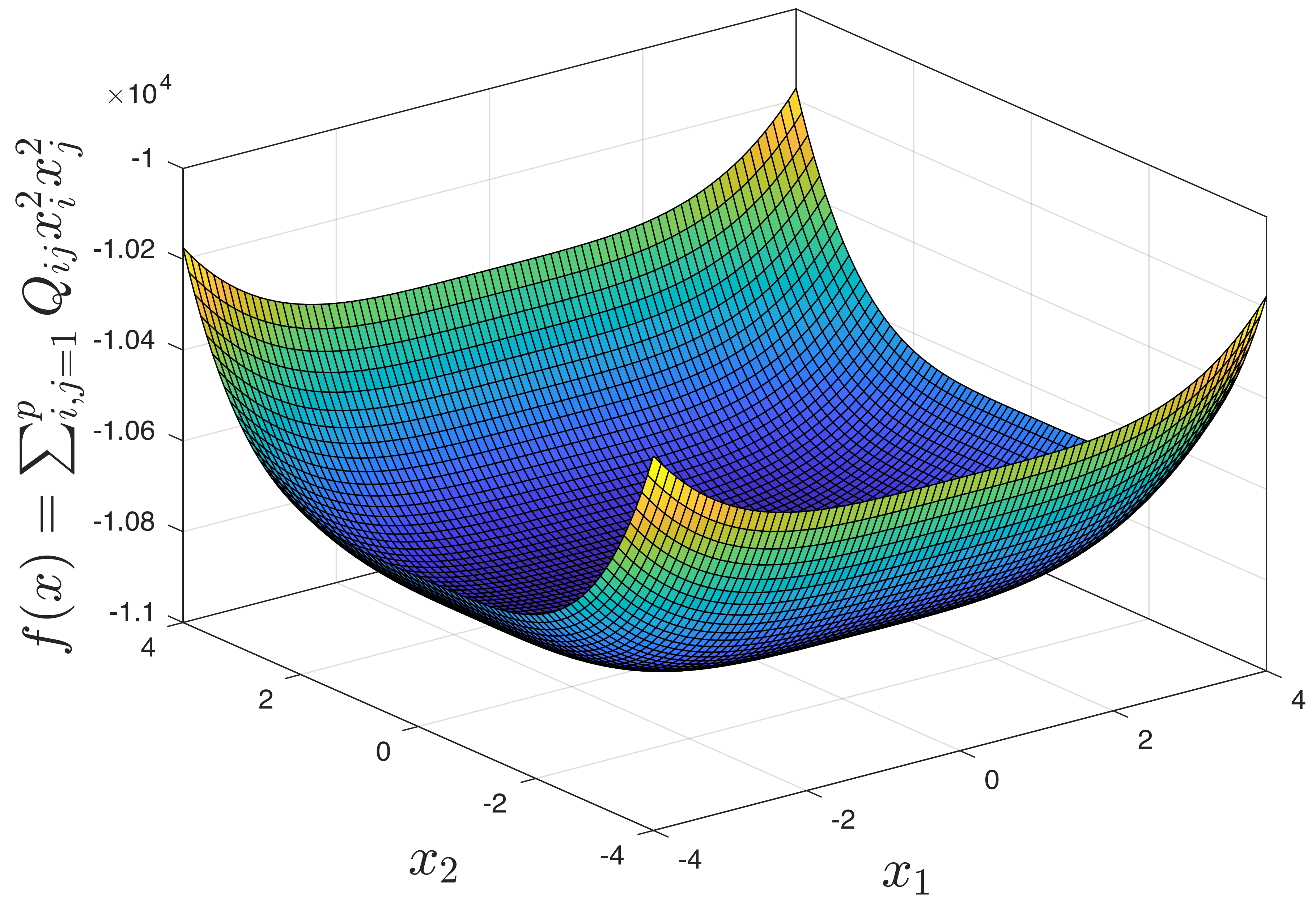
– Non−convex continuous optimization = NP−hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– $p = 2,\ Q \succeq 0$



NP–hardness

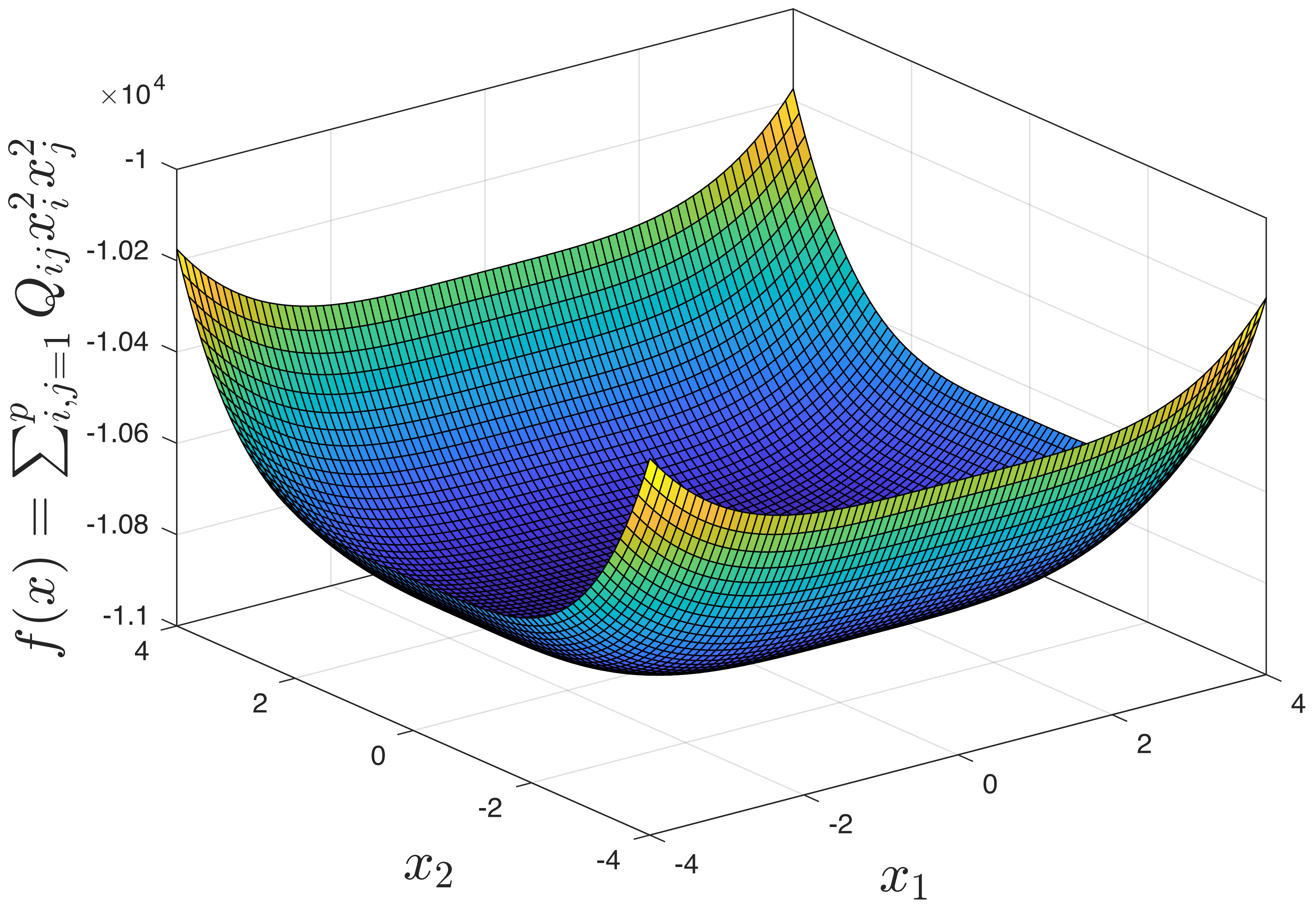
– Non–convex continuous optimization = NP–hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– $p = 2,\ Q \succeq 0$



NP−hardness

– Non−convex continuous optimization = NP−hard in general

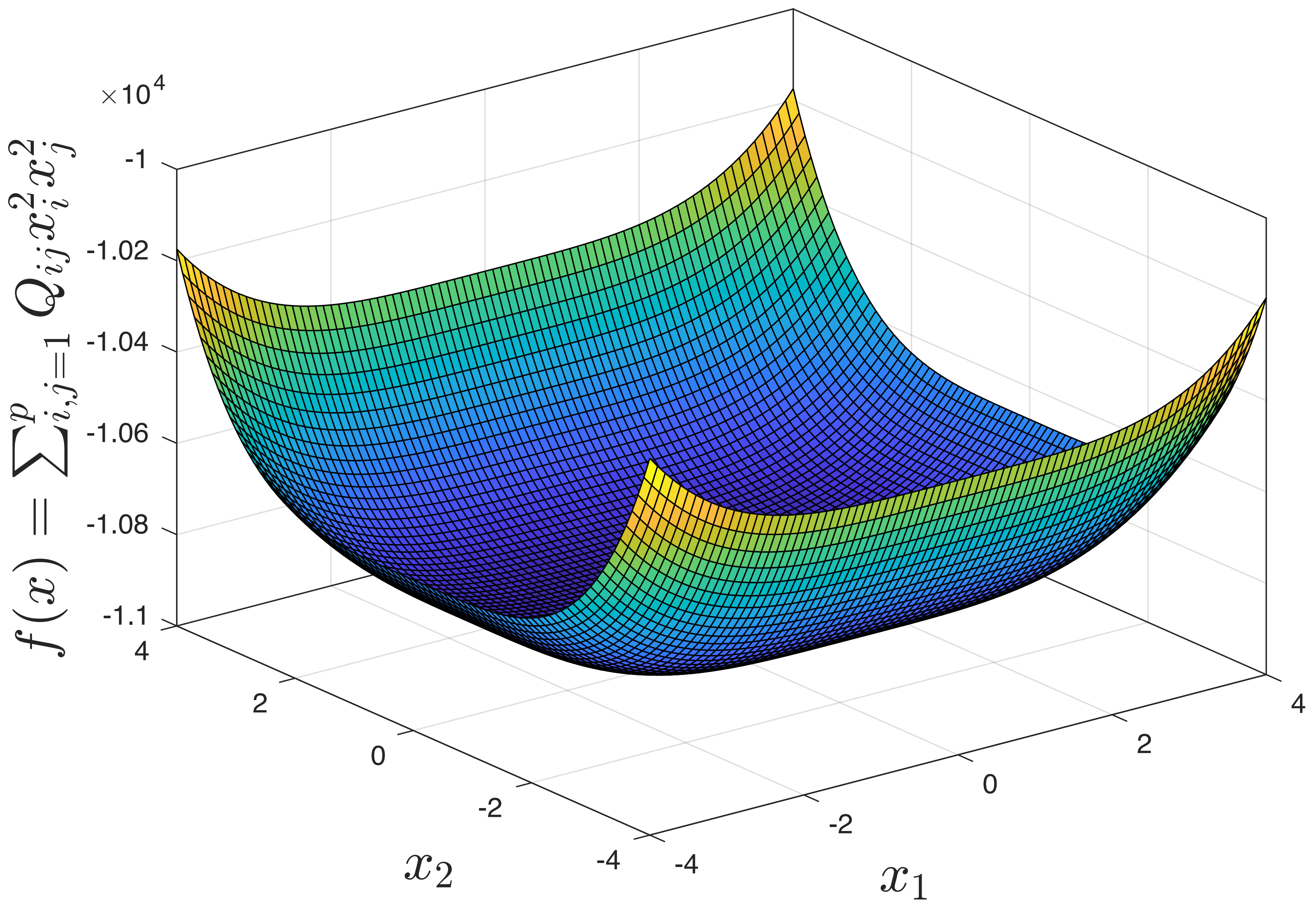
(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– Some observations:

– $p = 2,\ Q \succeq 0$



NP–hardness

- Non–convex continuous optimization = NP–hard in general

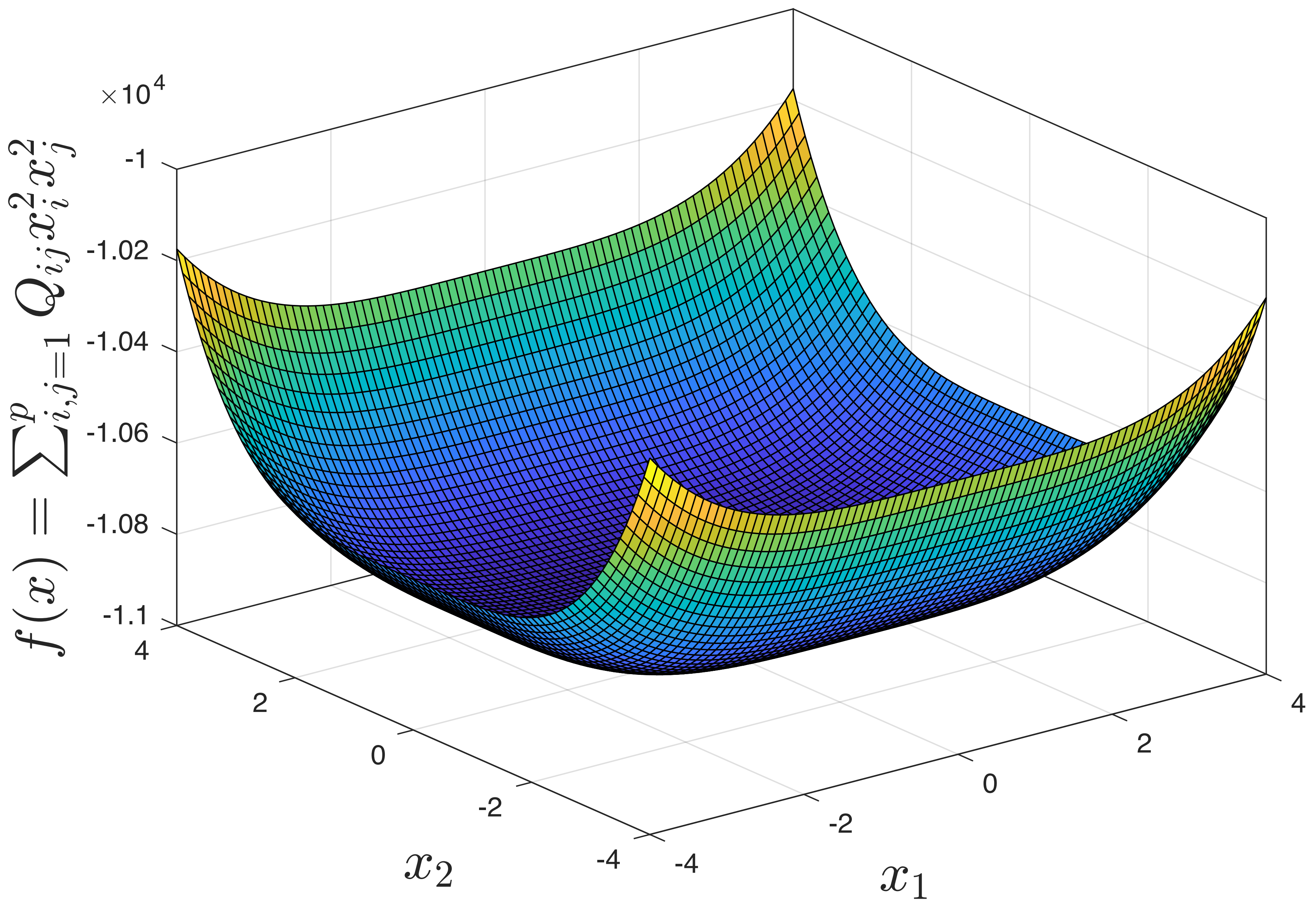
  (Specifically can be polynomially solvable)

- Example: **Homogeneous quartics**

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

- Some observations:
  - if $Q \succeq 0$, then $f(x) \geq 0, \ \forall x$

- $p = 2, \ Q \succeq 0$



NP–hardness

- Non–convex continuous optimization = NP–hard in general

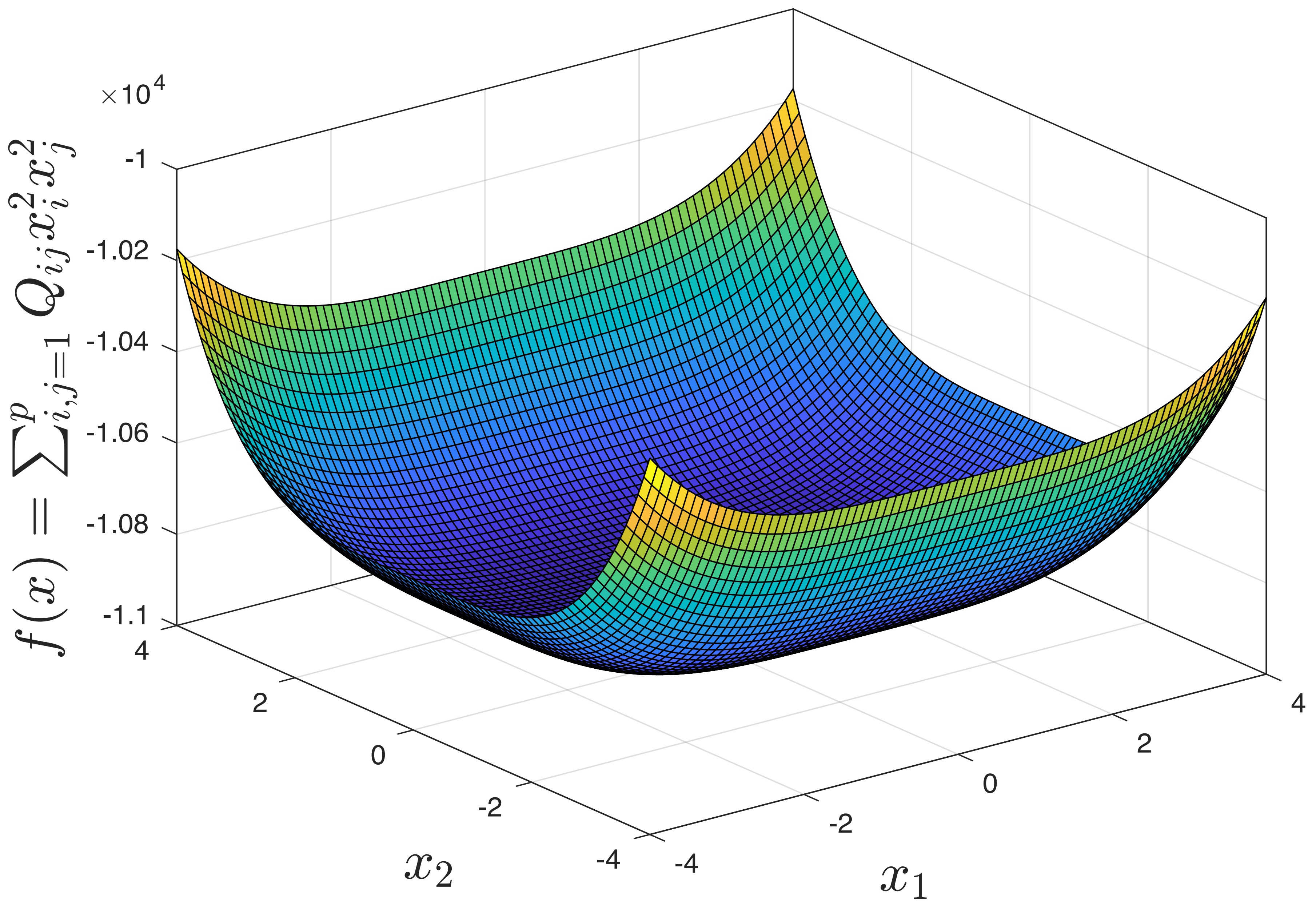
(Specifically can be polynomially solvable)

- Example: **Homogeneous quartics**

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

- Some observations:
  - if $Q \succeq 0$, then $f(x) \geq 0, \; \forall x$
  - Thus, $x = 0$ is global min.

- $p = 2, \; Q \succeq 0$



NP–hardness

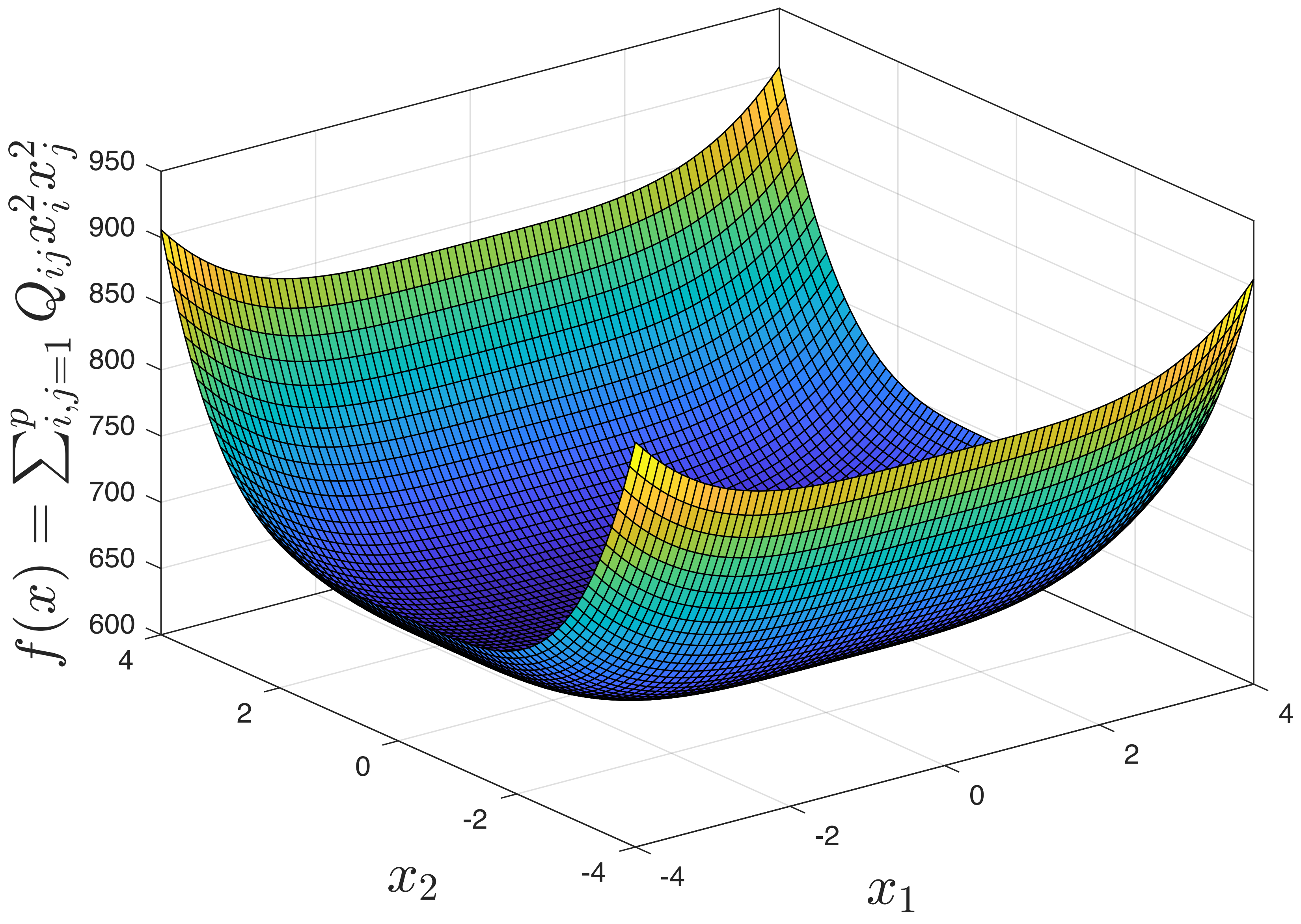
– Non–convex continuous optimization = NP–hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– $p = 2,\ Q \geq_{ij} 0$



NP–hardness

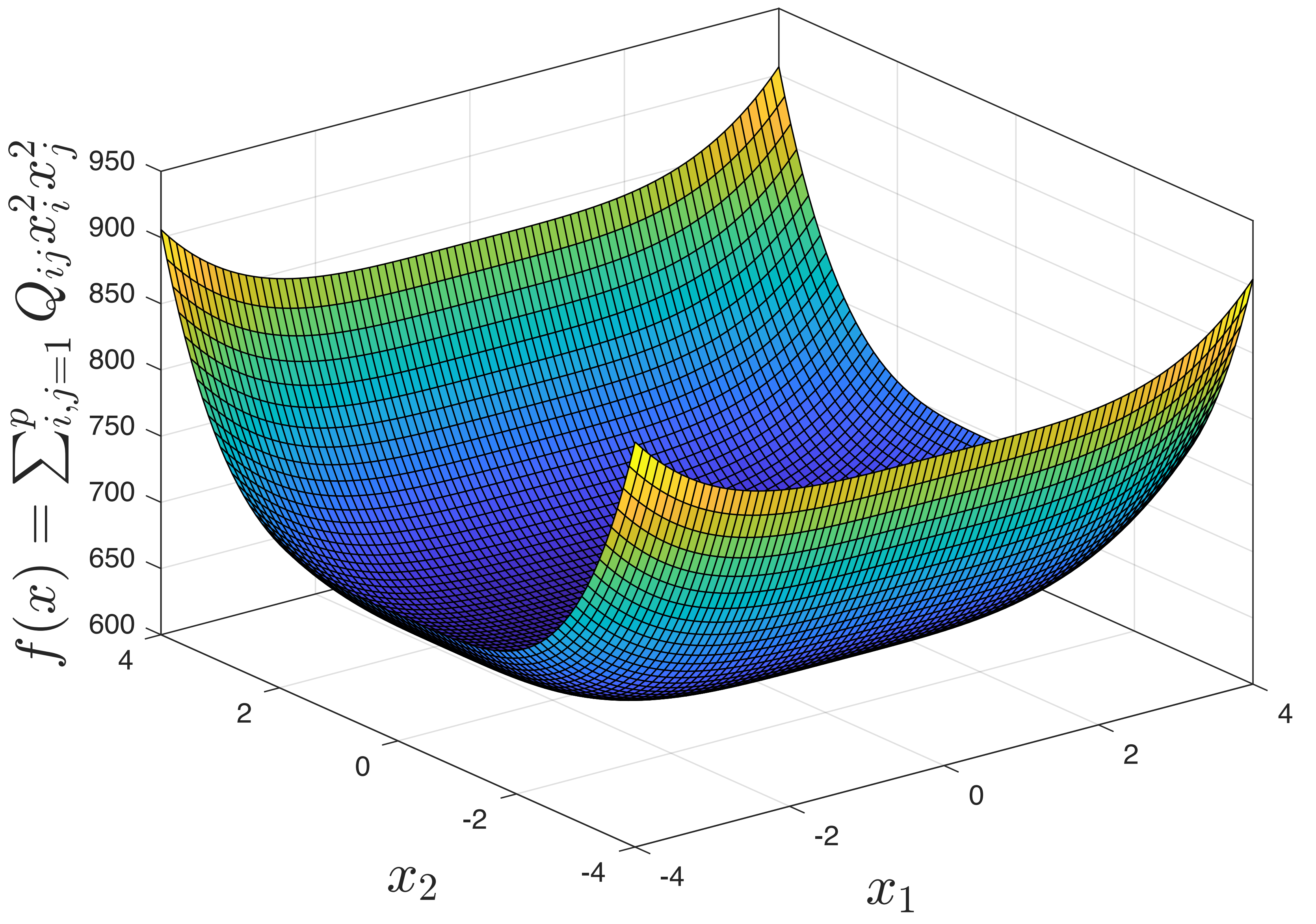
- Non–convex continuous optimization = NP–hard in general

(Specifically can be polynomially solvable)

- Example: **Homogeneous quartics**

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

- $p = 2, \; Q \geq_{ij} 0$



NP–hardness

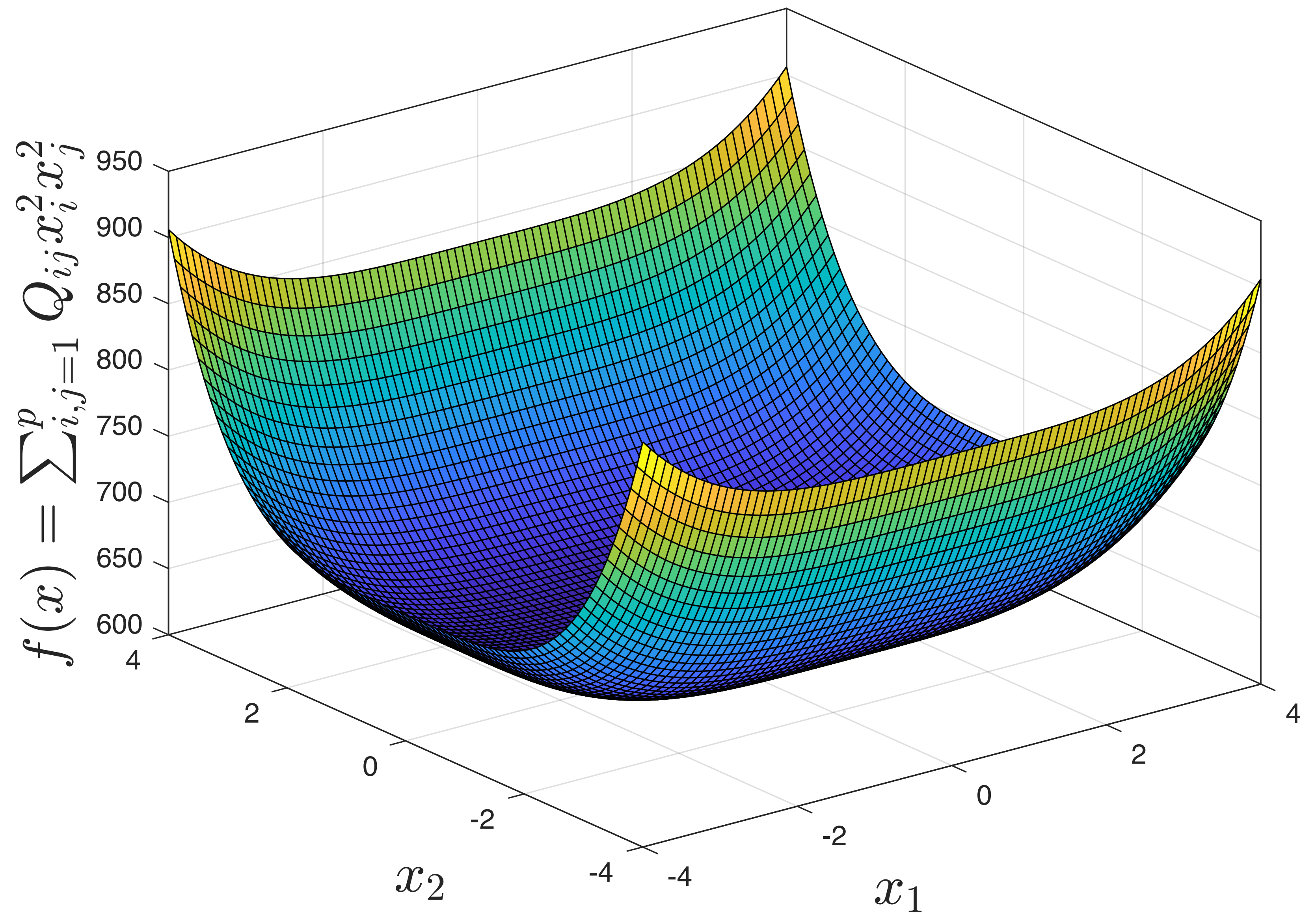
– Non–convex continuous optimization = NP–hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– $p = 2,\ Q \geq_{ij} 0$



NP−hardness

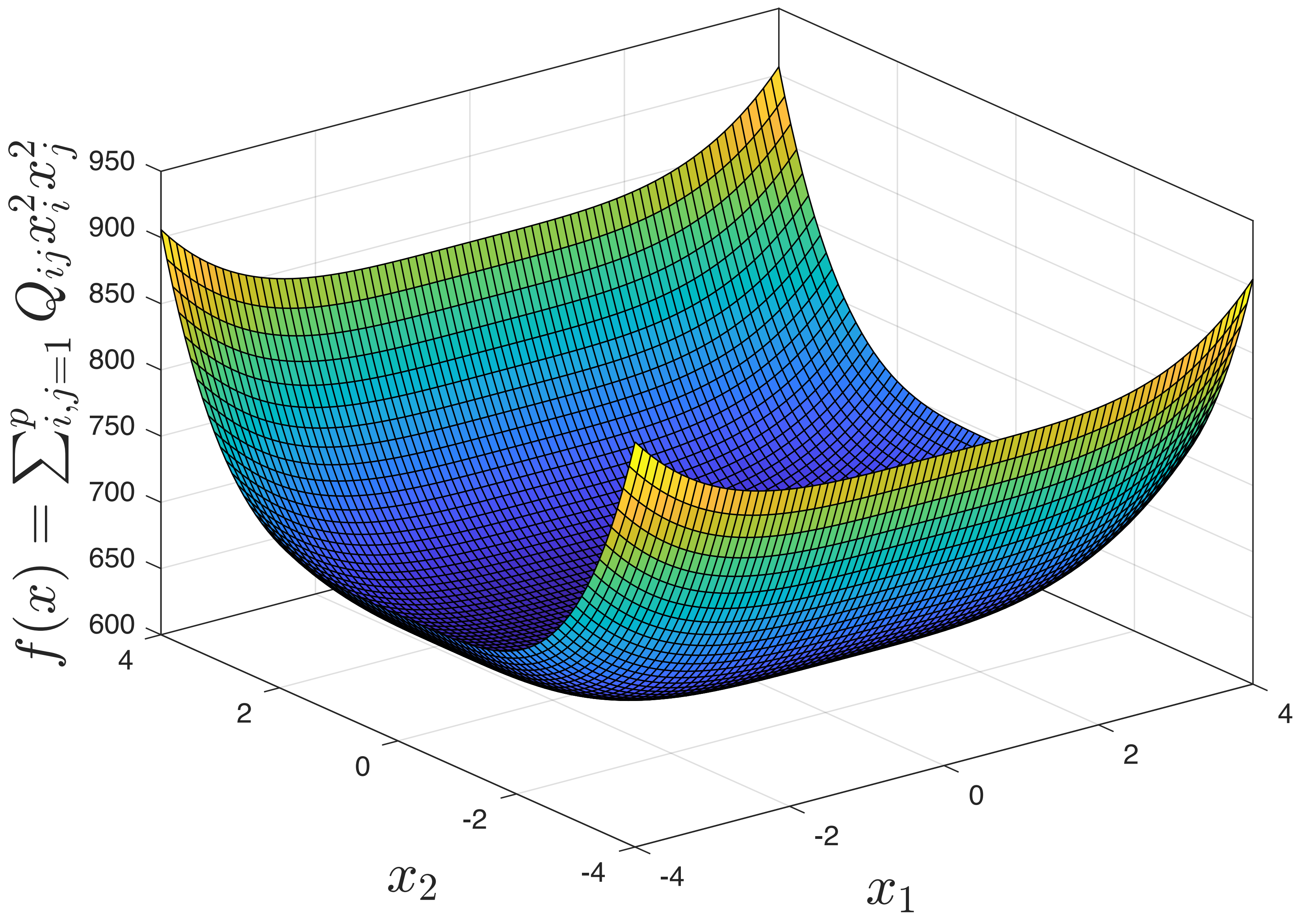
– Non−convex continuous optimization = NP−hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– $p = 2,\ Q \geq_{ij} 0$



– Same observations apply

NP−hardness

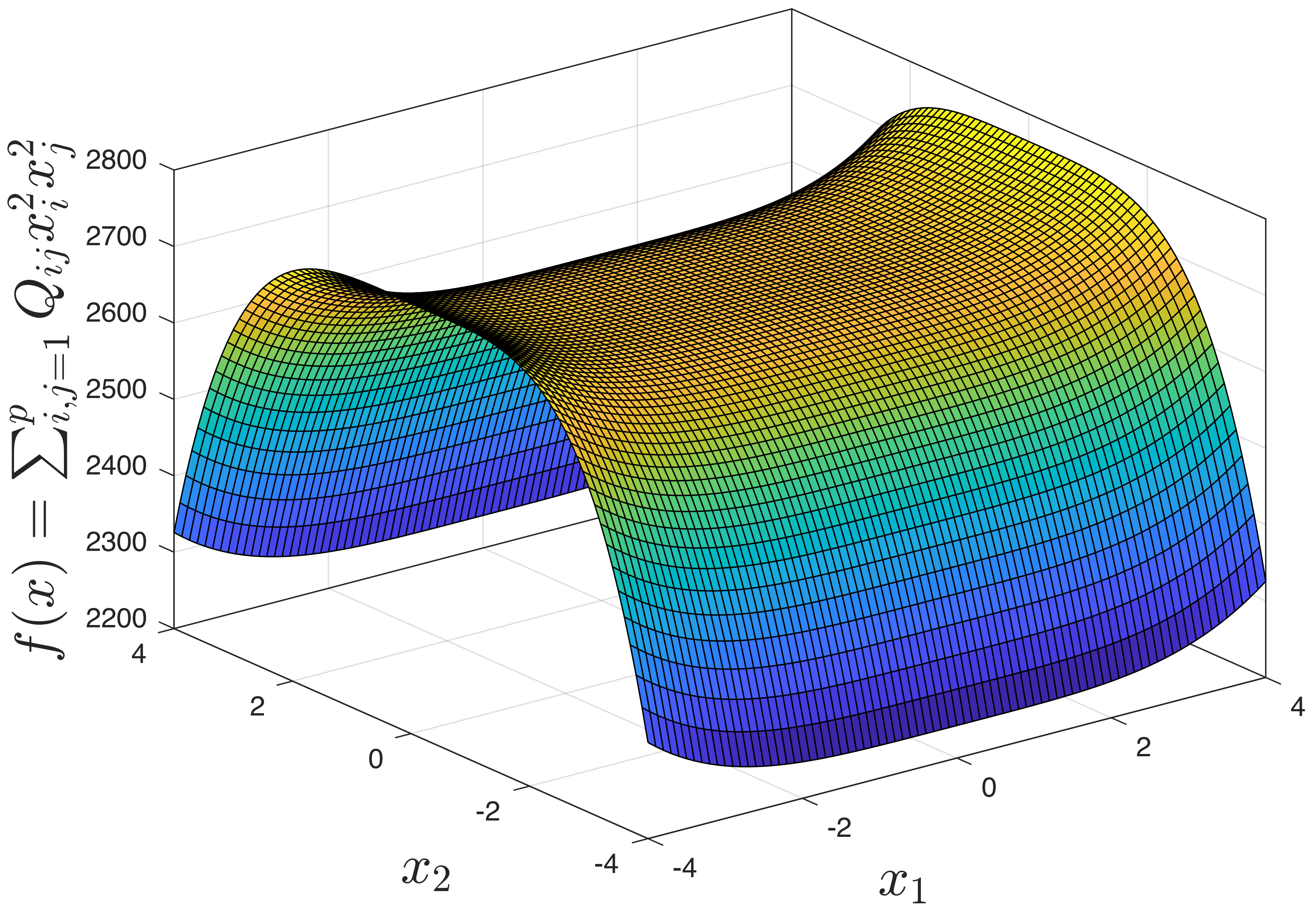
– Non−convex continuous optimization = NP−hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– $p = 2$, Q arbitrary



NP–hardness

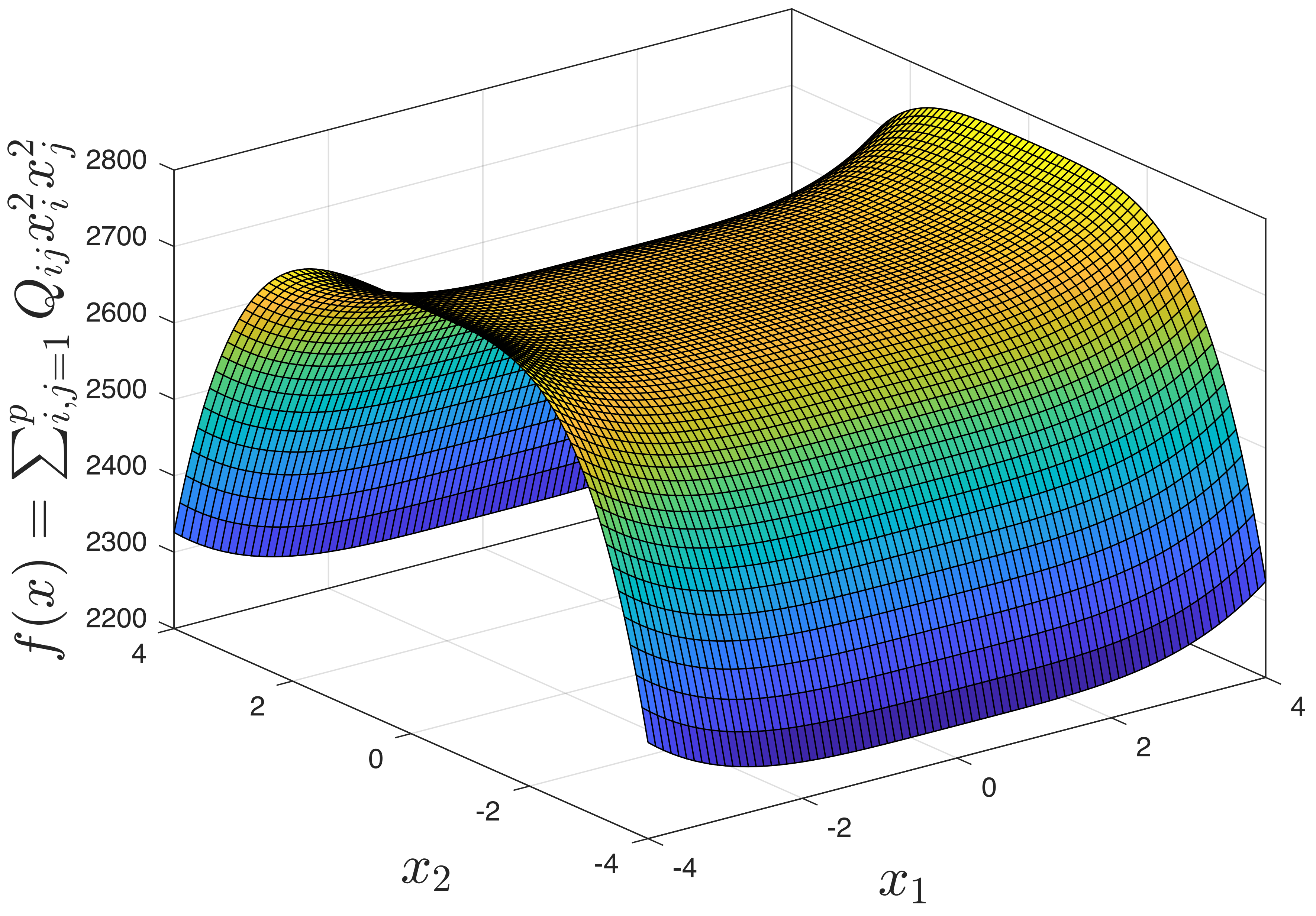
– Non–convex continuous optimization = NP–hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– $p = 2$, Q arbitrary



NP−hardness

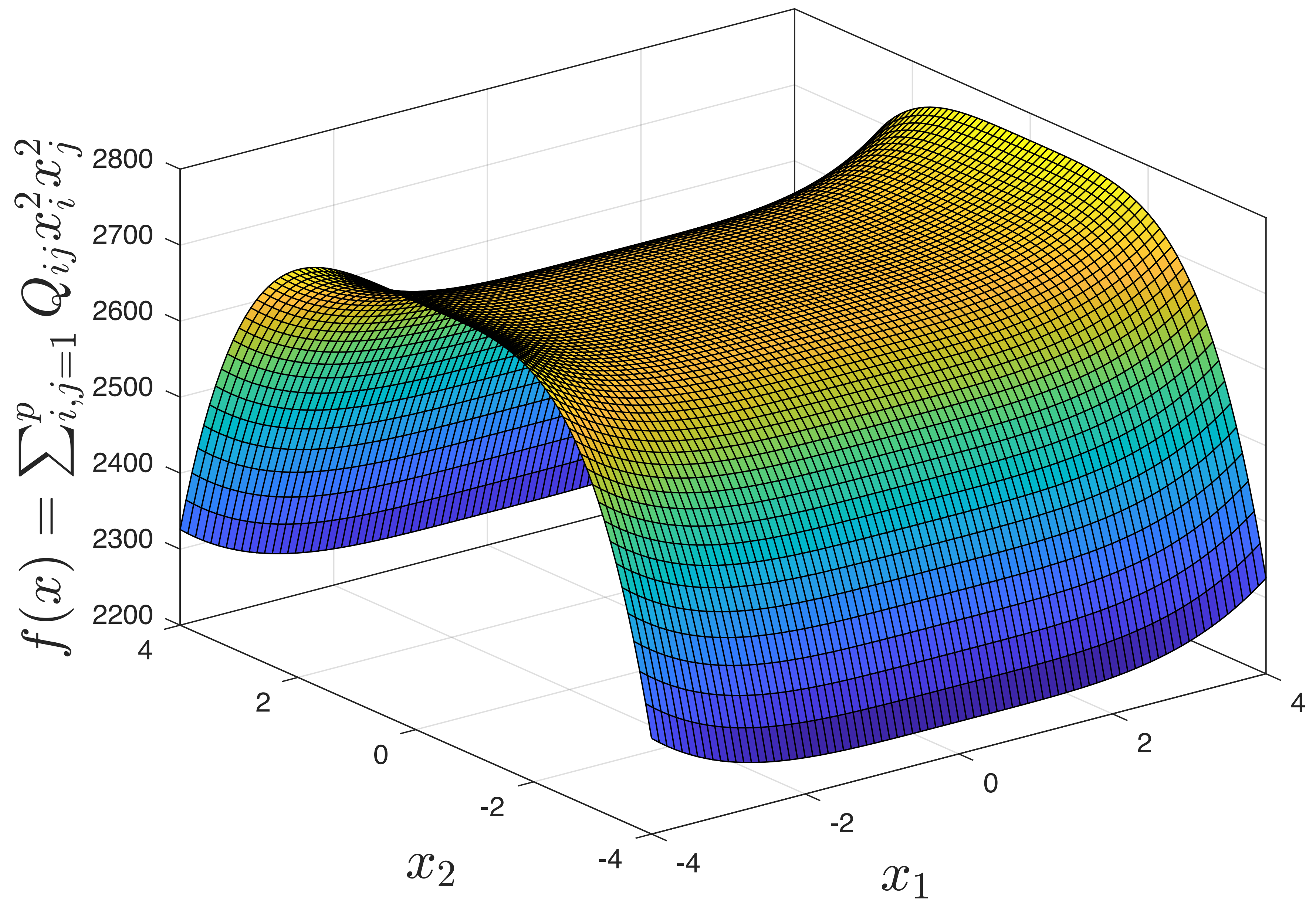
– Non−convex continuous optimization = NP−hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– $p = 2$, Q arbitrary



NP−hardness

– Non−convex continuous optimization = NP−hard in general

(Specifically can be polynomially solvable)

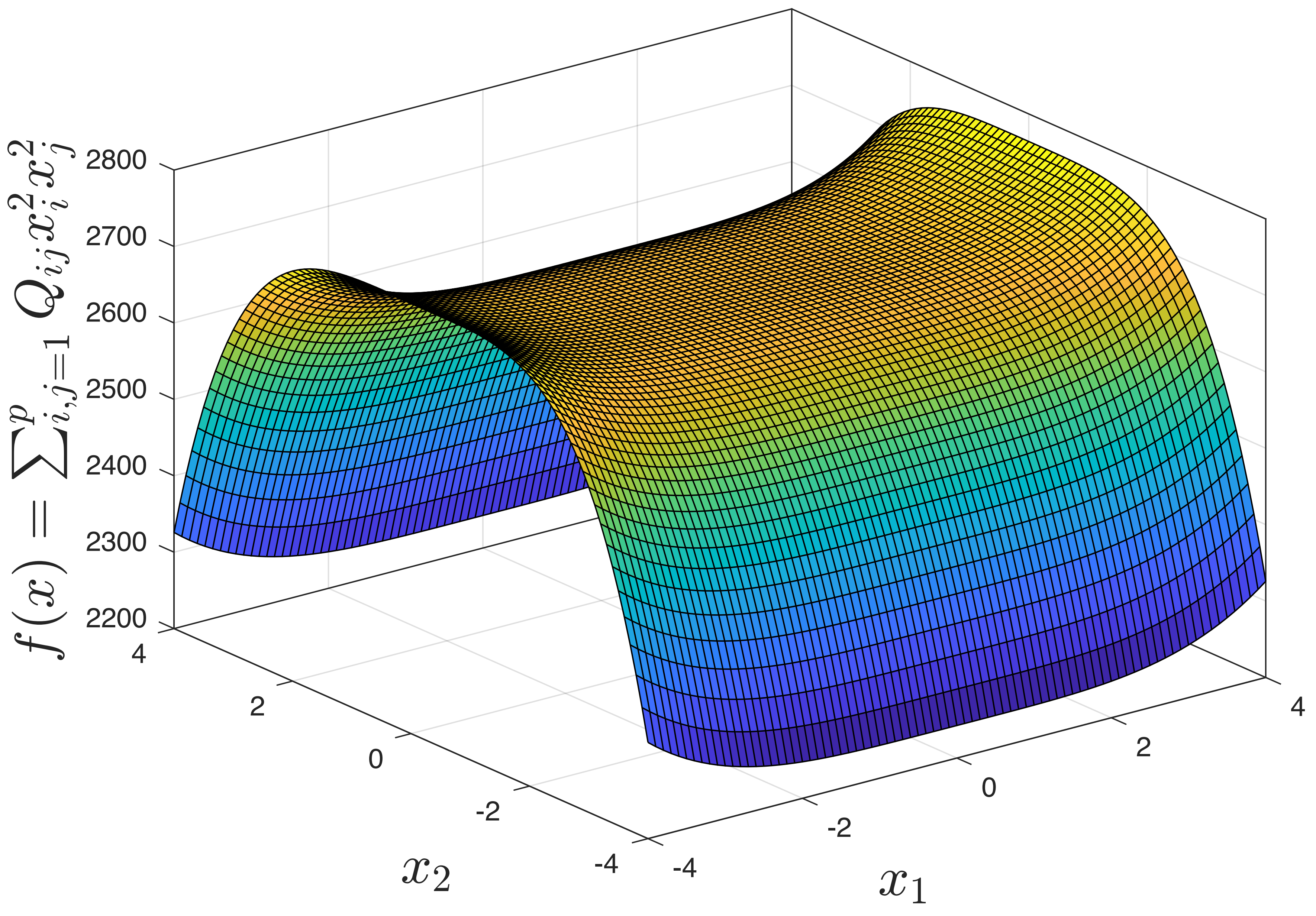
– Example: **Homogeneous quartics**

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

– Some observations:

  – Gradient at zero is zero

– $p = 2$, Q arbitrary



NP−hardness

- Non−convex continuous optimization = NP−hard in general

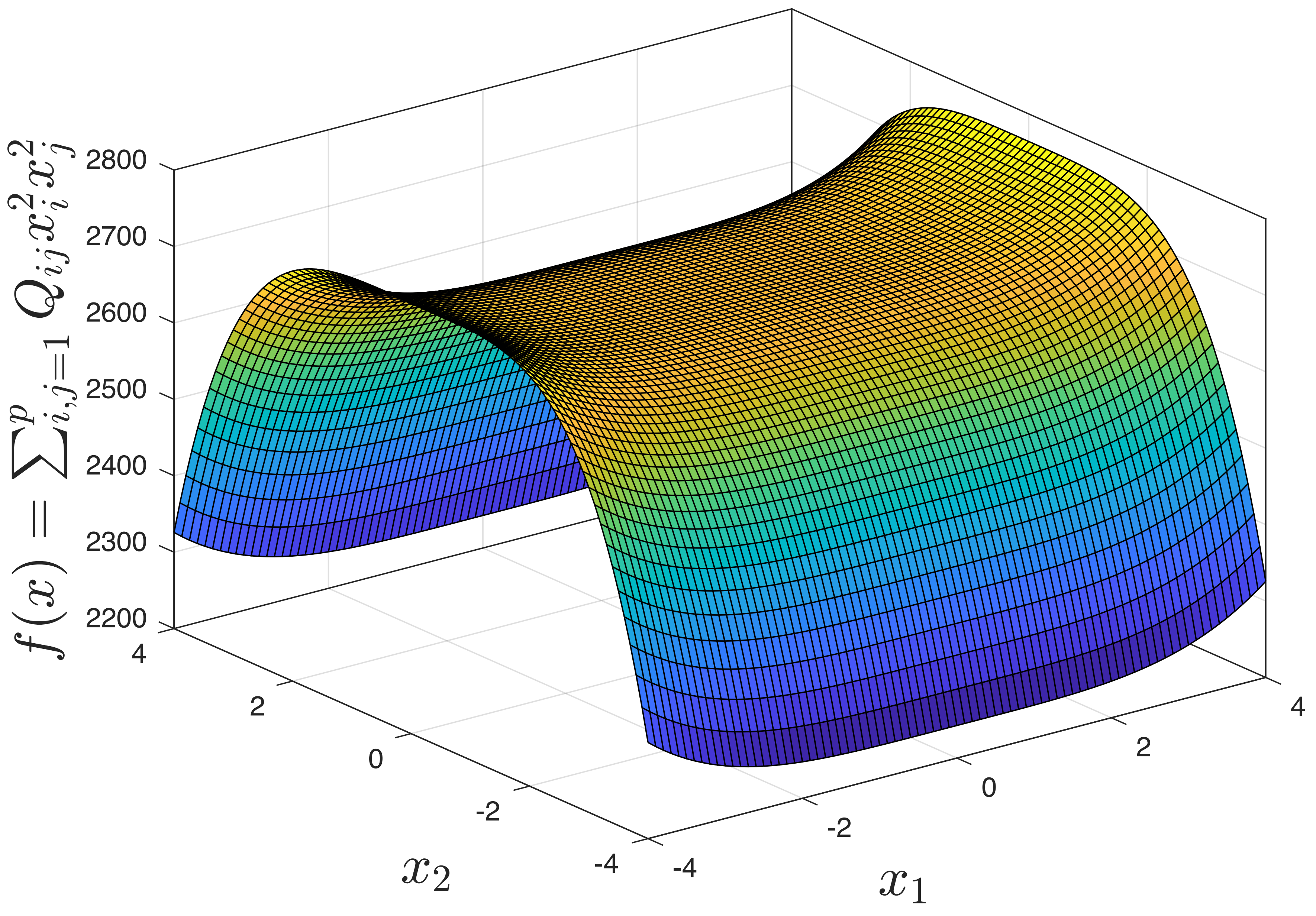
(Specifically can be polynomially solvable)

- Example: **Homogeneous quartics**

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

- Some observations:
  - Gradient at zero is zero
  - Thus, zero is a minimum, maximum or saddle point

- $p = 2$, Q arbitrary



NP−hardness

- Non−convex continuous optimization = NP−hard in general

  (Specifically can be polynomially solvable)

- Example: **Homogeneous quartics**

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

- How can we check what holds at zero point?

NP–hardness

– Non–convex continuous optimization = NP–hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– How can we check what holds at zero point?

(0 is not global minimizer if there is a point that leads to negative objective)

NP–hardness

- Non–convex continuous optimization = NP–hard in general

  (Specifically can be polynomially solvable)

- Example: **Homogeneous quartics**

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

- How can we check what holds at zero point?

  (0 is not global minimizer if there is a point that leads to negative objective)

- Change of variables: $u_i = x_i^2 \rightarrow f(u) = u^\top Q u$

NP−hardness

- Non−convex continuous optimization = NP−hard in general

  (Specifically can be polynomially solvable)

- Example: **Homogeneous quartics**

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

- How can we check what holds at zero point?

  (0 is not global minimizer if there is a point that leads to negative objective)

- Change of variables: $u_i = x_i^2 \rightarrow f(u) = u^\top Q u$
- 0 is not a global minimizer if there exists non−negative u such that $u^\top Q u < 0$

NP−hardness

- Non−convex continuous optimization = NP−hard in general

(Specifically can be polynomially solvable)

- Example: **Homogeneous quartics**

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

- How can we check what holds at zero point?

(0 is not global minimizer if there is a point that leads to negative objective)

- Change of variables: $u_i = x_i^2 \to f(u) = u^\top Q u$
- 0 is not a global minimizer if there exists non−negative u such that $u^\top Q u < 0$
- This is equivalent to checking if Q is not co−positive: NP−hard!

(equivalent to finding plant cliques in graphs)

NP−hardness

– Non−convex continuous optimization = NP−hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– What makes this case difficult? Let's compute the Hessian at zero:

NP–hardness

– Non–convex continuous optimization = NP–hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– What makes this case difficult? Let's compute the Hessian at zero:

$$\nabla^2 f(0) = 0_{p \times p}$$

(Hessian provides no information about the curvature)

NP–hardness

– Non–convex continuous optimization = NP–hard in general

(Specifically can be polynomially solvable)

– Example: **Homogeneous quartics**

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

– What makes this case difficult? Let's compute the Hessian at zero:

$$\nabla^2 f(0) = 0_{p \times p}$$

(Hessian provides no information about the curvature)

– Not the only example: QCQP, matrix completion/matrix sensing, etc.
tensor (matrix) decompositions

Flash back: GD and types of critical points

(also called stationary points)

Flash back: GD and types of critical points

(also called stationary points)

- Gradient descent for generic smooth functions:

$$x_{t+1} = x_t - \frac{1}{L}\nabla f(x_t)$$

Flash back: GD and types of critical points

(also called stationary points)

- Gradient descent for generic smooth functions:

$$x_{t+1} = x_t - \frac{1}{L}\nabla f(x_t)$$

- Critical point convergence guarantee:

$$f(x_{t+1}) \leq f(x_t) - \frac{1}{2L}\|\nabla f(x_t)\|_2^2$$

As t increases, the objective function decreases,
Until the point where the gradient has close to zero energy

Flash back: GD and types of critical points

(also called stationary points)

- Gradient descent for generic smooth functions:

$$x_{t+1} = x_t - \frac{1}{L}\nabla f(x_t)$$

- Critical point convergence guarantee:

$$f(x_{t+1}) \leq f(x_t) - \frac{1}{2L}\|\nabla f(x_t)\|_2^2$$

As t increases, the objective function decreases,
Until the point where the gradient has close to zero energy

- No guarantees on the type of critical point we converge to

Flash back: GD and types of critical points

(also called stationary points)

Flash back: GD and types of critical points

(also called stationary points)

- Global minima/global maxima:

Flash back: GD and types of critical points

(also called stationary points)

- Global minima/global maxima:

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and $f(x^\star) \leq f(x), \ \forall x \quad (f(x^\star) \geq f(x), \ \forall x)$

Flash back: GD and types of critical points

(also called stationary points)

- Global minima/global maxima:

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and $f(x^\star) \leq f(x), \ \forall x \quad (f(x^\star) \geq f(x), \ \forall x)$

- Local minima/local maxima:

Flash back: GD and types of critical points

(also called stationary points)

– Global minima/global maxima:

$$\nabla f(x^\star) = 0 \quad \text{and all directions go upwards (min.) or downwards (max.)}$$

$$\text{and} \qquad f(x^\star) \leq f(x), \ \forall x \quad (f(x^\star) \geq f(x), \ \forall x)$$

– Local minima/local maxima:

$$\nabla f(x^\star) = 0 \quad \text{and all directions go upwards (min.) or downwards (max.)}$$

$$\text{and maybe} \quad \exists x \text{ such that } f(x^\star) \geq f(x), \quad (f(x^\star) \leq f(x))$$

Flash back: GD and types of critical points

(also called stationary points)

– Global minima/global maxima:

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and $f(x^\star) \leq f(x), \ \forall x \quad (f(x^\star) \geq f(x), \ \forall x)$

– Local minima/local maxima:

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and maybe $\exists x$ such that $f(x^\star) \geq f(x), \quad (f(x^\star) \leq f(x))$

– Saddle points:

$\nabla f(x^\star) = 0$ and there are upwards, downwards and/or flat directions

Flash back: GD and types of critical points

(also called stationary points)

- Global minima/global maxima: ← Desirable but not easily attainable!

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and $f(x^\star) \leq f(x), \ \forall x \quad (f(x^\star) \geq f(x), \ \forall x)$

- Local minima/local maxima:

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and maybe $\exists x$ such that $f(x^\star) \geq f(x), \quad (f(x^\star) \leq f(x))$

- Saddle points:

$\nabla f(x^\star) = 0$ and there are upwards, downwards and/or flat directions

Flash back: GD and types of critical points

(also called stationary points)

– Global minima/global maxima:

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and $f(x^\star) \leq f(x), \ \forall x \quad (f(x^\star) \geq f(x), \ \forall x)$

– Local minima/local maxima: ⟵ Often the next best thing

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and maybe $\exists x$ such that $f(x^\star) \geq f(x), \quad (f(x^\star) \leq f(x))$

– Saddle points:

$\nabla f(x^\star) = 0$ and there are upwards, downwards and/or flat directions

What's the deal with local minima?

(and got such publicity?)

What's the deal with local minima?

(and got such publicity?)

– ``The loss surfaces of multilinear networks``, Chromanska et al., 2014

> *``We conjecture that both simulated annealing and SGD converge to the band of low critical points, and that all critical points found there are local minima of high quality measured by the test error. This emphasizes a major difference between large- and small-size networks where for the latter poor quality local minima have non-zero probability of being recovered.``*

(Not the only work on this subject)

What's the deal with local minima?

(and got such publicity?)

- ``The loss surfaces of multilinear networks``, Chromanska et al., 2014

> *``We conjecture that both simulated annealing and SGD converge to the band of low critical points, and that all critical points found there are local minima of high quality measured by the test error. This emphasizes a major difference between large- and small-size networks where for the latter poor quality local minima have non-zero probability of being recovered.``*

(Not the only work on this subject)

- For larger models, **a local minima is "good enough"**, since its loss value is roughly similar.

(We do not know the exact value of the global minima, so we can only conjecture)

What's the deal with local minima?

(and got such publicity?)

- ``The loss surfaces of multilinear networks``, Chromanska et al., 2014

> *``We conjecture that both simulated annealing and SGD converge to the band of low critical points, and that all critical points found there are local minima of high quality measured by the test error. This emphasizes a major difference between large- and small-size networks where for the latter poor quality local minima have non-zero probability of being recovered.``*

(Not the only work on this subject)

- For larger models, **a local minima is "good enough"**, since its loss value is roughly similar.

(We do not know the exact value of the global minima, so we can only conjecture)

- Why would this be true in practice?
  - Different random seeds lead to different models with similar performance

Flash back: GD and types of critical points

(also called stationary points)

- Global minima/global maxima:

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and $f(x^\star) \leq f(x), \; \forall x \quad (f(x^\star) \geq f(x), \; \forall x)$

- Local minima/local maxima:

$\nabla f(x^\star) = 0$ and all directions go upwards (min.) or downwards (max.)

and maybe $\exists x$ such that $f(x^\star) \geq f(x), \quad (f(x^\star) \leq f(x))$

- Saddle points: ← Often stall the convergence to a better point

$\nabla f(x^\star) = 0$ and there are upwards, downwards and/or flat directions

What's the deal with saddle points?

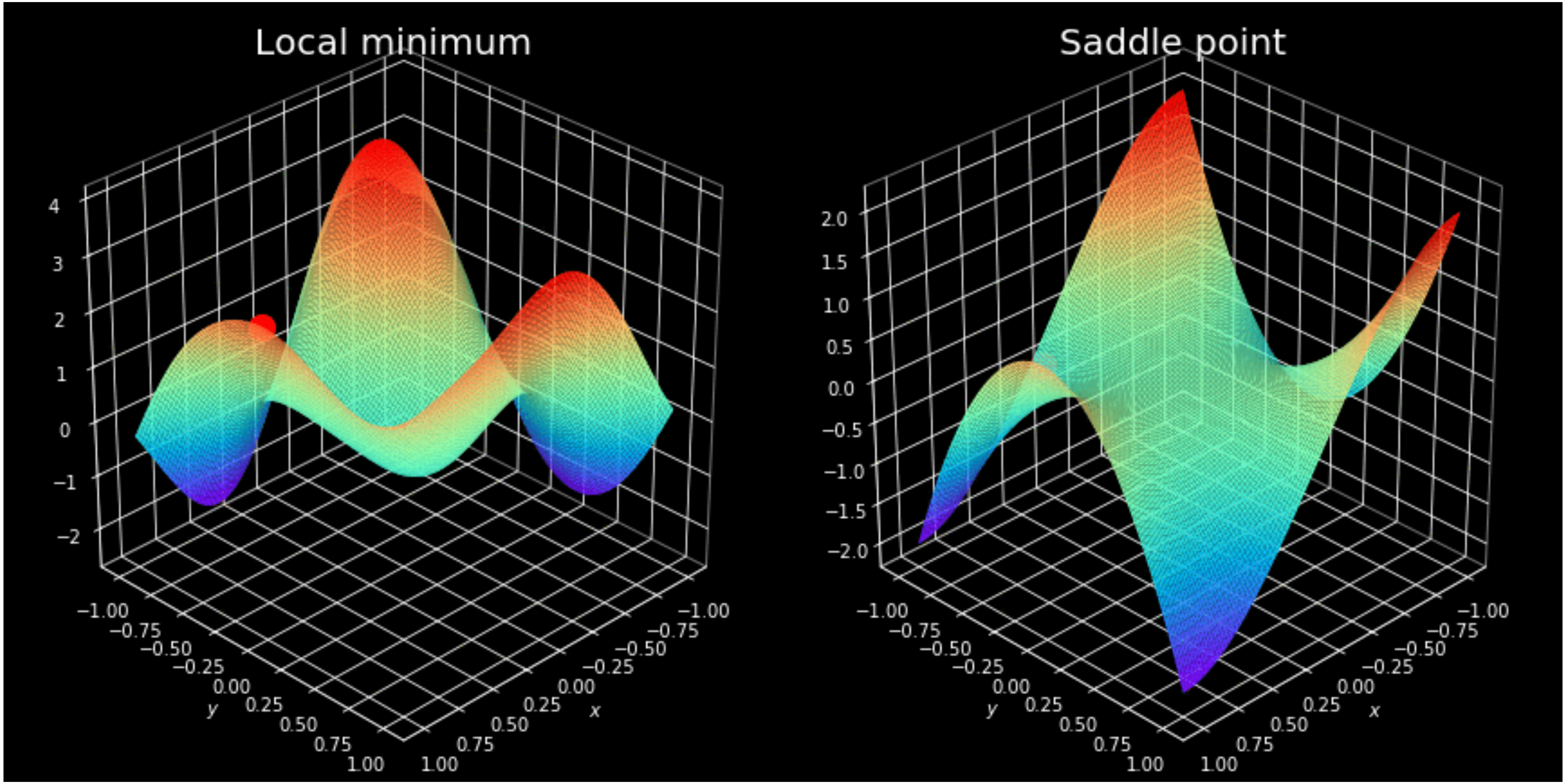
What's the deal with saddle points?

– ``Identifying and attacking the saddle point problem in high–dimensional non–convex optimization.``, Dauphin et al., 2014

> *``A deeper and more profound difficulty originates from the proliferation of saddle points, not local minima, especially in high dimensional problems of practical interest. Such saddle points are surrounded by high error plateaus that can dramatically slow down learning, and give the illusory impression of the existence of a local minimum.``*

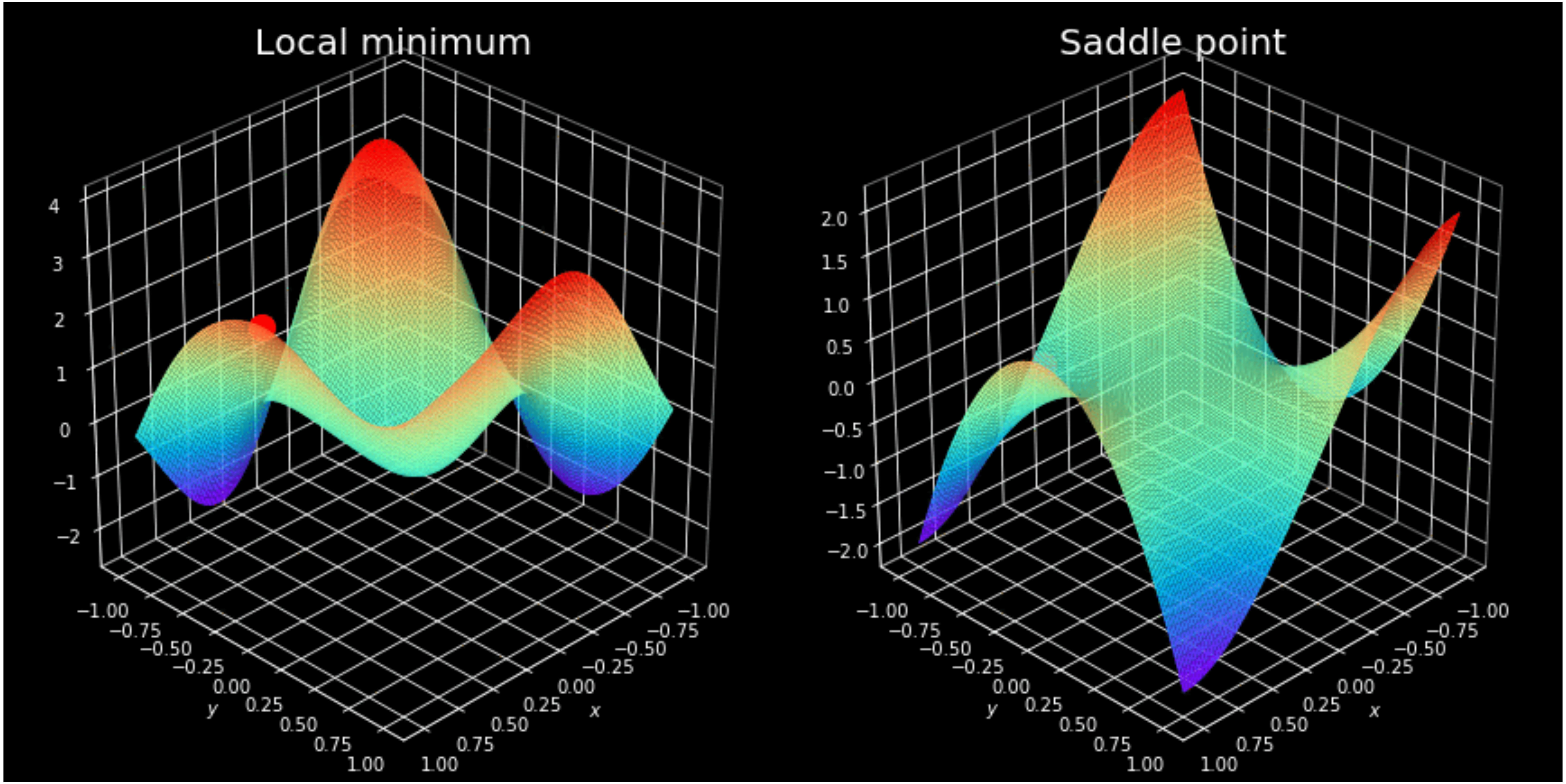
What's the deal with saddle points?

(Tribute: unknown)



What's the deal with saddle points?

(Tribute: unknown)



What's the deal with saddle points?

– ``Identifying and attacking the saddle point problem in high–dimensional non–convex optimization.``, Dauphin et al., 2014

> *``A deeper and more profound difficulty originates from the proliferation of saddle points, not local minima, especially in high dimensional problems of practical interest. Such saddle points are surrounded by high error plateaus that can dramatically slow down learning, and give the illusory impression of the existence of a local minimum.``*

– Saddle points can be large **plateaus/flat regions** or (approximately) regions with very **slow slope.**

What's the deal with saddle points?

- ``Identifying and attacking the saddle point problem in high-dimensional non-convex optimization.``, Dauphin et al., 2014

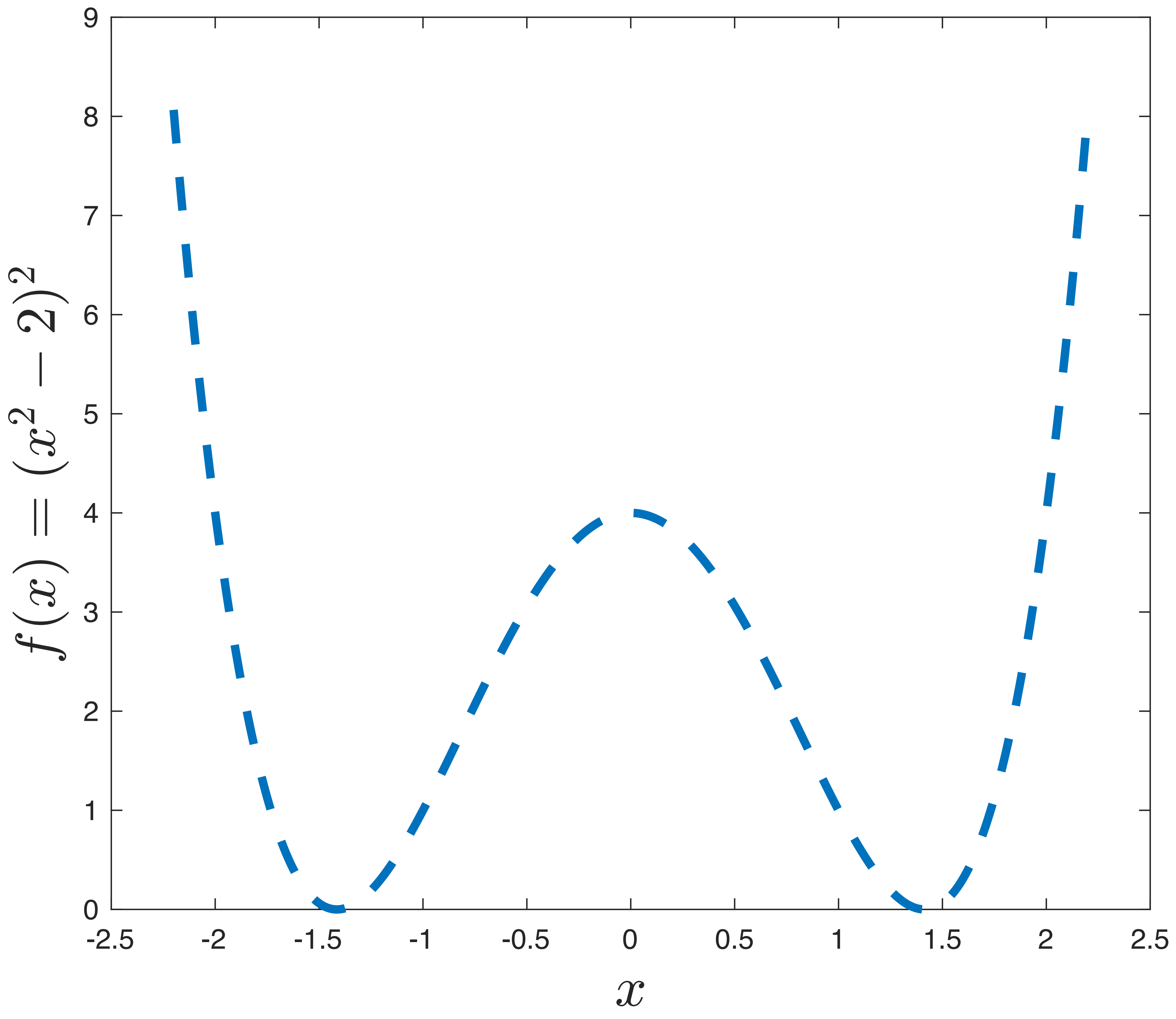
> *``A deeper and more profound difficulty originates from the proliferation of saddle points, not local minima, especially in high dimensional problems of practical interest. Such saddle points are surrounded by high error plateaus that can dramatically slow down learning, and give the illusory impression of the existence of a local minimum.``*

- Saddle points can be large **plateaus/flat regions** or (approximately) regions with very **slow slope.**

- How many saddle points be there?

How many saddle points could be there?

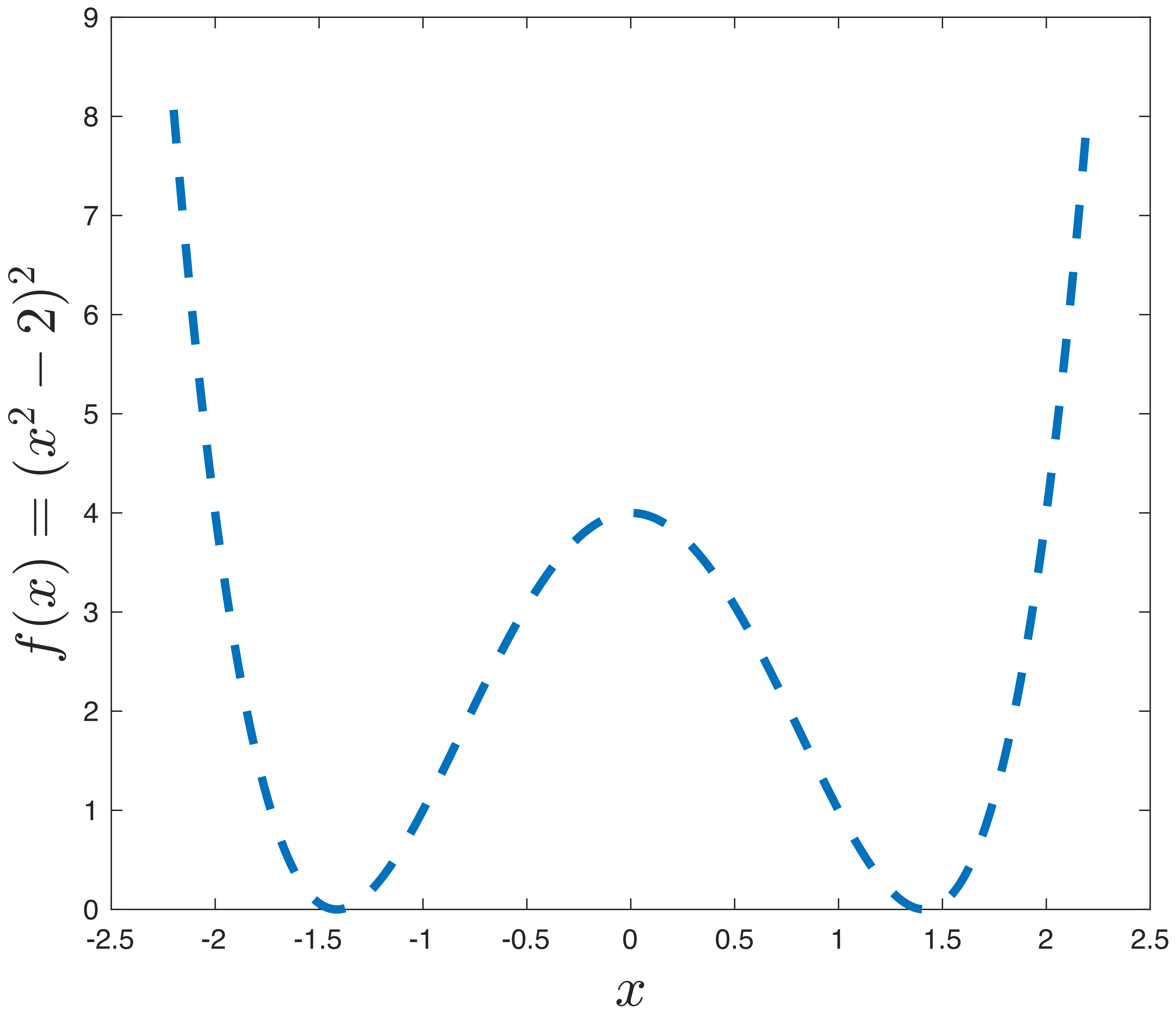
– Toy example #1: $f(x) = (x^2 - 2)^2$



How many saddle points could be there?

- Toy example #1: $f(x) = (x^2 - 2)^2$

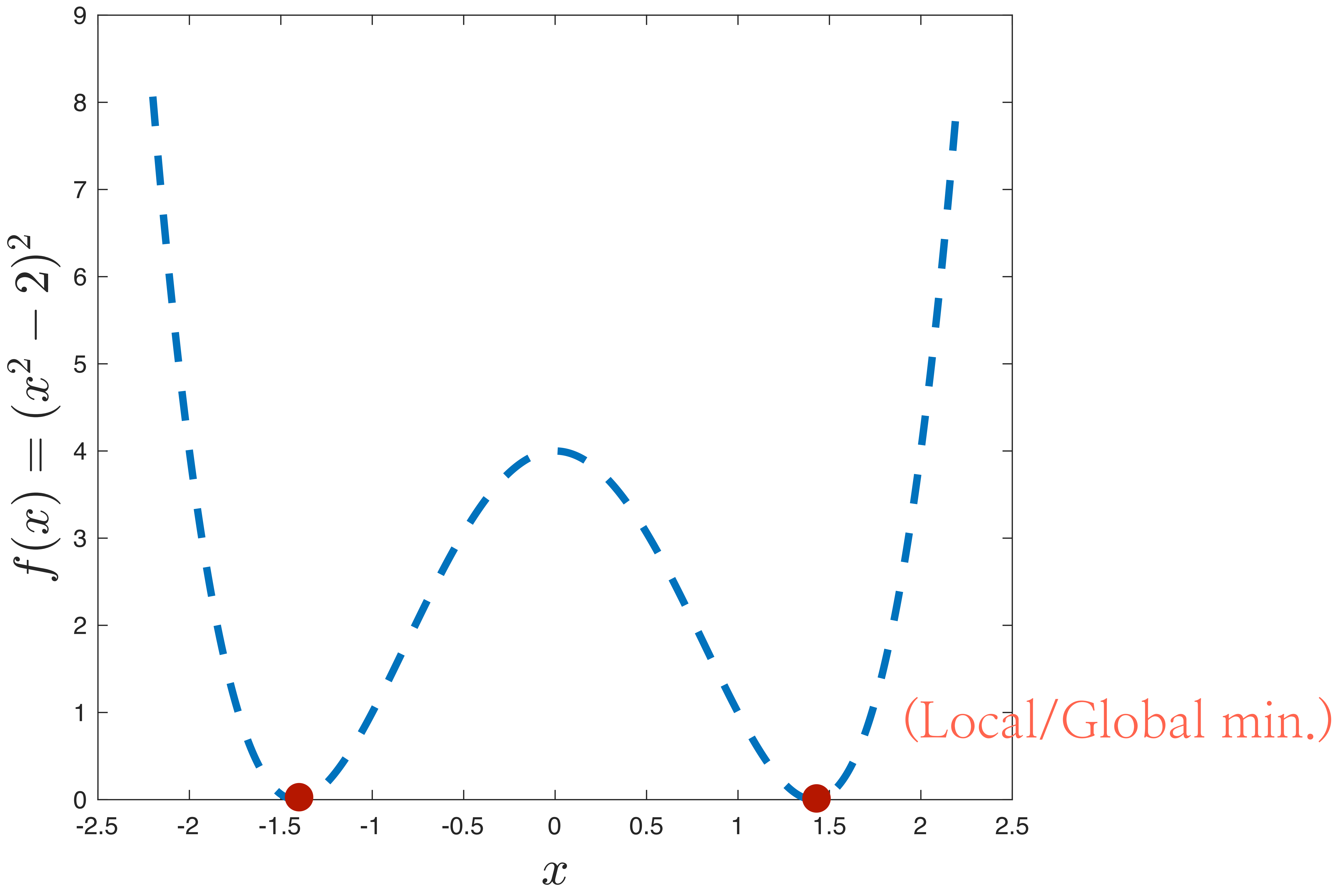
- Find:
  - Global min/max
  - Local min/max
  - Saddle points



How many saddle points could be there?

- Toy example #1: $f(x) = (x^2 - 2)^2$

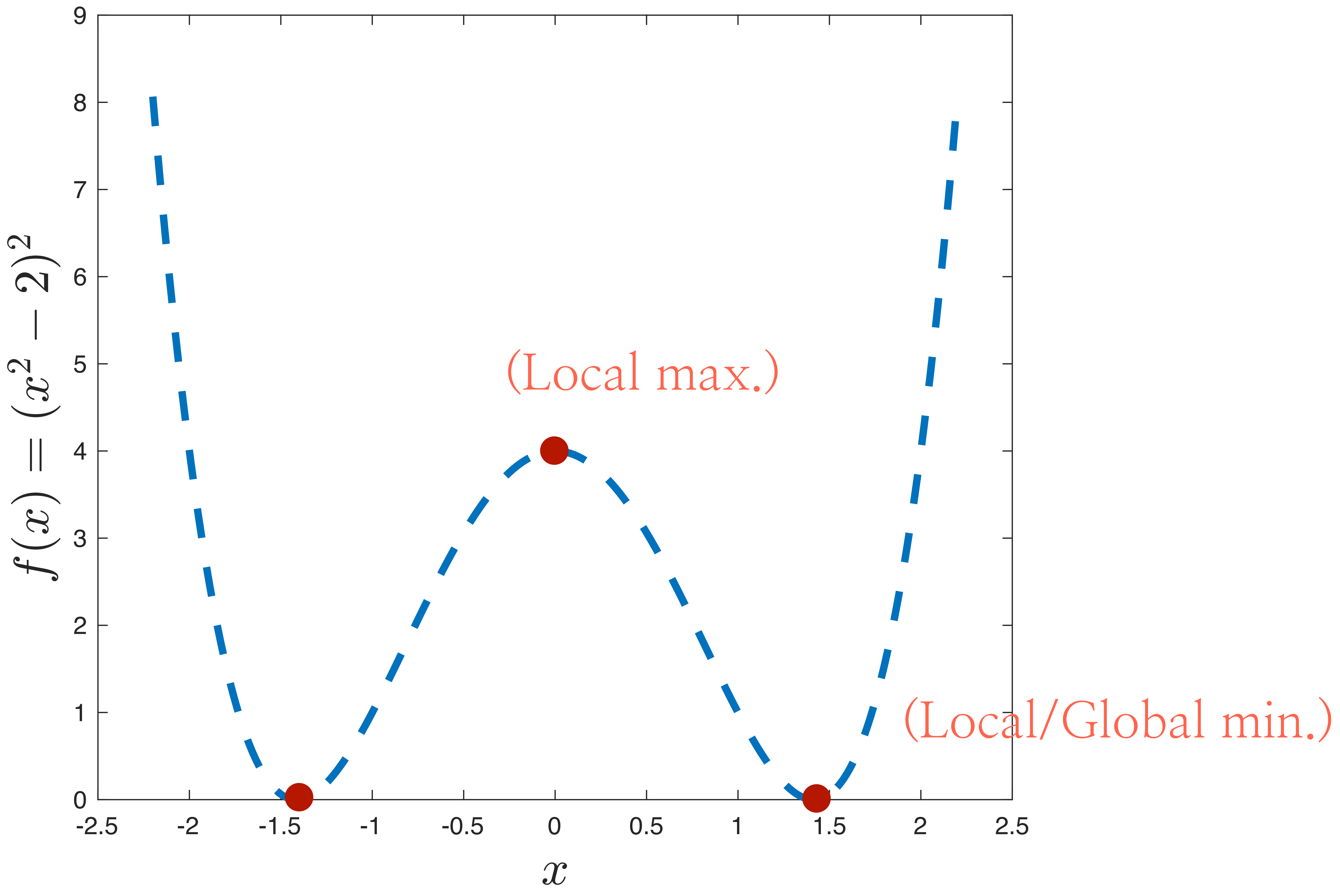
- Find:
  - Global min/max
  - Local min/max
  - Saddle points



How many saddle points could be there?

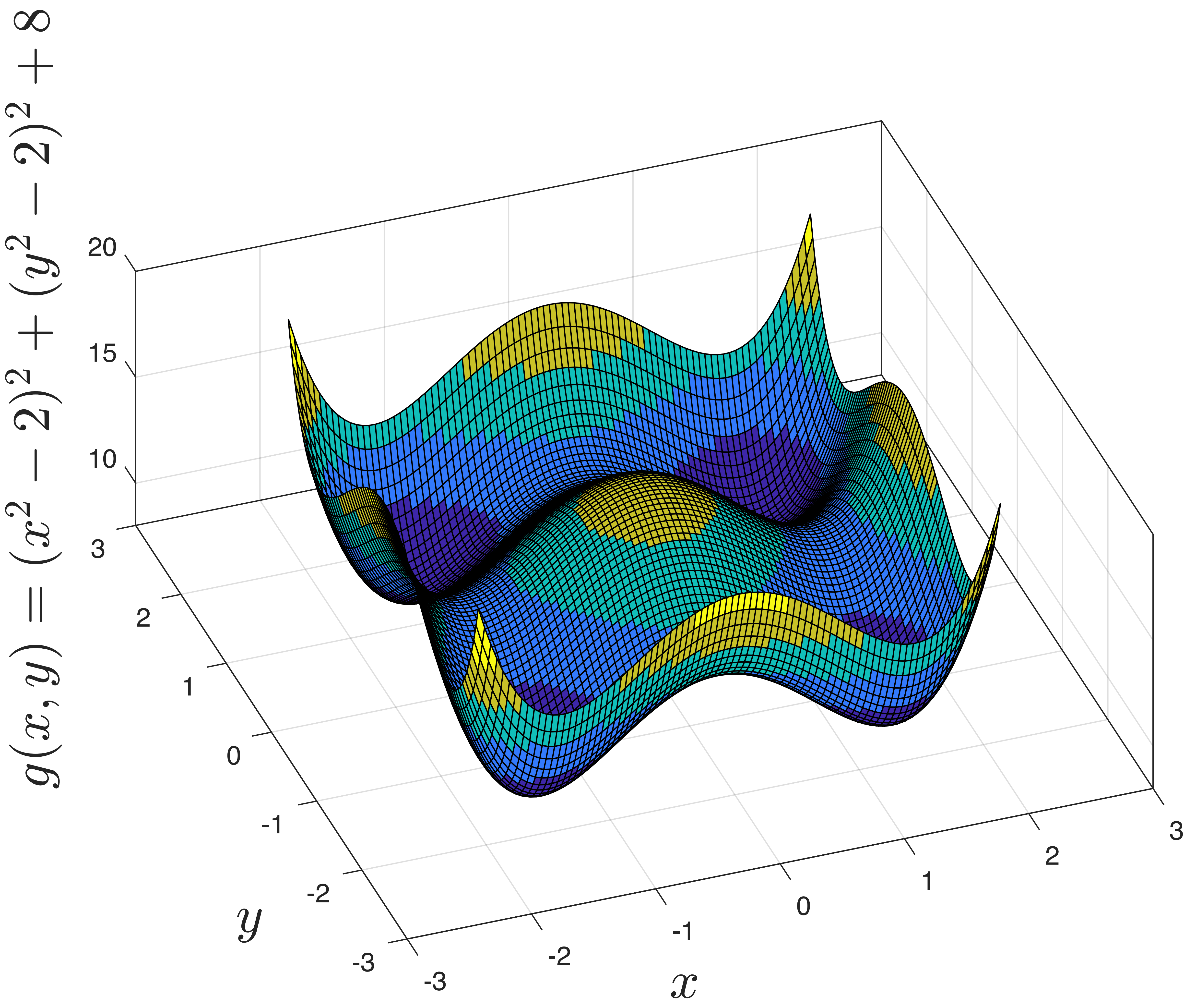
- Toy example #1: $f(x) = (x^2 - 2)^2$

- Find:
  - Global min/max
  - Local min/max
  - Saddle points



How many saddle points could be there?

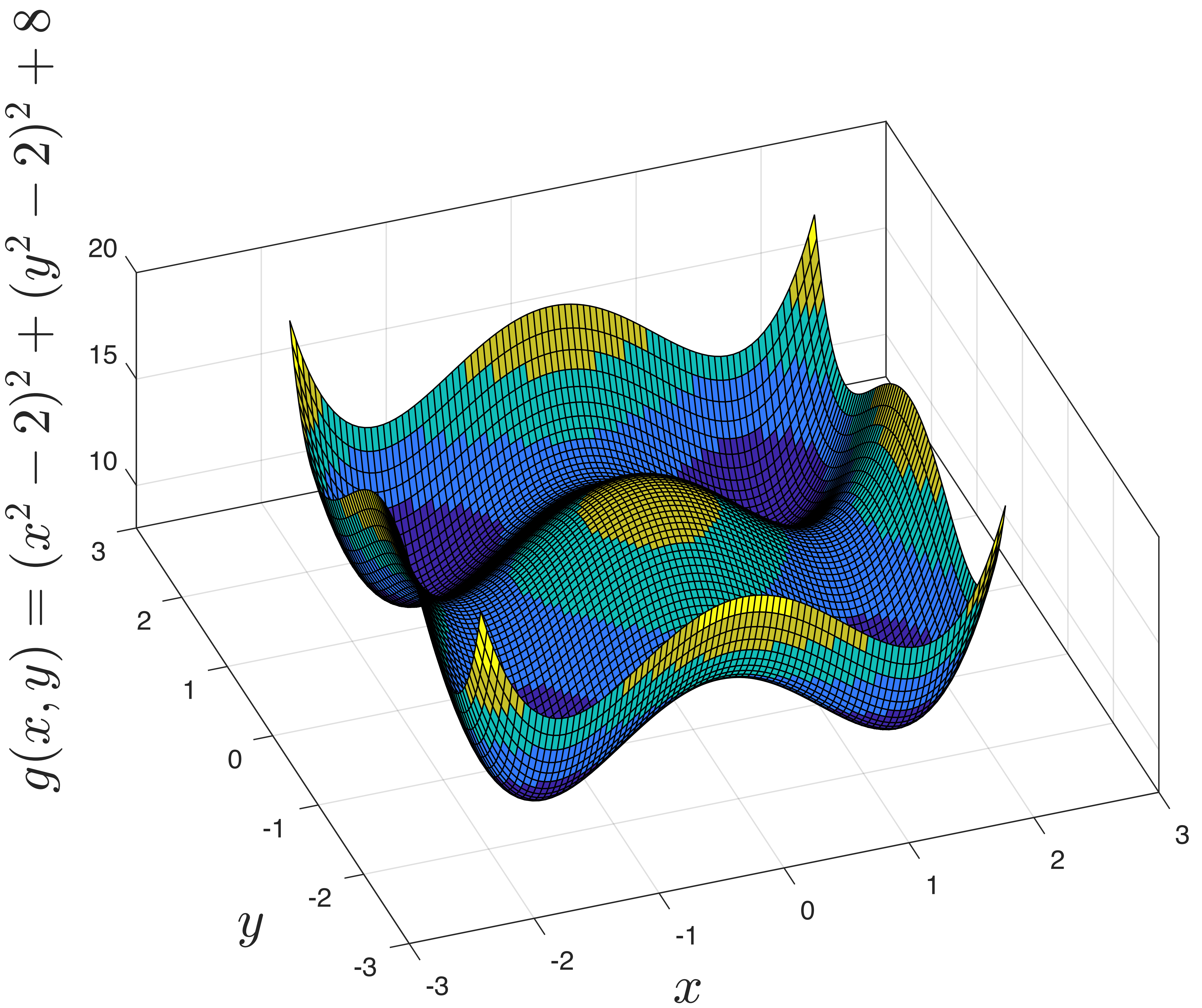
- Toy example #2: $g(x, y) = f(x) + f(y) + 8 = (x^2 - 2)^2 + (y^2 - 2)^2 + 8$



How many saddle points could be there?

- Toy example #2: $g(x,y) = f(x) + f(y) + 8 = (x^2-2)^2 + (y^2-2)^2 + 8$

- Find:
  - Global min/max
  - Local min/max
  - Saddle points



How many saddle points could be there?

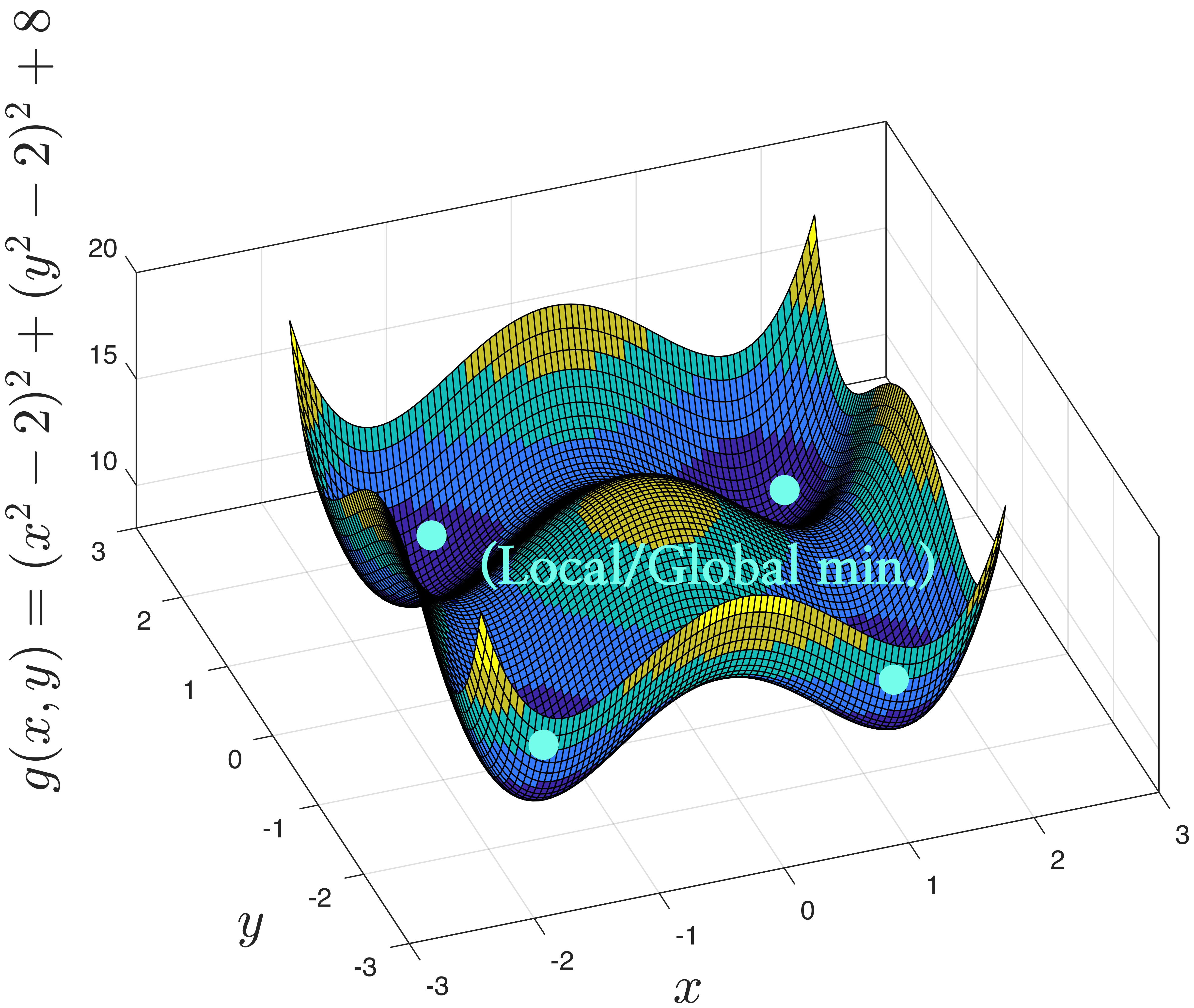
– Toy example #2: $g(x,y) = f(x) + f(y) + 8 = (x^2 - 2)^2 + (y^2 - 2)^2 + 8$

– Find:

– Global min/max

– Local min/max

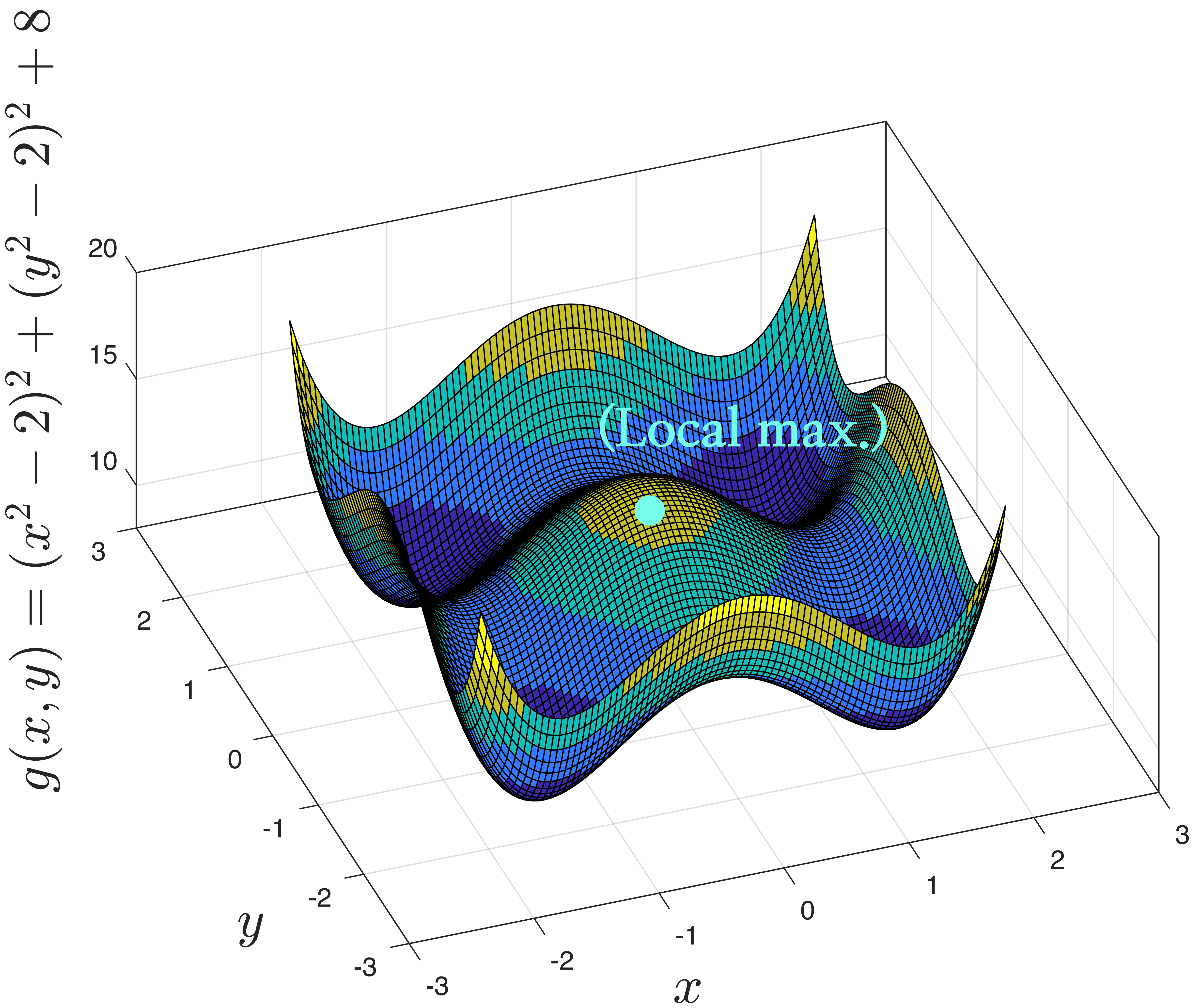
– Saddle points



How many saddle points could be there?

- Toy example #2: $g(x, y) = f(x) + f(y) + 8 = (x^2 - 2)^2 + (y^2 - 2)^2 + 8$

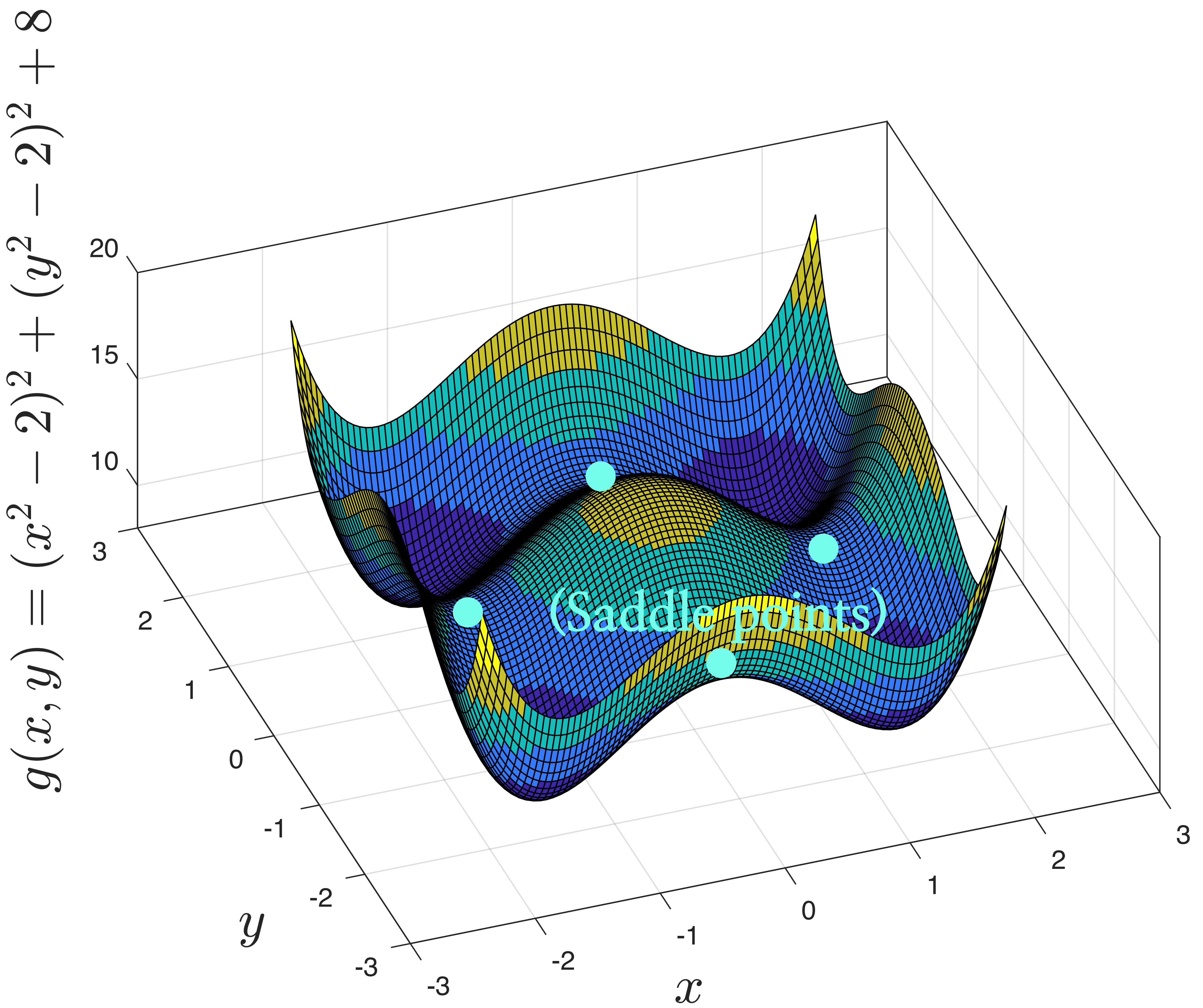
- Find:
  - Global min/max
  - Local min/max
  - Saddle points



How many saddle points could be there?

- Toy example #2: $g(x,y) = f(x) + f(y) + 8 = (x^2-2)^2 + (y^2-2)^2 + 8$

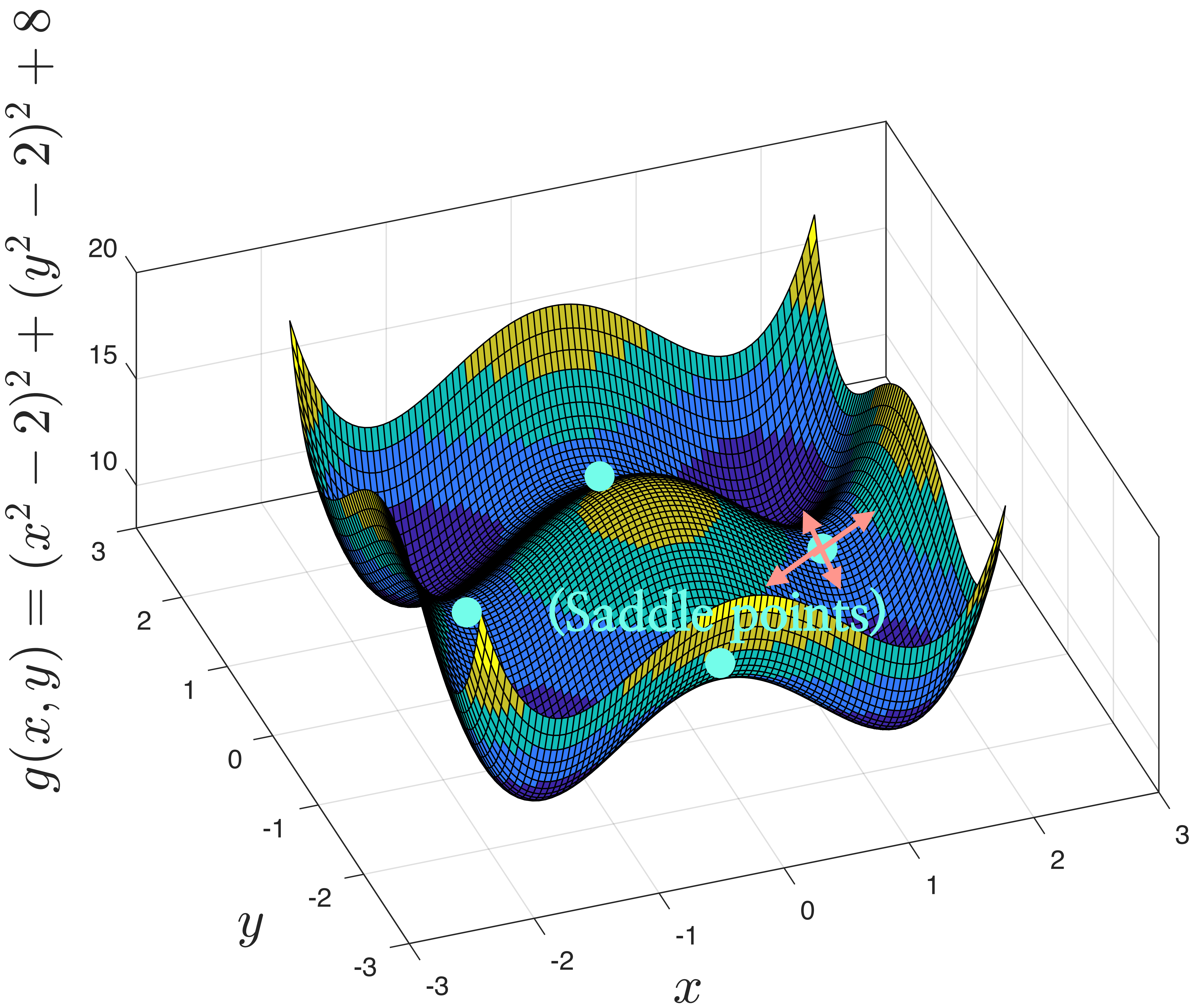
- Find:
  - Global min/max
  - Local min/max
  - Saddle points



How many saddle points could be there?

- Toy example #2: $g(x,y) = f(x) + f(y) + 8 = (x^2-2)^2 + (y^2-2)^2 + 8$

- Find:
  - Global min/max
  - Local min/max
  - Saddle points

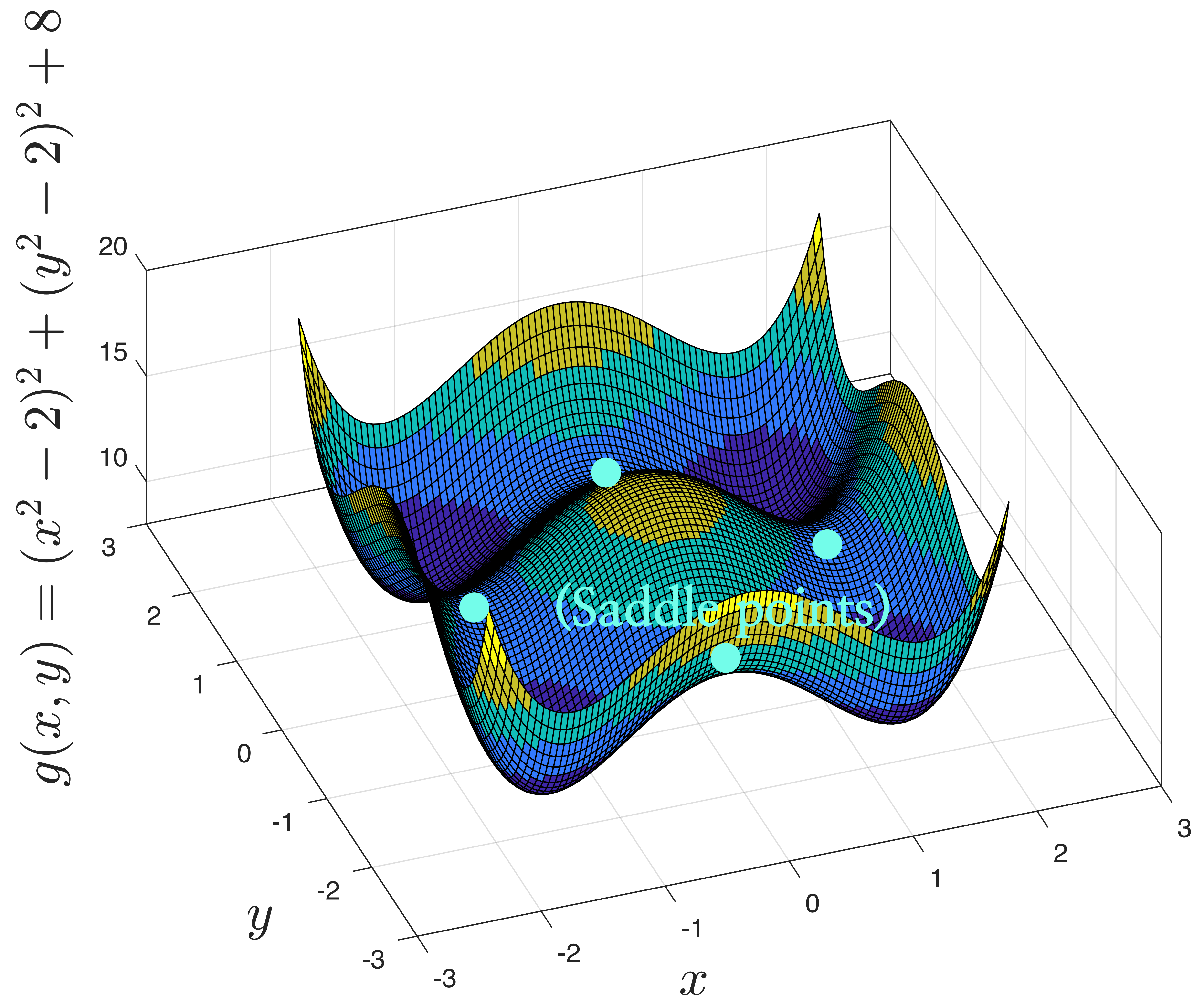


How many saddle points could be there?

- Toy example #2: $g(x, y) = f(x) + f(y) + 8 = (x^2 - 2)^2 + (y^2 - 2)^2 + 8$

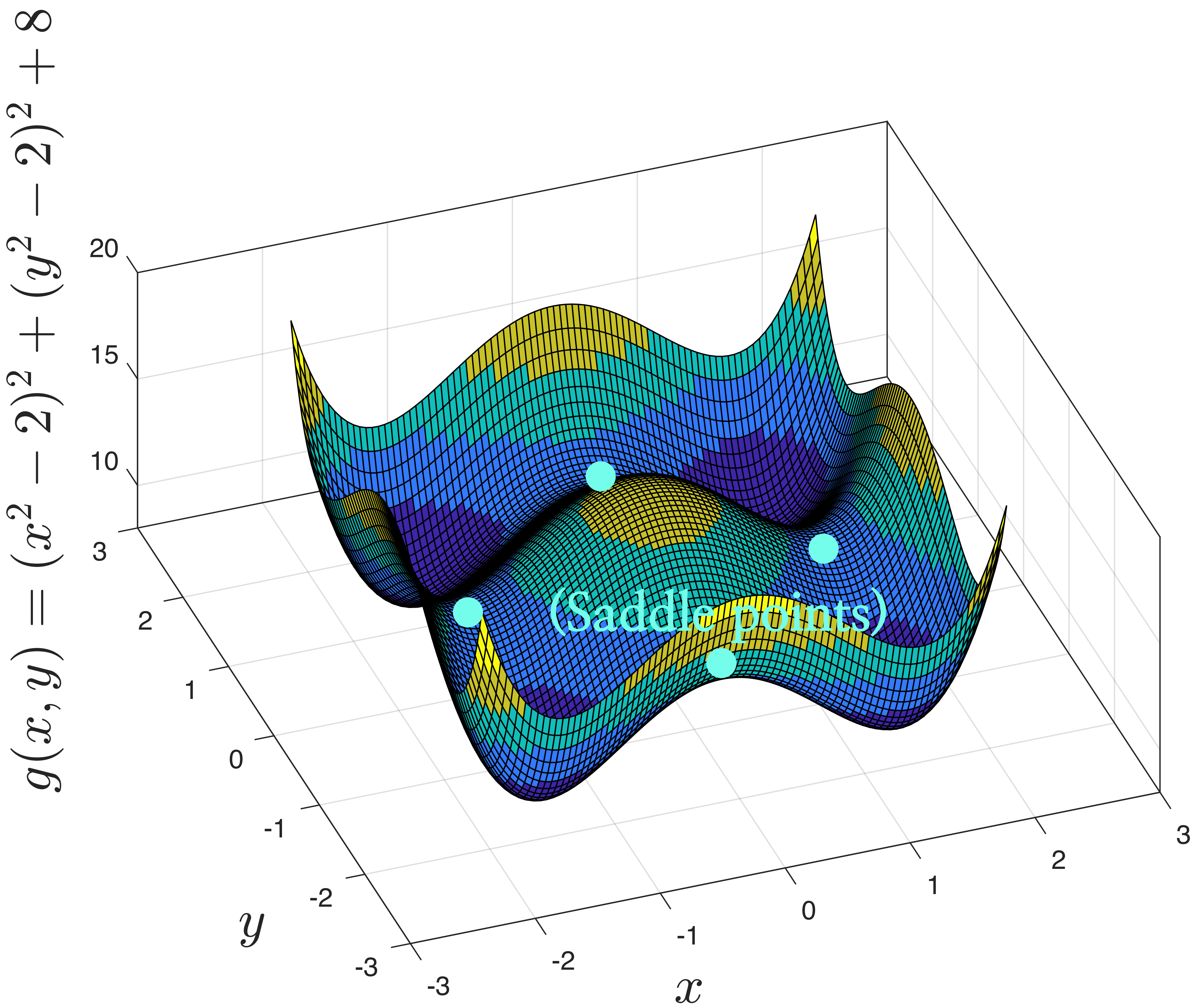
- Find:
  - Global min/max
  - Local min/max
  - Saddle points

From 1D to 2D, we get from 0 saddle points to 4 saddle points



How many saddle points could be there?

- Toy example #2: $g(x,y) = f(x) + f(y) + 8 = (x^2 - 2)^2 + (y^2 - 2)^2 + 8$



- Find:
  - Global min/max
  - Local min/max
  - Saddle points

From 1D to 2D, we get from 0 saddle points to 4 saddle points

- "Does it generalize?"

Yes! From 2D to 3D, we get 4 local minima, and 8 saddle points!

How many saddle points could be there?

- Another example: see papers at the Review section at the end of the lecture.

How many saddle points could be there?

- Another example: see papers at the Review section at the end of the lecture.
- In general, saddle points may emerge and their numbers increase (even exponentially) with **increasing dimensionality**.

How many saddle points could be there?

– Another example: see papers at the Review section at the end of the lecture.

– In general, saddle points may emerge and their numbers increase (even exponentially) with **increasing dimensionality**.

– **Does this mean that the situation is helpless**? Not necessarily!

How many saddle points could be there?

- Another example: see papers at the Review section at the end of the lecture.
- In general, saddle points may emerge and their numbers increase (even exponentially) with **increasing dimensionality**.
- **Does this mean that the situation is helpless**? Not necessarily!

1. Can we identify saddle points?

How many saddle points could be there?

- Another example: see papers at the Review section at the end of the lecture.
- In general, saddle points may emerge and their numbers increase (even exponentially) with **increasing dimensionality**.
- **Does this mean that the situation is helpless**? Not necessarily!

1. Can we identify saddle points?
2. How does methods such as gradient descent behave in practice?

How many saddle points could be there?

- Another example: see papers at the Review section at the end of the lecture.
- In general, saddle points may emerge and their numbers increase (even exponentially) with **increasing dimensionality**.
- **Does this mean that the situation is helpless**? Not necessarily!

1. Can we identify saddle points?
2. How does methods such as gradient descent behave in practice?
3. Are there conditions that indicate that always there is a way-out of saddle points?

Second–order derivative test

– Consider the Hessian at a critical point $x: \nabla^2 f(x) \in \mathbb{R}^{p \times p}$

Second–order derivative test

- Consider the Hessian at a critical point x: $\nabla^2 f(x) \in \mathbb{R}^{p \times p}$
- The Hessian is square and symmetric; we compute its eigenvalue decomp.:

$$\nabla^2 f(x) = U \Lambda U^\top$$

Second-order derivative test

- Consider the Hessian at a critical point x: $\nabla^2 f(x) \in \mathbb{R}^{p \times p}$
- The Hessian is square and symmetric; we compute its eigenvalue decomp.:

$$\nabla^2 f(x) = U \Lambda U^\top$$

- General rules:

  1. **Only positive eigenvalues: local minimum**

Second-order derivative test

- Consider the Hessian at a critical point x: $\nabla^2 f(x) \in \mathbb{R}^{p \times p}$
- The Hessian is square and symmetric; we compute its eigenvalue decomp.:

$$\nabla^2 f(x) = U \Lambda U^\top$$

- General rules:

  1. Only positive eigenvalues: local minimum

  Why? Positive eigenvalues mean positive definite Hessian. Thus:

$$\left\langle \nabla^2 f(x) u, u \right\rangle > 0, \ \forall u \neq 0 \in \mathbb{R}^p$$

Second–order derivative test

- Consider the Hessian at a critical point x: $\nabla^2 f(x) \in \mathbb{R}^{p \times p}$
- The Hessian is square and symmetric; we compute its eigenvalue decomp.:

$$\nabla^2 f(x) = U \Lambda U^\top$$

- General rules:

1. Only positive eigenvalues: local minimum

Why? Positive eigenvalues mean positive definite Hessian. Thus:

$$\langle \nabla^2 f(x) u, u \rangle > 0, \ \forall u \neq 0 \in \mathbb{R}^p$$

By second–order Taylor's expansion:

$$f(x + \eta u) \approx f(x) + \frac{\eta^2}{2} \langle \nabla^2 f(x) u, u \rangle > f(x)$$

(Local min.)

Second-order derivative test

- Consider the Hessian at a critical point x: $\nabla^2 f(x) \in \mathbb{R}^{p \times p}$
- The Hessian is square and symmetric: we compute its eigenvalue decomp.:

$$\nabla^2 f(x) = U \Lambda U^\top$$

- General rules:

1. **Only positive eigenvalues: local minimum**

Second-order derivative test

- Consider the Hessian at a critical point x: $\nabla^2 f(x) \in \mathbb{R}^{p \times p}$
- The Hessian is square and symmetric: we compute its eigenvalue decomp.:

$$\nabla^2 f(x) = U \Lambda U^\top$$

- General rules:

 1. **Only positive eigenvalues: local minimum**
 2. **Only negative eigenvalues: local maximum**

Second-order derivative test

- Consider the Hessian at a critical point x: $\nabla^2 f(x) \in \mathbb{R}^{p \times p}$
- The Hessian is square and symmetric: we compute its eigenvalue decomp.:

$$\nabla^2 f(x) = U \Lambda U^\top$$

- General rules:

  1. **Only positive eigenvalues: local minimum**
  2. **Only negative eigenvalues: local maximum**
  3. **Only positive and negative eigenvalues: (strict) saddle point**

(We will discuss about this later on)

Second–order derivative test

– Consider the Hessian at a critical point x: $\nabla^2 f(x) \in \mathbb{R}^{p \times p}$

– The Hessian is square and symmetric: we compute its eigenvalue decomp.:

$$\nabla^2 f(x) = U \Lambda U^\top$$

– General rules:

1. Only positive eigenvalues: local minimum
2. Only negative eigenvalues: local maximum
3. Only positive and negative eigenvalues: (strict) saddle point

(We will discuss about this later on)

4. Positive, negative and zero eigenvalues: general saddle point

What can we hope for at saddle points?

- One can use intuition from **second-order Taylor** expansion:

$$f(y) \approx f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2} \left\langle \nabla^2 f(x)(y - x), y - x \right\rangle$$

What can we hope for at saddle points?

- One can use intuition from **second-order Taylor** expansion:

$$f(y) \approx f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2} \left\langle \nabla^2 f(x)(y - x), y - x \right\rangle$$

  - Even if $\nabla f(x) = 0$, we hope we can find a direction $(y - x)$ such that

$$\left\langle \nabla^2 f(x)(y - x), y - x \right\rangle < 0$$

What can we hope for at saddle points?

- One can use intuition from **second-order Taylor** expansion:

$$f(y) \approx f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2} \left\langle \nabla^2 f(x)(y - x), y - x \right\rangle$$

  - Even if $\nabla f(x) = 0$, we hope we can find a direction $(y - x)$ such that

$$\left\langle \nabla^2 f(x)(y - x), y - x \right\rangle < 0$$

- Intuition suggests that saddle points with several directions that satisfy

$$\left\langle \nabla^2 f(x)(y - x), y - x \right\rangle \ll 0$$

  means that we can find directions that decrease the function (and thus, we escape saddle points)

What can we hope for at saddle points?

- One can use intuition from **second-order Taylor** expansion:

$$f(y) \approx f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2} \left\langle \nabla^2 f(x)(y - x), y - x \right\rangle$$

  - Even if $\nabla f(x) = 0$, we hope we can find a direction $(y - x)$ such that

$$\left\langle \nabla^2 f(x)(y - x), y - x \right\rangle < 0$$

- Intuition suggests that saddle points with several directions that satisfy

$$\left\langle \nabla^2 f(x)(y - x), y - x \right\rangle \ll 0$$

  means that we can find directions that decrease the function (and thus, we escape saddle points)

- Thus, we need to characterize the # of steps we might require to escape

Strict saddle property and functions

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

i. The gradient is large, i.e., $\|\nabla f(x)\|_2 \geq \alpha$

ii. The Hessian has at least one negative eigenvalue, bounded away from zero, i.e., $\lambda_{\min}\left(\nabla^2 f(x)\right) \leq -\gamma$

iii. x is near a local minimum``

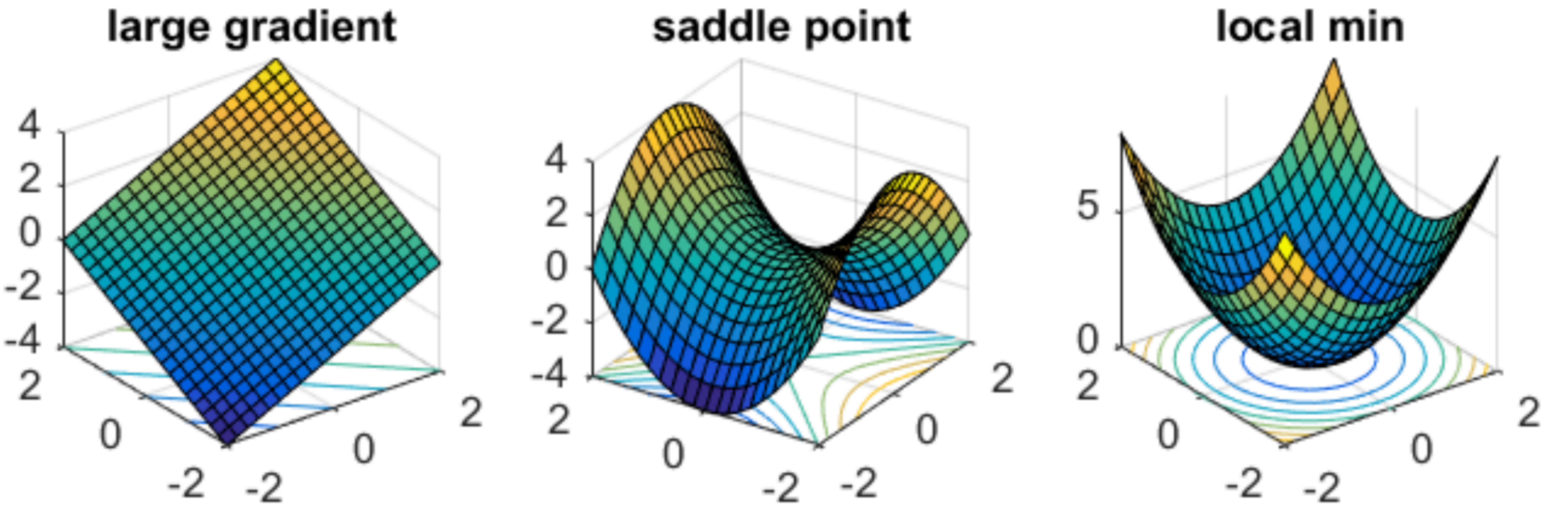
Strict saddle property and functions

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

i. The gradient is large, i.e., $\|\nabla f(x)\|_2 \geq \alpha$

ii. The Hessian has at least one negative eigenvalue, bounded away from zero, i.e., $\lambda_{\min}\left(\nabla^2 f(x)\right) \leq -\gamma$

iii. x is near a local minimum``



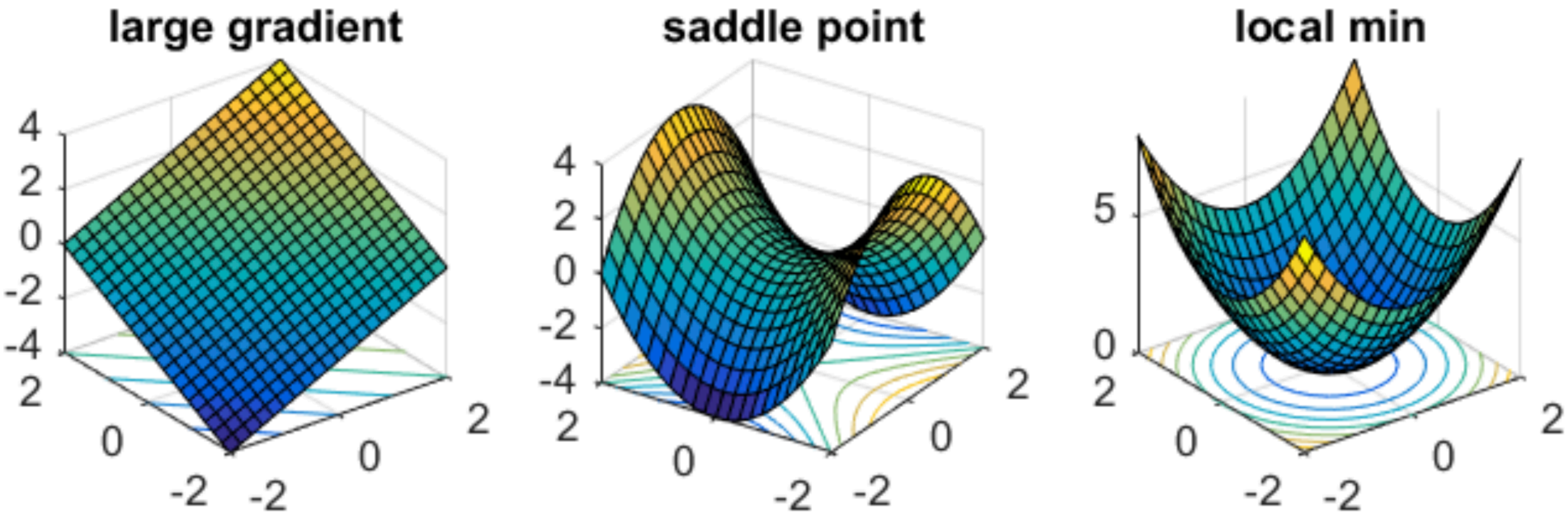
Strict saddle property and functions

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

i. The gradient is large, i.e., $\|\nabla f(x)\|_2 \geq \alpha$

ii. The Hessian has at least one negative eigenvalue, bounded away from zero, i.e., $\lambda_{\min}\left(\nabla^2 f(x)\right) \leq -\gamma$

iii. x is near a local minimum``



– Tensor decomposition, dictionary learning, phase retrieval, matrix sensing..

What can we do in practice?

- So far theory suggests that we should look at 2nd–order information
  - Methods such as **trust–region methods, and cubic regularization** handle saddles points this way ⟵ But might be time–consuming!

What can we do in practice?

- So far theory suggests that we should look at 2nd-order information
    - Methods such as **trust-region methods, and cubic regularization** handle saddles points this way ⟵ But might be time-consuming!
- Can we escape such saddle points with first-order methods such as GD?

``Really.. can we? Gradient information at saddle points is null``

What can we do in practice?

- So far theory suggests that we should look at 2nd-order information
  - Methods such as **trust-region methods, and cubic regularization** handle saddles points this way ⟵ But might be time-consuming!
- Can we escape such saddle points with first-order methods such as GD?

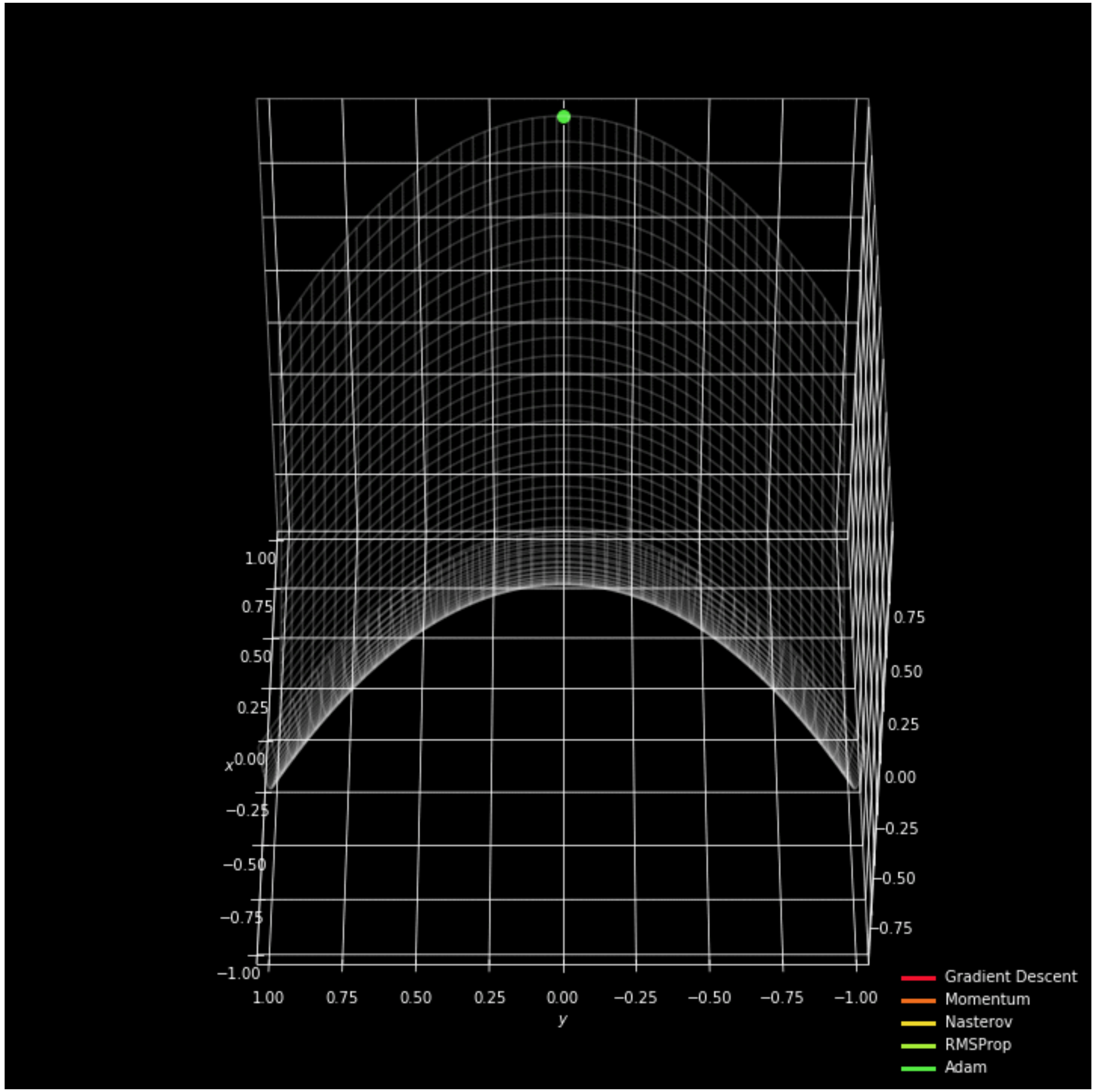
"Really.. can we? Gradient information at saddle points is null"

- Key observation: (strict) saddle points are quite **unstable**!

(Starting from the saddle point, if we perturb our location only a bit, we will fall from that point)

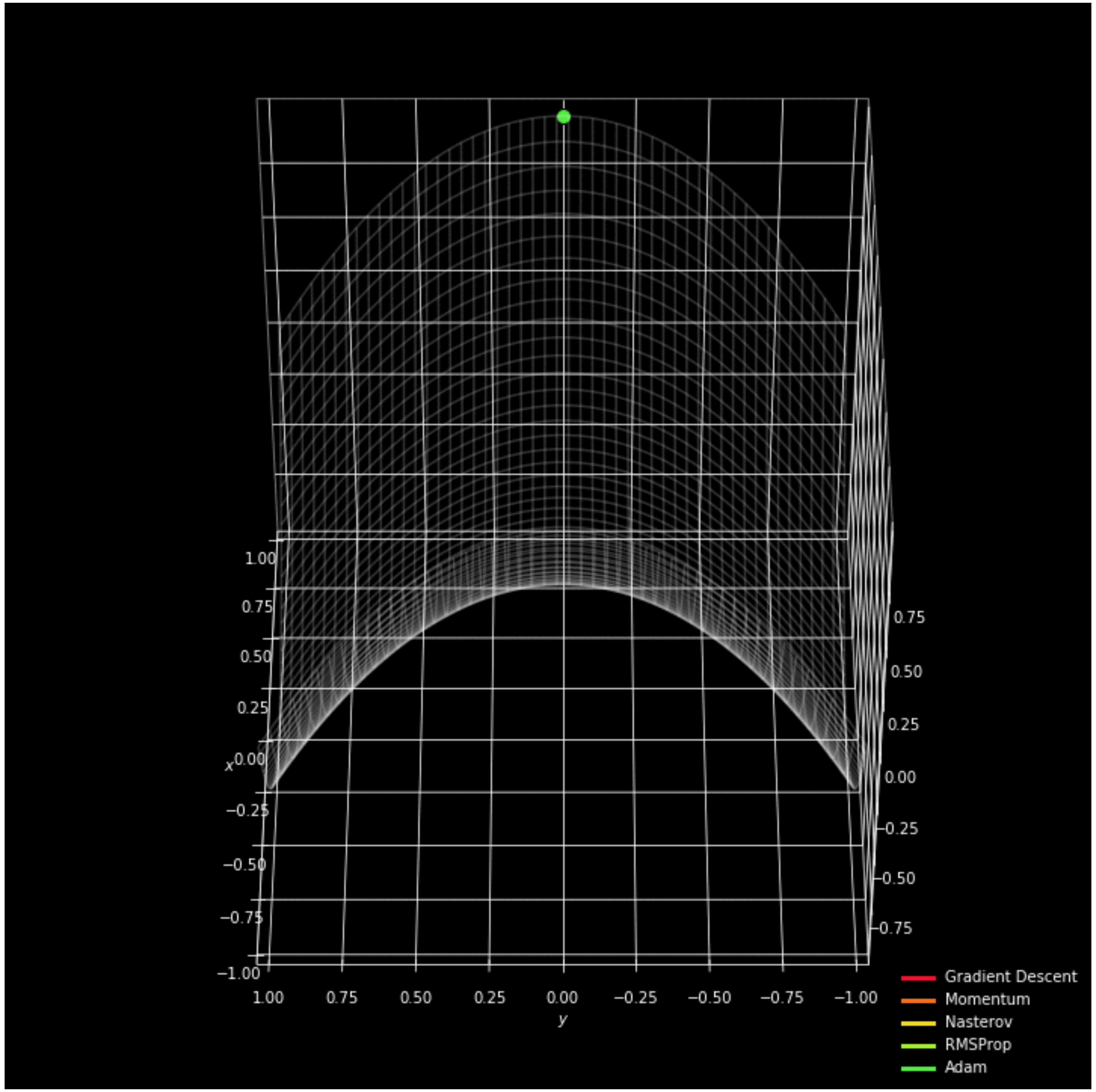
What can we do in practice?

(Tribute: unknown)



What can we do in practice?

(Tribute: unknown)



What can we do in practice?

- So far theory suggests that we should look at 2nd-order information
  - Methods such as **trust-region methods, and cubic regularization** handle saddles points this way ⟵ But very time-consuming!
- Can we escape such saddle points with first-order methods such as GD?

  "Really.. can we? Gradient information at saddle points is null"
- Key observation: (strict) saddle points are quite **unstable**!

  (Starting from the saddle point, if we perturb our location only a bit, we will fall from that point)

What can we do in practice?

- So far theory suggests that we should look at 2nd-order information
  - Methods such as **trust-region methods, and cubic regularization** handle saddles points this way ← But very time-consuming!
- Can we escape such saddle points with first-order methods such as GD?

"Really.. can we? Gradient information at saddle points is null"

- Key observation: (strict) saddle points are quite **unstable**!

(Starting from the saddle point, if we perturb our location only a bit, we will fall from that point)

- What if we impute some **noise** in the gradient descent step?

What can we do in practice?

- So far theory suggests that we should look at 2nd−order information
  - Methods such as **trust−region methods, and cubic regularization** handle saddles points this way ⟵ But very time−consuming!
- Can we escape such saddle points with first−order methods such as GD?

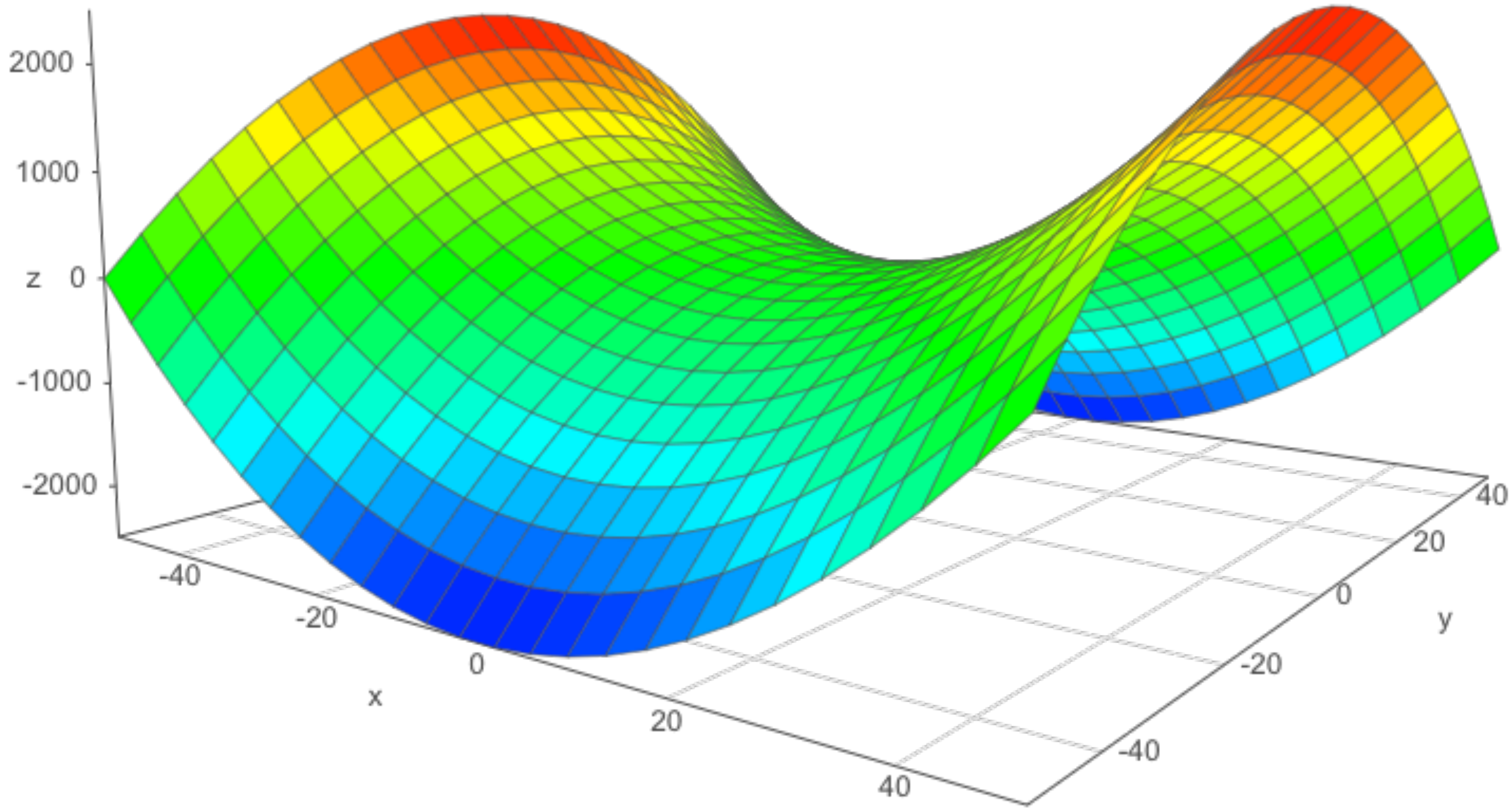
  "Really.. can we? Gradient information at saddle points is null"
- Key observation: (strict) saddle points are quite **unstable**!

  (Starting from the saddle point, if we perturb our location only a bit, we will fall from that point)
- What if we impute some **noise** in the gradient descent step?

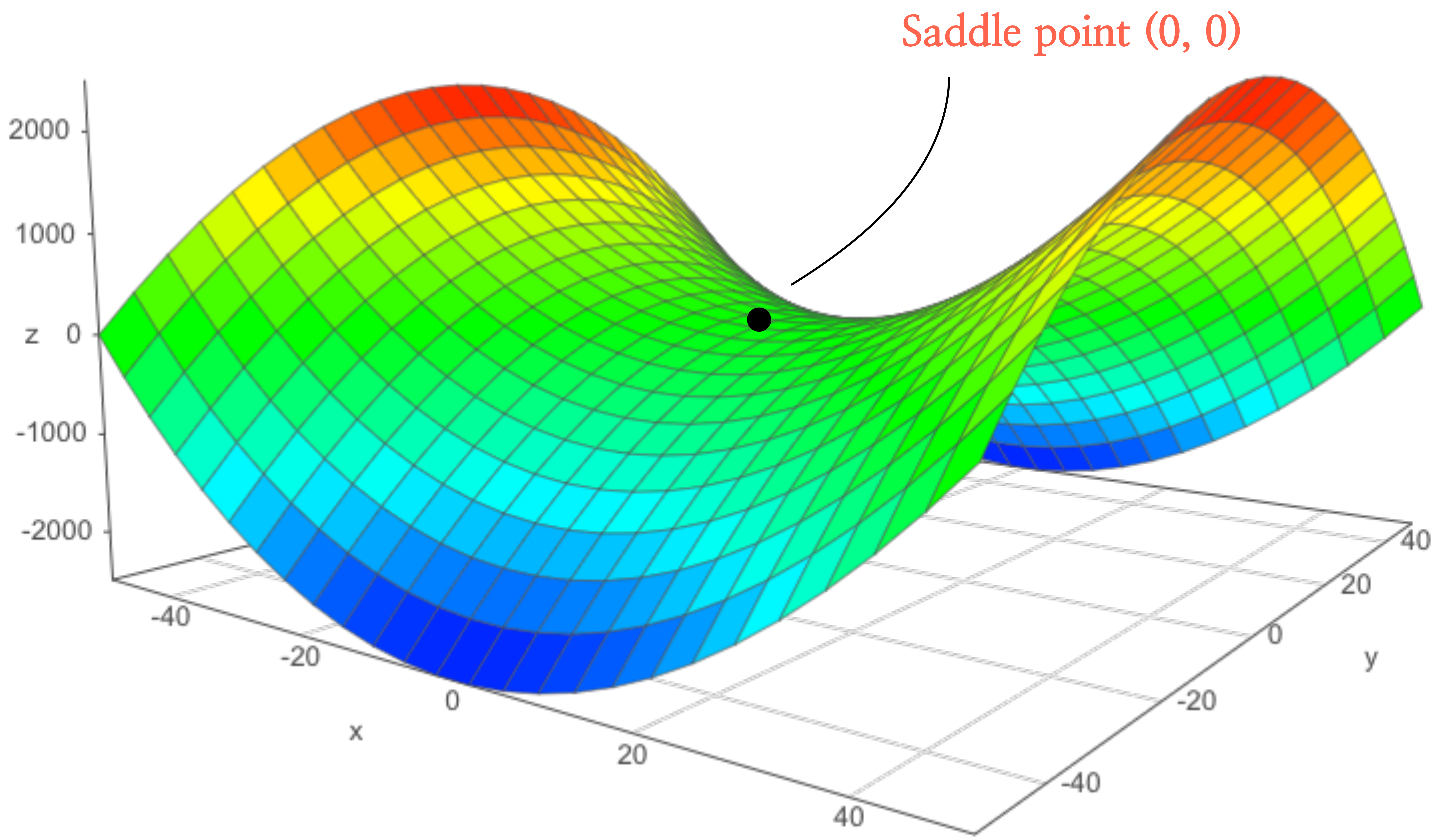
$$x_{t+1} = x_t - \eta \nabla f(x_t) + \varepsilon, \;\; \varepsilon \sim \eta \cdot \mathcal{S}^{p-1}$$

(Noisy gradient descent)

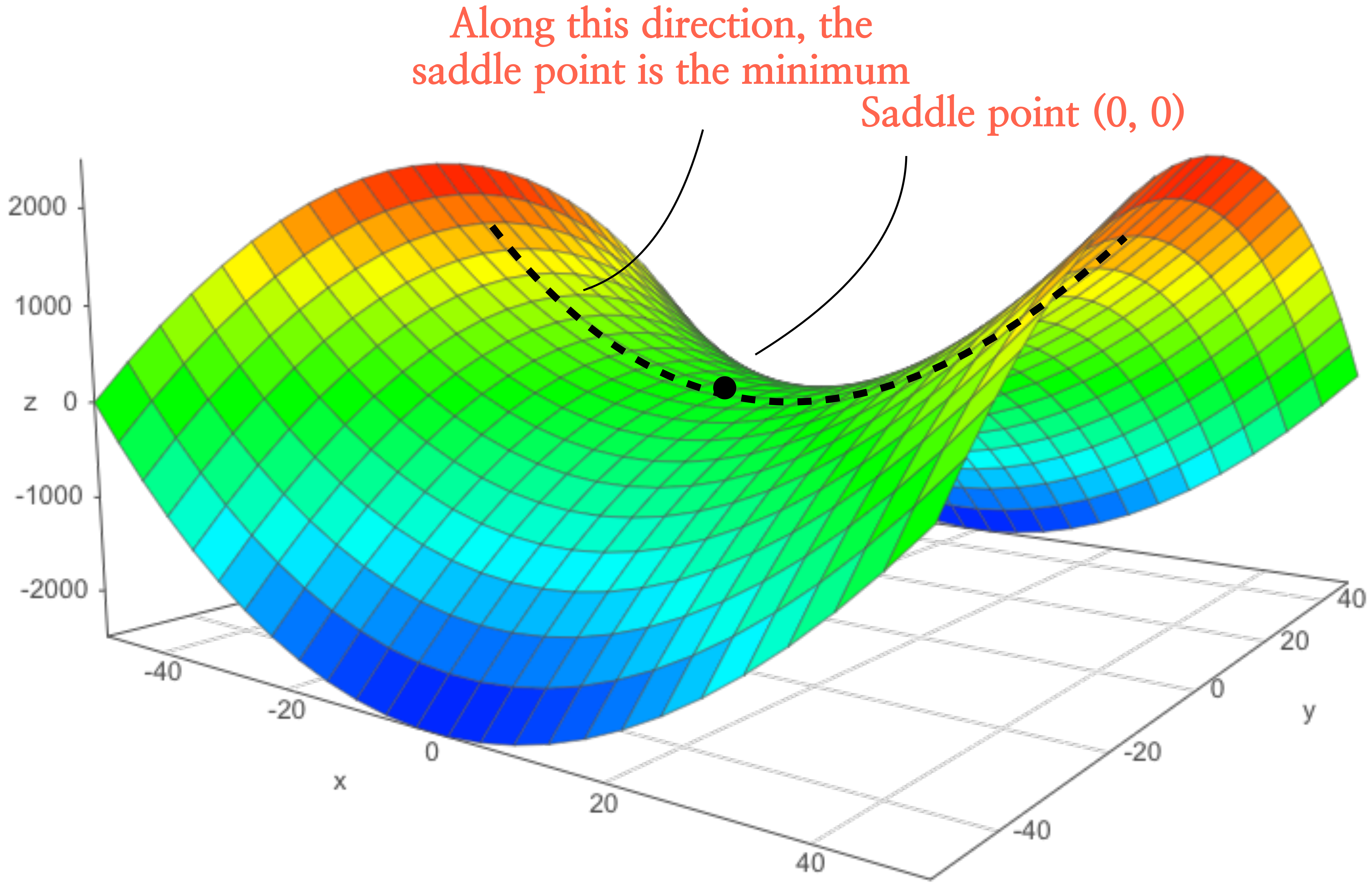
What can we do in practice?



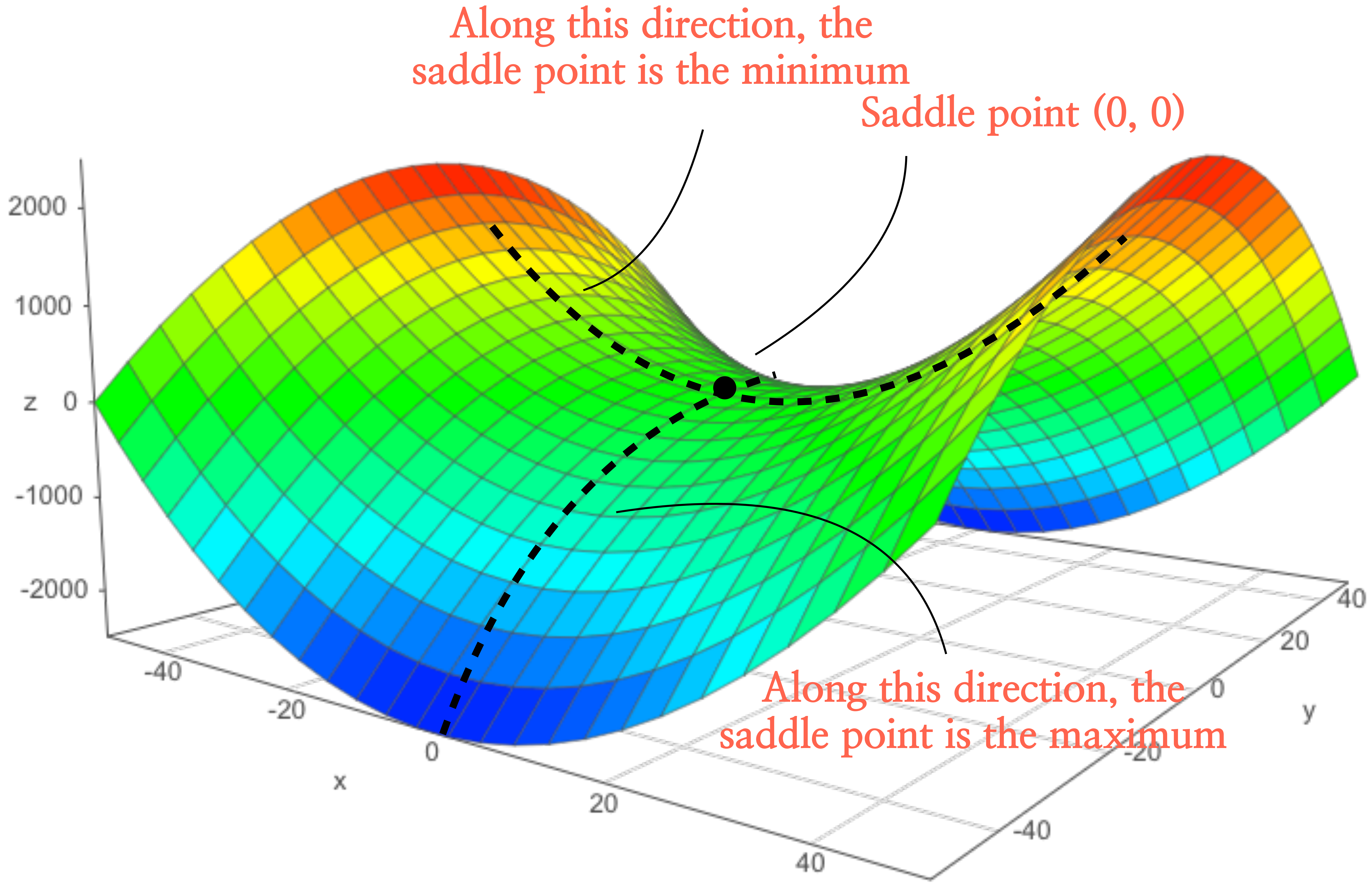
What can we do in practice?



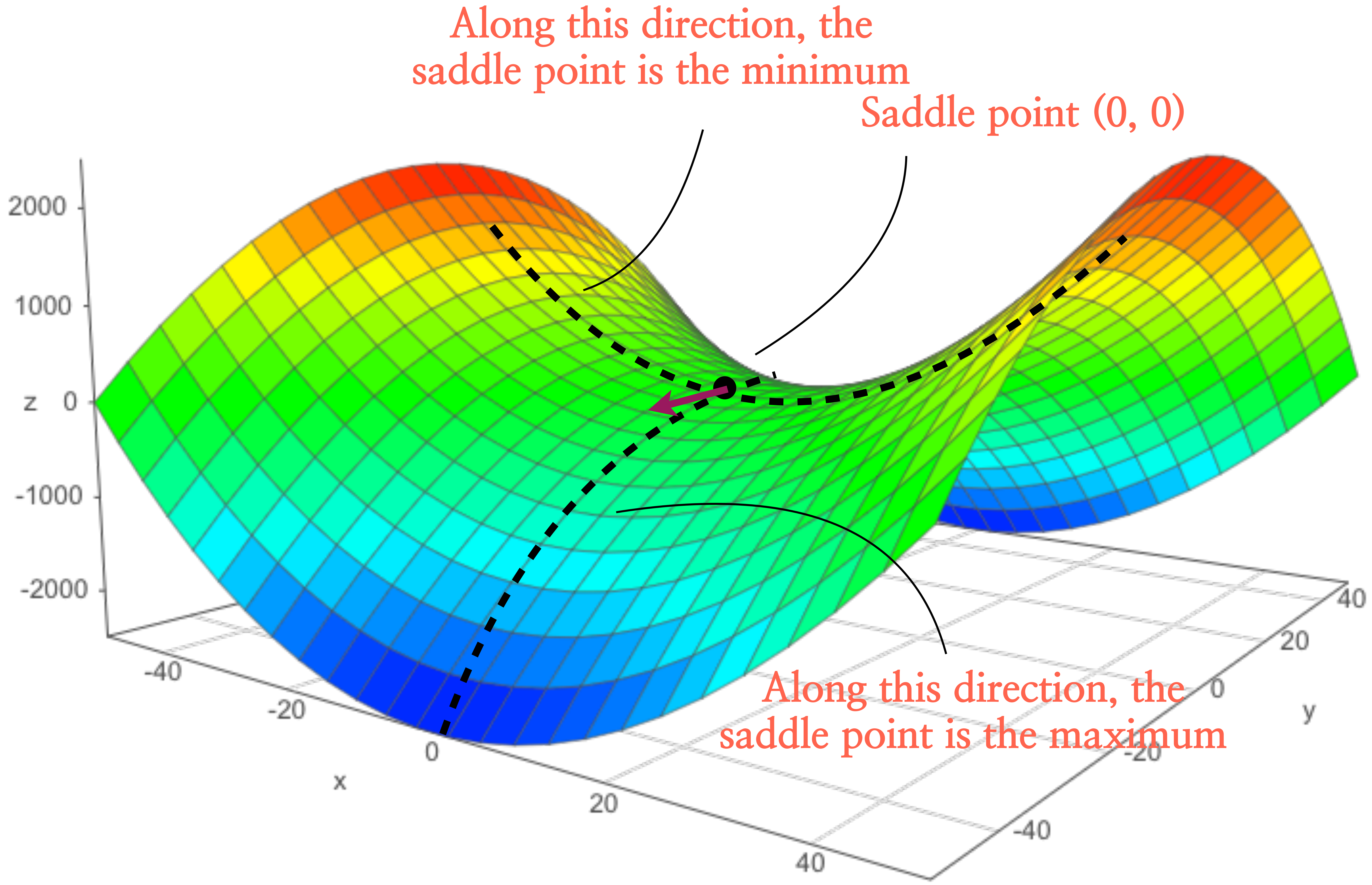
What can we do in practice?



What can we do in practice?



What can we do in practice?



What can we do in practice?

- So far theory suggests that we should look at 2nd-order information
  - Methods such as **trust-region methods, and cubic regularization** handle saddles points this way ⟵ But very time-consuming!
- Can we escape such saddle points with first-order methods such as GD?

  "Really.. can we? Gradient information at saddle points is null"
- Key observation: (strict) saddle points are quite **unstable**!

  (Starting from the saddle point, if we perturb our location only a bit, we will fall from that point)

What can we do in practice?

- So far theory suggests that we should look at 2nd-order information
  - Methods such as **trust-region methods, and cubic regularization** handle saddles points this way ⟵ But very time-consuming!
- Can we escape such saddle points with first-order methods such as GD?

  "Really.. can we? Gradient information at saddle points is null"
- Key observation: (strict) saddle points are quite **unstable**!

  (Starting from the saddle point, if we perturb our location only a bit, we will fall from that point)
- Or even easier, rely on **stochastic gradient descent** for noise imputation:

What can we do in practice?

- So far theory suggests that we should look at 2nd-order information
  - Methods such as **trust-region methods, and cubic regularization** handle saddles points this way ⟵ But very time-consuming!
- Can we escape such saddle points with first-order methods such as GD?

  "Really.. can we? Gradient information at saddle points is null"
- Key observation: (strict) saddle points are quite **unstable**!

  (Starting from the saddle point, if we perturb our location only a bit, we will fall from that point)
- Or even easier, rely on **stochastic gradient descent** for noise imputation:

$$x_{t+1} = x_t - \eta \nabla f_{i_t}(x_t) = x_t - \eta \nabla f(x_t) + \varepsilon, \;\text{where } \varepsilon = \eta \left(\nabla f(x_t) - \nabla f_{i_t}(x_t) \right)$$

("Noisy" stochastic gradient descent - stochasticity is not a problem, but a feature)

How does noisy gradient descent perform?

How does noisy gradient descent perform?

- Informal result:

> *``Noisy gradient descent finds a local minimum of a function that satisfies the strict saddle property in polynomial time``*

(This justifies that the method is not hopeless in practice)

How does noisy gradient descent perform?

- Informal result:

> *``Noisy gradient descent finds a local minimum of a function that satisfies the strict saddle property in polynomial time``*

(This justifies that the method is not hopeless in practice)

- Formal result:

> *``With probability* $1-\delta$*, after* $t \geq \log \frac{p}{\eta^2} \cdot \log \frac{2}{\delta}$ *iterations, noisy gradient descent converges close to a local minimum* $x^\star$ *such that* $\|x_t - x^\star\|_2 \leq \epsilon$*. Here,*
>
> $$\eta < \min\left\{ \frac{\epsilon^2}{\log(1/\epsilon\delta)}, \frac{\mu}{L^2}, \frac{\xi^2}{\log(1/\xi\delta)}, \frac{1}{(L+\rho)^2}, \frac{\alpha^2}{L} \right\}$$
>
> *where* ρ *is the Hessian-Lipschitz constant.``*

How does noisy gradient descent perform?

– Informal result:

> ``*Noisy gradient descent finds a local minimum of a function that satisfies the strict saddle property in polynomial time*``

(This justifies that the method is not hopeless in practice)

– Formal result:

> ``*With probability* $1 - \delta$*, after* $t \geq \log \frac{p}{\eta^2} \cdot \log \frac{2}{\delta}$ *iterations, noisy gradient descent converges close to a local minimum* $x^\star$ *such that* $\|x_t - x^\star\|_2 \leq \epsilon$*. Here,*
>
> $$\eta < \min \left\{ \frac{\epsilon^2}{\log(1/\epsilon\delta)}, \frac{\mu}{L^2}, \frac{\xi^2}{\log(1/\xi\delta)}, \frac{1}{(L+\rho)^2}, \frac{\alpha^2}{L} \right\}$$
>
> *where* ρ *is the Hessian-Lipschitz constant.*``

– Overall, total runtime could be up to $O(p^3)$ — differently, $\widetilde{O}(1/\epsilon^4)$ iters.

Should we worry about saddle points?
A different perspective

- From previous plots, we can easily see that saddle points can be unstable!
  (Moving slightly from saddle points, we fall off the saddle)

Should we worry about saddle points? A different perspective

- From previous plots, we can easily see that saddle points can be unstable!

  (Moving slightly from saddle points, we fall off the saddle)

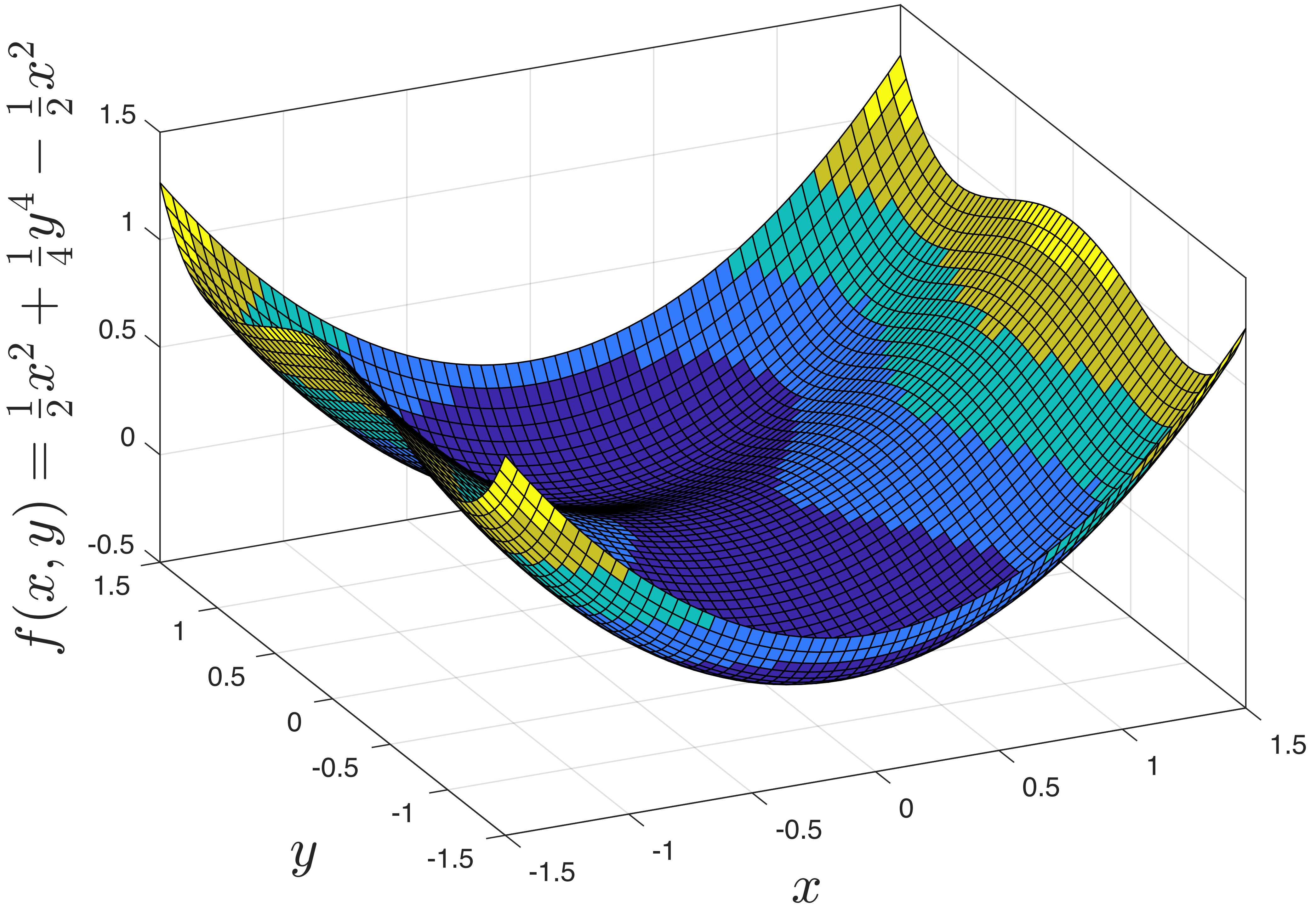
- Consider another toy example: $f(x, y) = \frac{1}{2}x^2 + \frac{1}{4}y^4 - \frac{1}{2}y^2$

Should we worry about saddle points? A different perspective

– From previous plots, we can easily see that saddle points can be unstable!

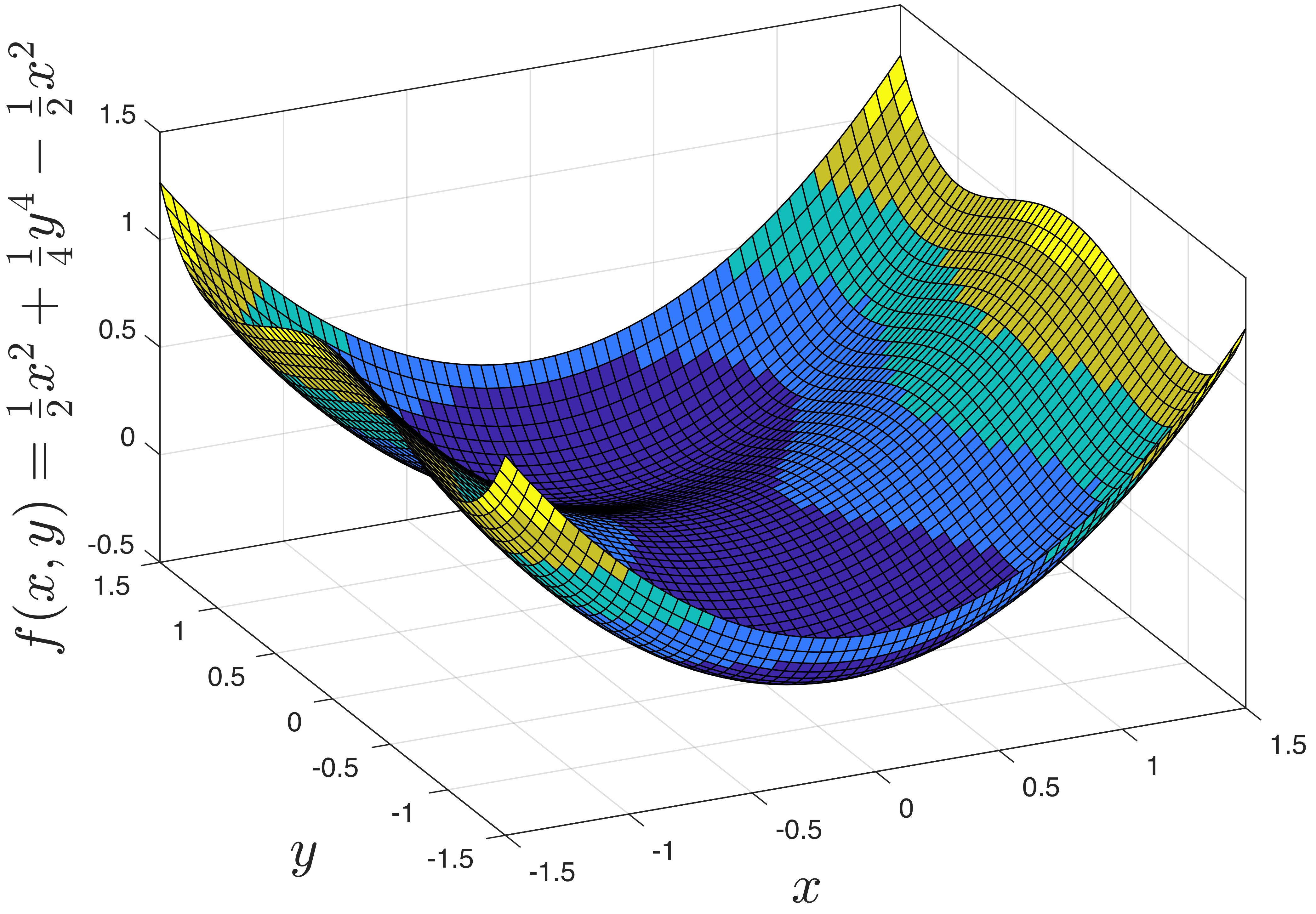
(Moving slightly from saddle points, we fall off the saddle)

– Consider another toy example: $f(x,y) = \frac{1}{2}x^2 + \frac{1}{4}y^4 - \frac{1}{2}y^2$



Should we worry about saddle points? A different perspective

- From previous plots, we can easily see that saddle points can be unstable!
  (Moving slightly from saddle points, we fall off the saddle)
- Consider another toy example: $f(x,y) = \frac{1}{2}x^2 + \frac{1}{4}y^4 - \frac{1}{2}y^2$



- Critical points: $\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

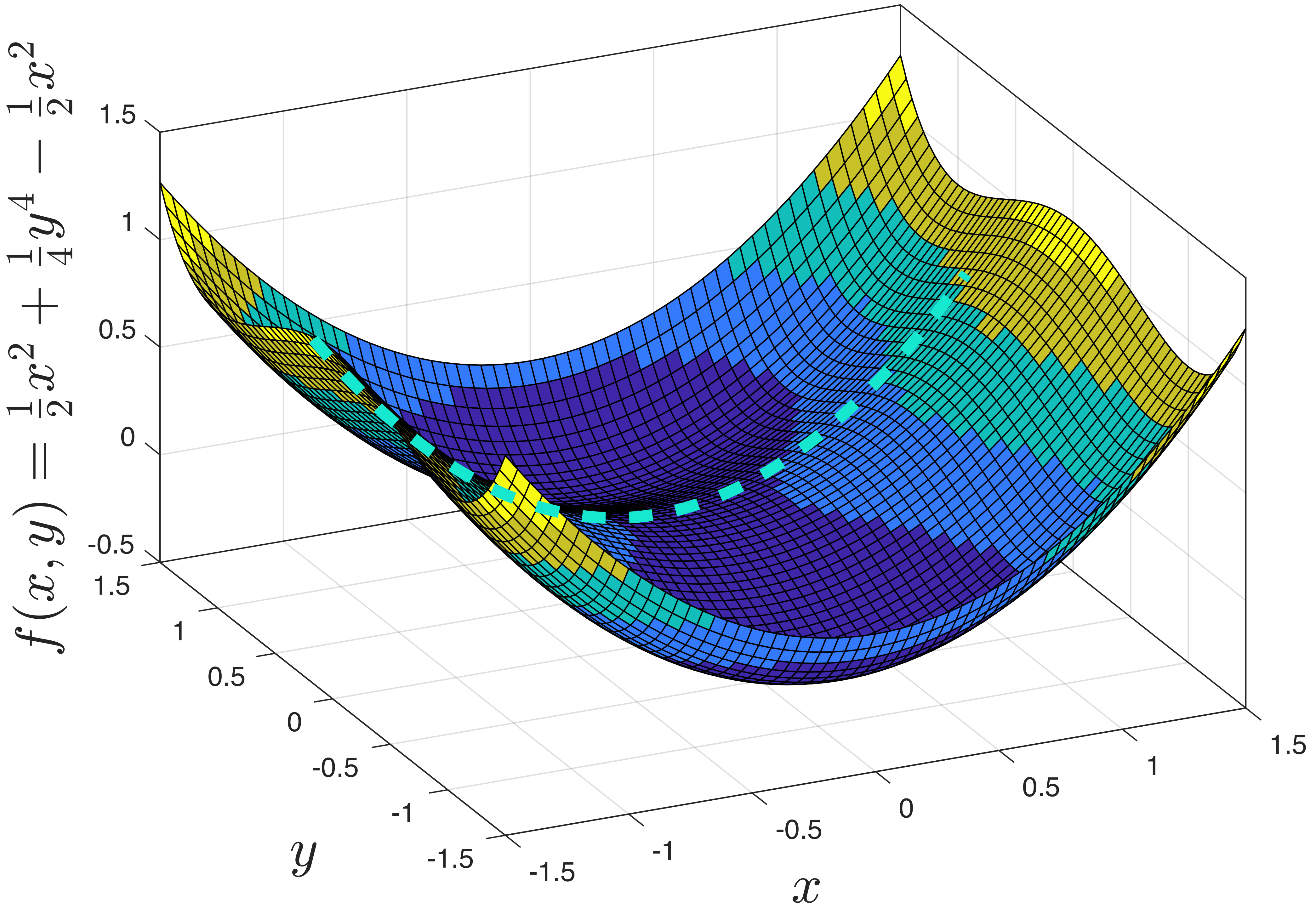
Saddle point   Local minima

Should we worry about saddle points? A different perspective

- From previous plots, we can easily see that saddle points can be unstable!

(Moving slightly from saddle points, we fall off the saddle)

- Consider another toy example: $f(x, y) = \frac{1}{2}x^2 + \frac{1}{4}y^4 - \frac{1}{2}y^2$



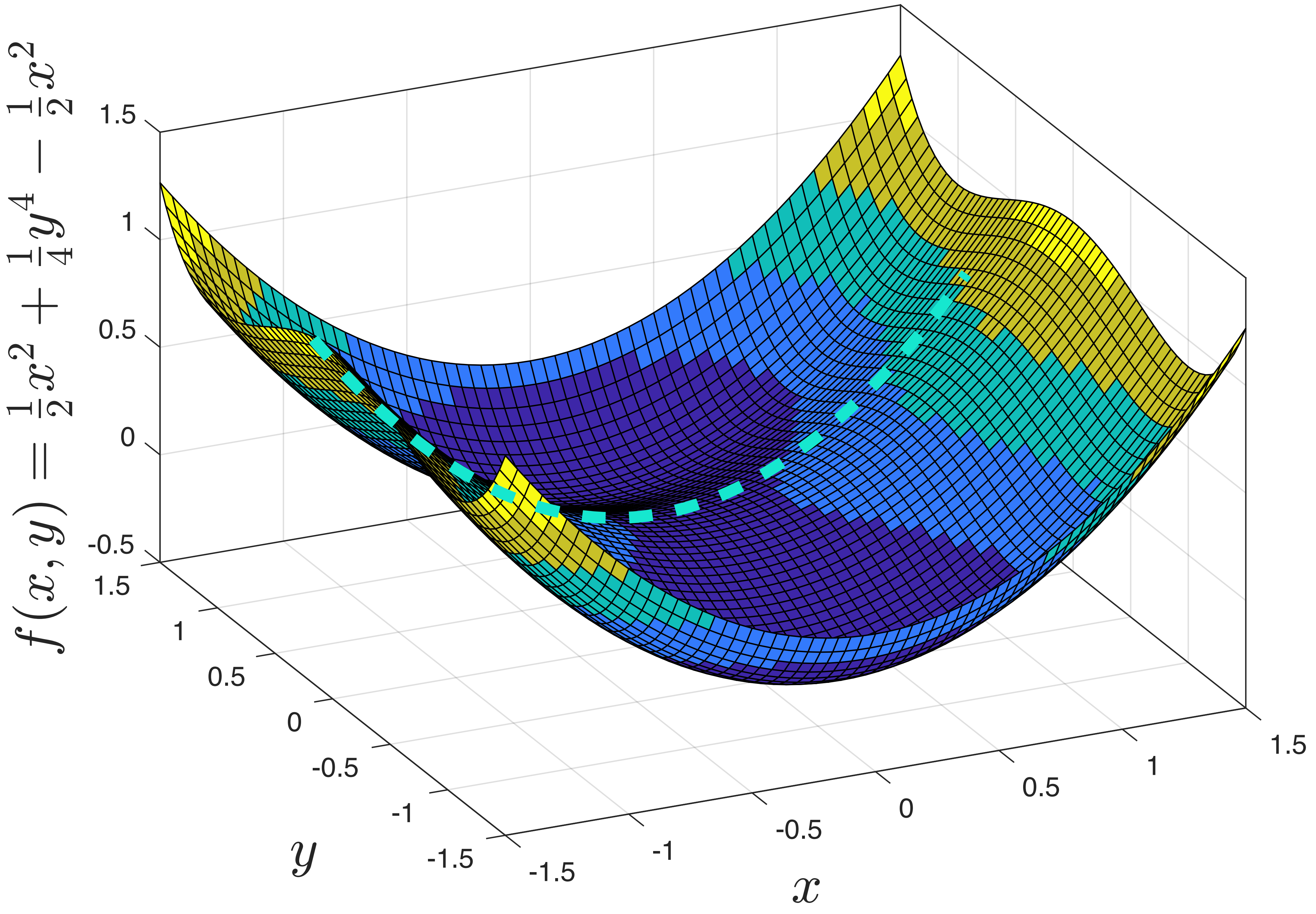
- Critical points: $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $\begin{bmatrix} 0 \\ -1 \end{bmatrix}$, $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Saddle point **Local minima**

- Any initialization of the form $\begin{bmatrix} x \\ 0 \end{bmatrix}$ (cyan) leads to convergence to the saddle point

Should we worry about saddle points? A different perspective

- From previous plots, we can easily see that saddle points can be unstable!
  (Moving slightly from saddle points, we fall off the saddle)
- Consider another toy example: $f(x,y) = \frac{1}{2}x^2 + \frac{1}{4}y^4 - \frac{1}{2}y^2$



  - Critical points: $\begin{bmatrix}0\\0\end{bmatrix}$, $\begin{bmatrix}0\\-1\end{bmatrix}$, $\begin{bmatrix}0\\1\end{bmatrix}$
    Saddle point   Local minima
  - Any initialization of the form $\begin{bmatrix}x\\0\end{bmatrix}$ (cyan) leads to convergence to the saddle point
  - But, **any other initialization converges to local minimizer**!
    (With random initialization, this happens with prob. 1)

Should we worry about saddle points?
A different perspective

- This idea was made more rigorous using ideas from dynamical systems

(In particular, using the Stable Manifold Theorem – outside our scope)

Should we worry about saddle points?
A different perspective

- This idea was made more rigorous using ideas from dynamical systems

  (In particular, using the Stable Manifold Theorem – outside our scope)

- The idea is that gradient descent satisfies such a theorem, and the set of saddle points (under the assumptions made by the theory so far) has measure zero!

Should we worry about saddle points? A different perspective

- This idea was made more rigorous using ideas from dynamical systems
  (In particular, using the Stable Manifold Theorem – outside our scope)

- The idea is that gradient descent satisfies such a theorem, and the set of saddle points (under the assumptions made by the theory so far) has measure zero!

- In practice: **if you pick any random initial point, you are safe not to converge to a saddle point**

When we know more about our problem at hand

When we know more about our problem at hand

- We know how to escape saddle points; what about local minima?
  Can we infer that local = global minima in non-convex settings?

When we know more about our problem at hand

- We know how to escape saddle points; what about local minima?
  Can we infer that local = global minima in non-convex settings?

- Example: Matrix sensing using RIP and PSD matrix factorization

Whiteboard

When we know more about our problem at hand

- We know how to escape saddle points; what about local minima? Can we infer that local = global minima in non-convex settings?

- Example: Matrix sensing using RIP and PSD matrix factorization

Whiteboard

- Similar results have been proven for: phase retrieval, matrix completion, dictionary recovery, semidefinite programming (SDPs), signal recovery from quadratic measurements, ..

Landscape characterization with strict saddles

Landscape characterization with strict saddles

- Write down gradient and Hessian expressions

Landscape characterization with strict saddles

- Write down gradient and Hessian expressions
- Compute critical/stationary points condition:

$$\nabla f(x) = 0$$

← Solving analytically this equation provides an expression for stationary points

Landscape characterization with strict saddles

- Write down gradient and Hessian expressions
- Compute critical/stationary points condition:

$$\nabla f(x) = 0$$

← Solving analytically this equation provides an expression for stationary points

- Consider all cases of stationary points:

Landscape characterization with strict saddles

- Write down gradient and Hessian expressions
- Compute critical/stationary points condition:

$$\nabla f(x) = 0$$

← Solving analytically this equation provides an expression for stationary points

- Consider all cases of stationary points:
  - For local minima, we analyze the quadratic form

$$z^\top \nabla^2 f(x) z \geq 0, \ \forall z, \ \text{and given stationary point } x$$

  and compare with global minima (to show potential equivalence)

Landscape characterization with strict saddles

- Write down gradient and Hessian expressions
- Compute critical/stationary points condition:

$$\boxed{\nabla f(x) = 0}$$

← Solving analytically this equation provides an expression for stationary points

- Consider all cases of stationary points:
  - For local minima, we analyze the quadratic form

    $$z^\top \nabla^2 f(x) z \geq 0, \ \forall z, \text{ and given stationary point } x$$

    and compare with global minima (to show potential equivalence)
- For saddles, identify a (negative) upper bound for

$$\lambda_{\min} \left(\nabla^2 f(x) \right)$$

Conclusion

- We discussed about **types of stationary points**, focus on **saddle points** and study some of their properties

- We introduced conditions that allow **escaping from saddle points**

- We studied (overview) matrix sensing as a test case, and how to prove ``no spurious local minima`` arguments