Plan for today

- Dynamic Programming
  - Winning in Vegas Problem

- Nonlinear Optimization
  - Local and global min
  - Convex functions
  - Verifying convexity

DYNAMIC PROGRAMMING

… where we see the algorithmic technique of dynamic programming (through examples)

WINNING IN VEGAS

Example 5: Winning in Vegas

- You would like to show-off your casino skills to your friend
- Friend challenges you to the following game:
  - You start with **3** chips and you are allowed to play **3** rounds
  - In each round,
    - you can bet either none or some, or all of the chips that you have at the beginning of that round
    - Casino odds: 1/3 probability of losing all chips that you bet
      2/3 probability of doubling the chips that you bet
- You win the game if you end up with at least 5 chips after 3 rounds; otherwise you lose

Question: determine the number of chips to bet in each round to
maximize the probability of winning the game against your friend

States, Stages, Decision Vars, Optimality Criterion

Stages: 3 stages (one for each round) and one dummy terminal stage

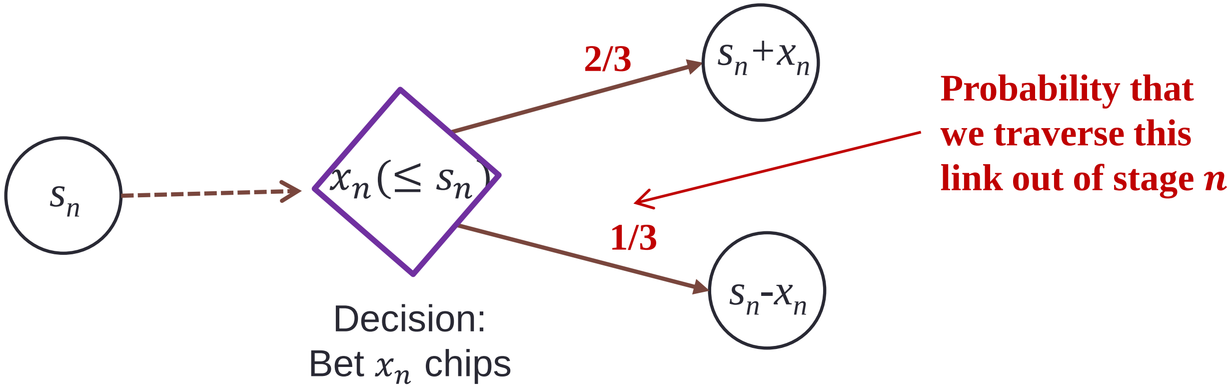
States: s_n ≔ No. of chips available at the beginning of stage n (i.e., round n)

Start with three chips, so $s_1 = 3$

Decision Variables: x_n ≔ No. of chips to bet at stage n (i.e., round n)

x_n can be at most s_n

Optimality Criterion: Maximize the probability of at least 5 chips at the terminal stage



Value function

finishing 3 rounds with at least 5 chips

$f_n(s_n, x_n) :=$ Probability of winning the game by starting from state s_n in stage n, making immediate decision to bet x_n and optimal decisions thereafter

$$f_n^*(s_n) = \max_{x_n \in \{0,\ldots,s_n\}} f_n(s_n, x_n)$$

Need: $f_1^*(s_1 = 3)$

Value function: final stage

finishing 3 rounds with at least 5 chips

$f_n(s_n, x_n) :=$ Probability of $\underbrace{\text{winning the game}}$ by starting from state s_n in stage n, making immediate decision to bet x_n and optimal decisions thereafter

$$f_n^*(s_n) = \max_{x_n \in \{0,\dots,s_n\}} f_n(s_n, x_n)$$

Need: $f_1^*(s_1 = 3)$

At the terminal (4th) stage,

either at least 5 chips $\Rightarrow$ win

or less than 5 chips $\Rightarrow$ lose

$f_4^*(s_4) = 1$ if $s_4 \geq 5$

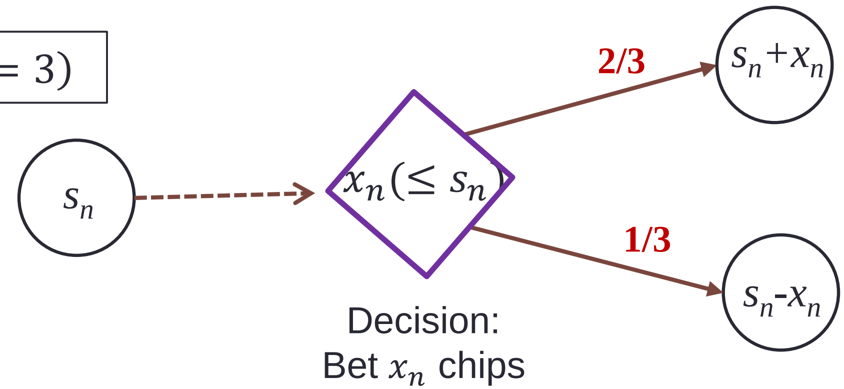
$$f_4^*(0) = f_4^*(1) = f_4^*(2) = f_4^*(3) = f_4^*(4) = 0$$

Value function: recursive relationship

$f_n(s_n, x_n) :=$ Probability of winning the game (finishing 3 rounds with at least 5 chips) by starting from state s_n in stage n, making immediate decision to bet x_n and optimal decisions thereafter

$$f_n^*(s_n) = \max_{x_n \in \{0,\dots,s_n\}} f_n(s_n, x_n)$$

Need: $f_1^*(s_1 = 3)$



At the terminal (4th) stage,

either at least 5 chips $\Rightarrow$ win

or less than 5 chips $\Rightarrow$ lose

$f_4^*(s_4) = 1$ if $s_4 \geq 5$

$$f_4^*(0) = f_4^*(1) = f_4^*(2) = f_4^*(3) = f_4^*(4) = 0$$

At stage n,

$$f_n(s_n, x_n) = \frac{1}{3} f_{n+1}^*(s_n - x_n) + \frac{2}{3} f_{n+1}^*(s_n + x_n)$$

$$f_n^*(s_n) = \max_{x_n \in \{0,1,\dots,s_n\}} f_n(s_n, x_n)$$

$f_n(s_n, x_n) =$

Probability of losing x_n $\times$ Probability of winning by optimal decisions with the remaining chips after losing x_n in the remaining rounds

$+$

Probability of doubling x_n $\times$ Probability of winning by optimal decisions with the remaining chips after doubling x_n in the remaining rounds

Solution Procedure: Backward induction, Stage 3

$$f_n(s_n, x_n) = \frac{1}{3} f^*_{n+1}(s_n - x_n) + \frac{2}{3} f^*_{n+1}(s_n + x_n)$$

$$f^*_n(s_n) = \max_{x_n \in \{0,1,\ldots,s_n\}} f_n(s_n, x_n)$$

Need: $f^*_1(s_1 = 3)$

$f^*_4(s_4) = 1$ if $s_4 \geq 5$

$f^*_4(0) = f^*_4(1) = f^*_4(2) = f^*_4(3) = f^*_4(4) = 0$

$$f_3(s_3, x_3) = \frac{1}{3} f^*_4(s_3 - x_3) + \frac{2}{3} f^*_4(s_3 + x_3)$$

$$f^*_3(s_3) = \max_{x_3 \in \{0,1,\ldots,s_3\}} f_3(s_3, x_3)$$

	$f_3(s_3, x_3)$ evaluated at						
s_3	$x_3 = 0$	$x_3 = 1$	$x_3 = 2$	$x_3 = 3$	$x_3 = 4$	$f^*_3(s_3)$	x^*_3
0	0					0	–
1	0	0				0	–
2	0	0	0			0	–
3	0	0	2/3	2/3		2/3	2 or 3
4	0	2/3	2/3	2/3	2/3	2/3	1 or 2 or 3 or 4
≥ 5	1					1	0

Solution Procedure: Backward induction, Stage 2

$$f_n(s_n, x_n) = \frac{1}{3} f_{n+1}^*(s_n - x_n) + \frac{2}{3} f_{n+1}^*(s_n + x_n)$$

$$f_n^*(s_n) = \max_{x_n \in \{0,1,\dots,s_n\}} f_n(s_n, x_n)$$

Need: $f_1^*(s_1 = 3)$

s_3	$f_3^*(s_3)$
0	0
1	0
2	0
3	2/3
4	2/3
≥ 5	1

$$f_2(s_2, x_2) = \frac{1}{3} f_3^*(s_2 - x_2) + \frac{2}{3} f_3^*(s_2 + x_2)$$

$$f_2^*(s_2) = \max_{x_2 \in \{0,1,\dots,s_2\}} f_2(s_2, x_2)$$

	$f_2(s_2, x_2)$ evaluated at						
s_2	$x_2 = 0$	$x_2 = 1$	$x_2 = 2$	$x_2 = 3$	$x_2 = 4$	$f_2^*(s_2)$	x_2^*
0	0					0	–
1	0	0				0	–
2	0	4/9	4/9			4/9	1 or 2
3	2/3	4/9	2/3	2/3		2/3	0,2, or 3
4	2/3	8/9	2/3	2/3	2/3	8/9	1
≥ 5	1					1	0

Solution Procedure: Backward induction, Stage 1

$$f_n(s_n, x_n) = \frac{1}{3} f_{n+1}^*(s_n - x_n) + \frac{2}{3} f_{n+1}^*(s_n + x_n)$$

$$f_n^*(s_n) = \max_{x_n \in \{0,1,\ldots,s_n\}} f_n(s_n, x_n)$$

Need: $f_1^*(s_1 = 3)$

s_2	$f_2^*(s_2)$
0	0
1	0
2	$4/9$
3	$2/3$
4	$8/9$
≥ 5	1

Starting with $s_1 = 3$ chips

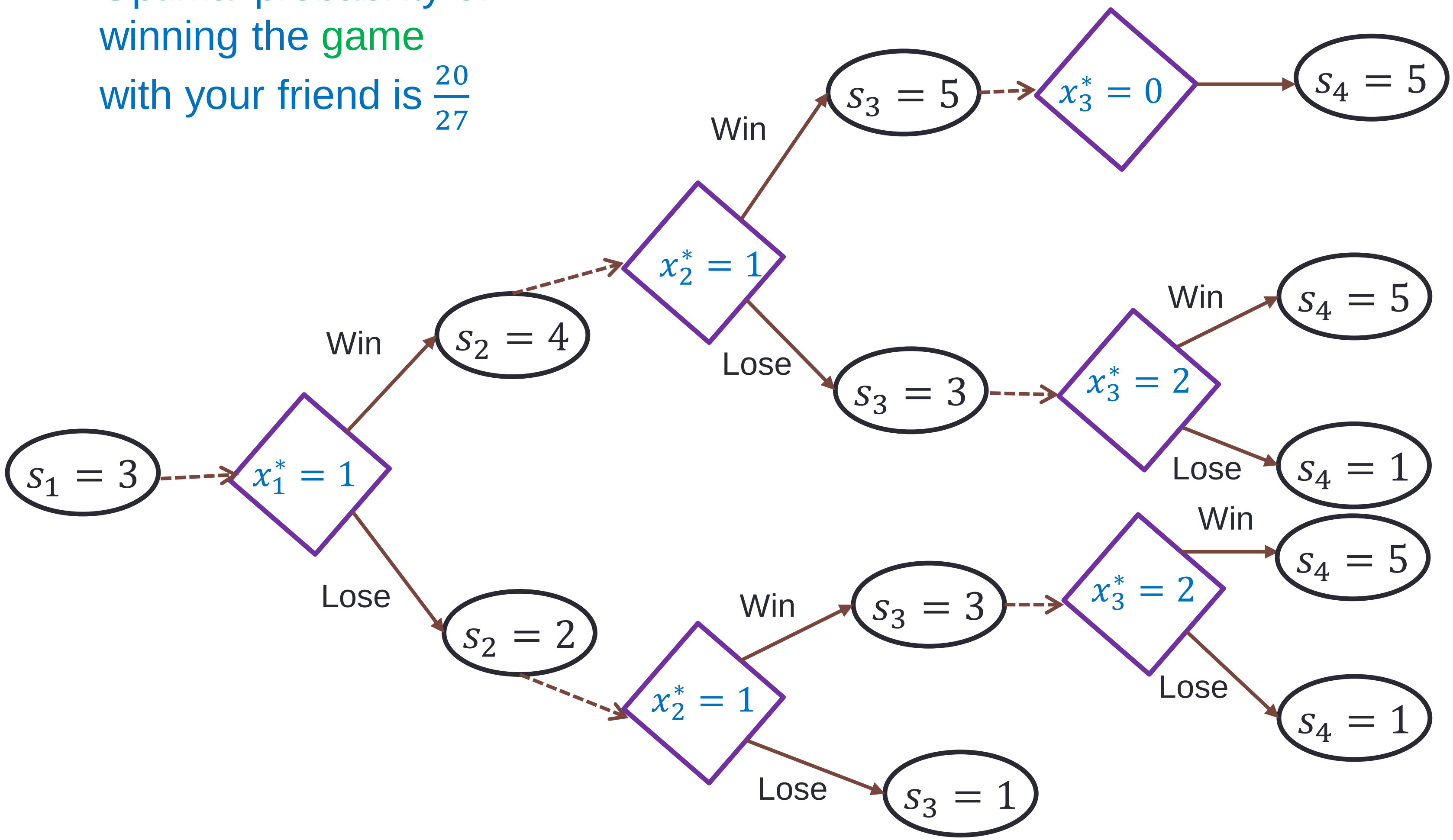
$$f_1(s_1 = 3, x_1) = \frac{1}{3} f_2^*(s_1 - x_1) + \frac{2}{3} f_2^*(s_1 + x_1)$$

$$f_1^*(s_1 = 3) = \max_{x_1 \in \{0,1,\ldots,s_1\}} f_1(s_1 = 3, x_1)$$

	$f_1(s_1 = 3, x_1)$ evaluated at					
s_1	$x_1 = 0$	$x_1 = 1$	$x_1 = 2$	$x_1 = 3$	$f_1^*(3)$	x_1^*
3	$2/3$	$20/27$	$2/3$	$2/3$	$20/27$	1

Solution Procedure: Optimal Policy

Optimal probability of winning the game with your friend is $\frac{20}{27}$



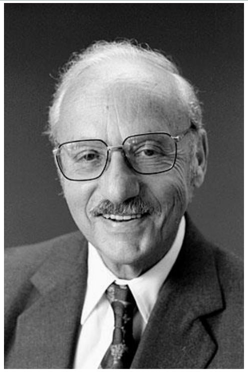
DP: Summary

- DP is a solution technique unlike LP which is a formulation
  - Starts with a small portion of the original problem
    - Finds optimal solution for this sub-problem
  - Gradually enlarges the problem
    - Finds the current optimal solution from the preceding one
  - Requires identifying a recursive relationship

- Useful for making a sequence of interrelated decisions
- Provides computational savings for very large problems
  - In run-time and storage-memory needed

NONLINEAR OPTIMIZATION

… where we see nonlinear optimization models and define convex functions



An anecdote on Linear Programming

Date: July 1948

A young and frightened George Dantzig, presents his newfangled "Linear Programming" to a meeting of the Econometric Society of Wisconsin, attended by distinguished scientists like Hotelling, Koopmans, and Von Neumann.

Following the lecture, Hotelling (in Dantzig's words, "a huge whale of a man") pronounced to the audience:

"But we all know the world is nonlinear!"

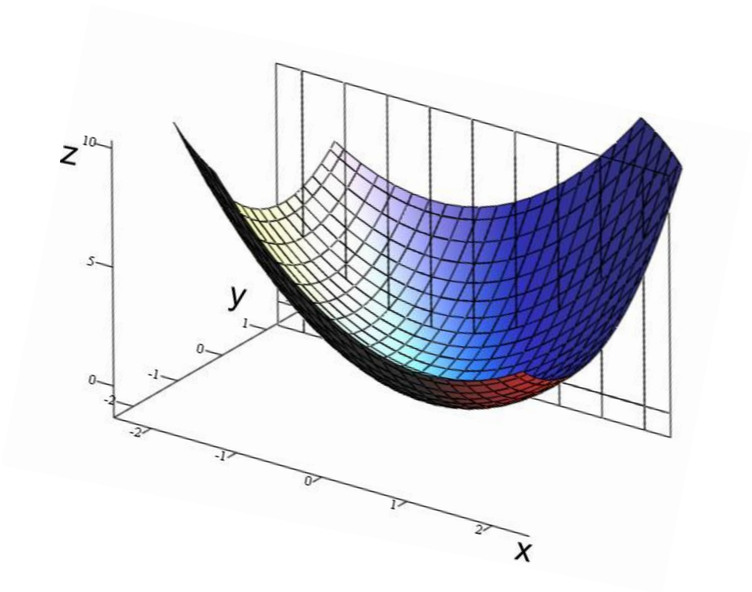
NLP: UNCONSTRAINED

Basic Concepts in NLP

Unconstrained nonlinear programming

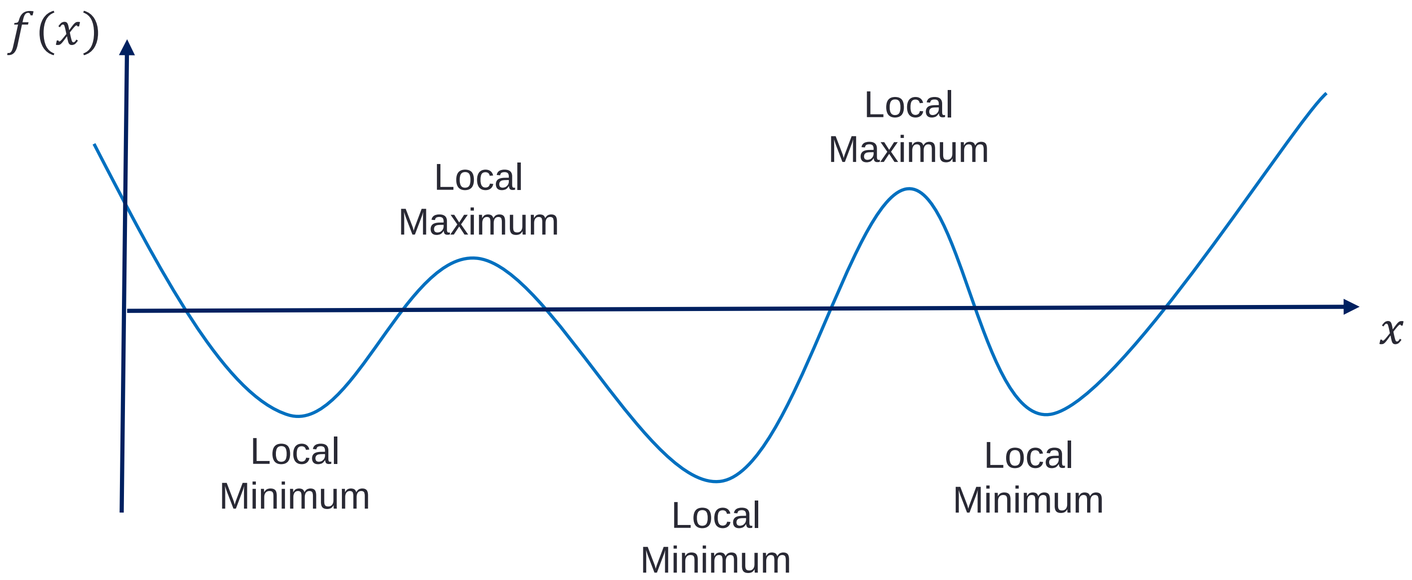
- Suppose that our mathematical program is of the form

$$\min f(\boldsymbol{x})$$
$$\boldsymbol{x} \text{ unrestricted}$$



Recall that $\max g(x)$ is equivalent to $\min -g(x)$, so we will only talk about $\min f(x)$ for the sake of following a convention

Local vs Global Minima/Maxima



What are the global minimum and maximum?

Local vs Global Minima

- **Local Minimum**:
  - A point is a local minimum
    if there are no better (feasible) solutions "nearby"
  - Formally: A point $\boldsymbol{x}^*$ is a local minimum if there exists some $\epsilon > 0$ such that $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all feasible $\boldsymbol{x}$ such that $\|\boldsymbol{x} - \boldsymbol{x}^*\| < \epsilon$

- **Global Minimum**:
  - A point is a global minimum
    if there are no better (feasible) solutions anywhere
  - Formally: A point $\boldsymbol{x}^*$ is a global minimum if $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all feasible $\boldsymbol{x}$

How to find local min/max for single var functions?

- Suppose that our mathematical program is of the form

$$\min f(x)$$
$$x \text{ unrestricted}$$

where $f(x)$ is a single var twice-differentiable function

- From calculus, we can find stationary points by solving for x from

$$\frac{df(x)}{dx} = 0$$

- If x^* is a stationary point, then

If $f''(x^*) > 0$, then x^* is a local minimum
If $f''(x^*) < 0$, then x^* is a local maximum
If $f''(x^*) = 0$, then x^* could be a local minimum or a local maximum
(or a saddle point, or an inflection point)

How to find stationary points for multivar functions?

- Suppose that our mathematical program is of the form

$$\min f(\boldsymbol{x})$$
$$\boldsymbol{x} \text{ unrestricted}$$

where $f(\boldsymbol{x})$ is a multi var differentiable function

- From calculus, we can find stationary points by solving for x from

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

where the gradient of f is defined as $\boldsymbol{\nabla f}(\boldsymbol{x}) := \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n}\right)$

- Checking if a stationary point is a local min/max is much harder

Local vs Global Minima/Maxima

- **Local Minimum**:
  - A point is a local minimum
    if there are no better (feasible) solutions "nearby"
  - Formally: A point $\boldsymbol{x}^*$ is a local minimum if there exists some $\epsilon > 0$ such that $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all feasible $\boldsymbol{x}$ such that $||\boldsymbol{x} - \boldsymbol{x}^*|| < \epsilon$

- **Global Minimum**:
  - A point is a global minimum
    if there are no better (feasible) solutions anywhere
  - Formally: A point $\boldsymbol{x}^*$ is a global minimum if $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all feasible $\boldsymbol{x}$

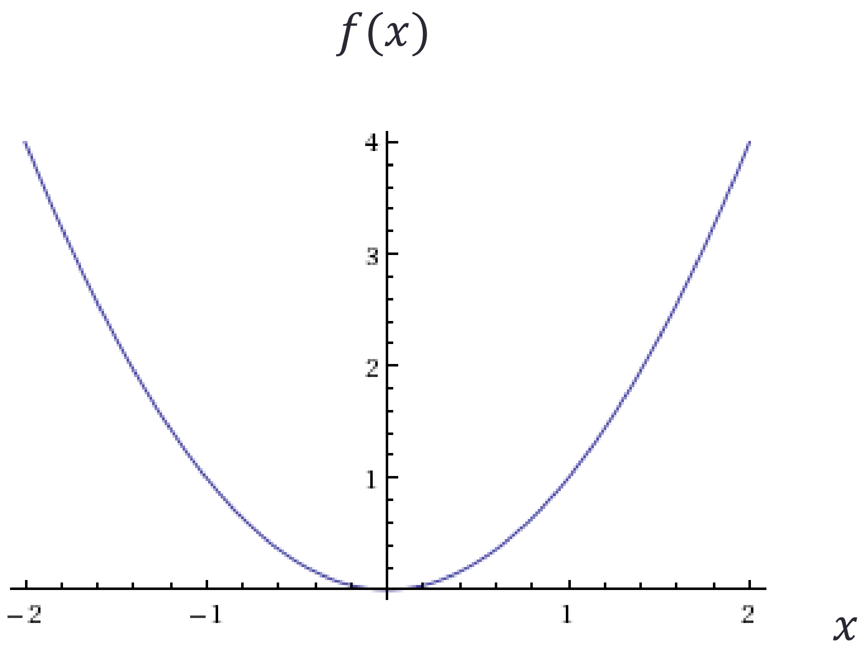
Question: Suppose we know that $\boldsymbol{x}^*$ is a stationary point.
Can we tell whether it is a global minimum?

Convexity helps!

- If $f(\boldsymbol{x})$ is a *convex function*, then any stationary point is a global minimum
- If $f(\boldsymbol{x})$ is a *concave function*, then any stationary point is a global maximum

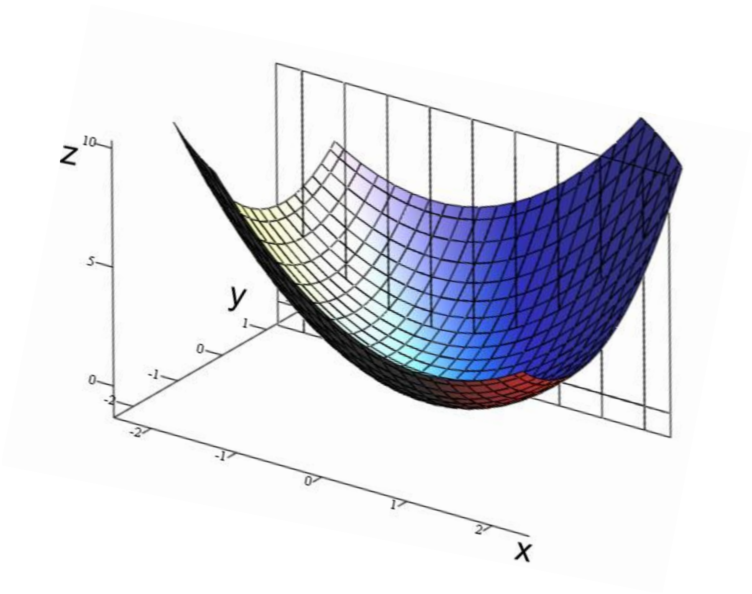
Convex Functions

- Every stationary point is a global minimum



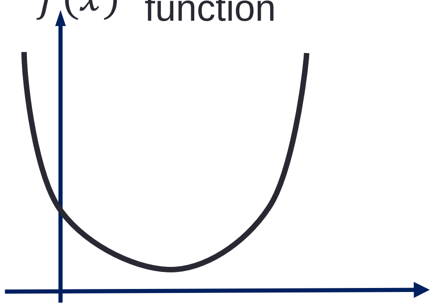
Convex Functions

- Every stationary point is a global minimum



x

Convex Functions



A function $f(\boldsymbol{x})$ is a

1. **convex function** if

   for each pair of points $\boldsymbol{a}$ and $\boldsymbol{b}$ and every $0 < \lambda < 1$:

$$f(\lambda\boldsymbol{a} + (1-\lambda)\boldsymbol{b}) \le \lambda f(\boldsymbol{a}) + (1-\lambda)f(\boldsymbol{b})$$

2. **strict convex function** if

   for each pair of points $\boldsymbol{a}$ and $\boldsymbol{b}$ and every $0 < \lambda < 1$:

$$f(\lambda\boldsymbol{a} + (1-\lambda)\boldsymbol{b}) < \lambda f(\boldsymbol{a}) + (1-\lambda)f(\boldsymbol{b})$$

3. **concave function** if

   for each pair of points $\boldsymbol{a}$ and $\boldsymbol{b}$ and every $0 < \lambda < 1$:

$$f(\lambda\boldsymbol{a} + (1-\lambda)\boldsymbol{b}) \ge \lambda f(\boldsymbol{a}) + (1-\lambda)f(\boldsymbol{b})$$

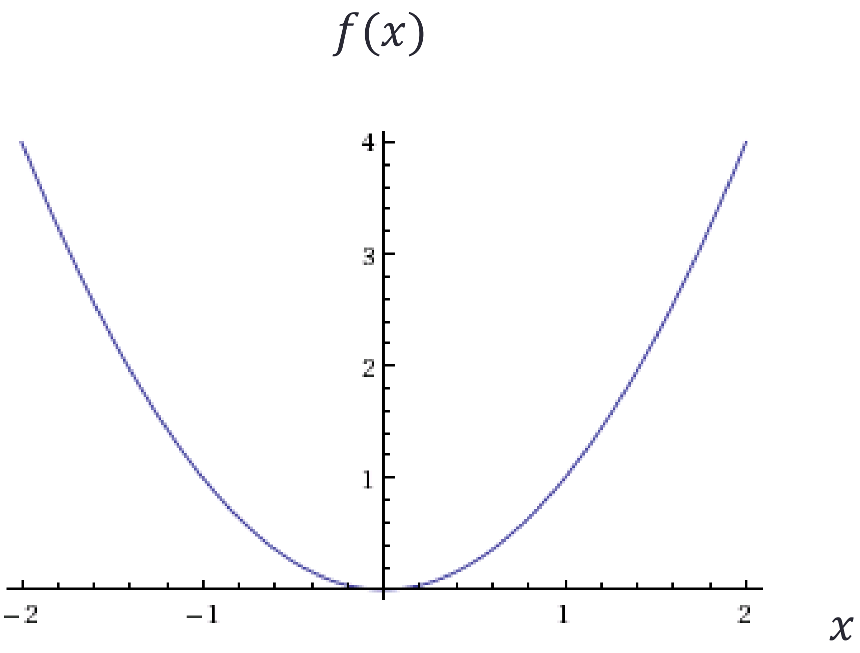
4. **strict concave function** if

   for each pair of points $\boldsymbol{a}$ and $\boldsymbol{b}$ and every $0 < \lambda < 1$:

$$f(\lambda\boldsymbol{a} + (1-\lambda)\boldsymbol{b}) > \lambda f(\boldsymbol{a}) + (1-\lambda)f(\boldsymbol{b})$$

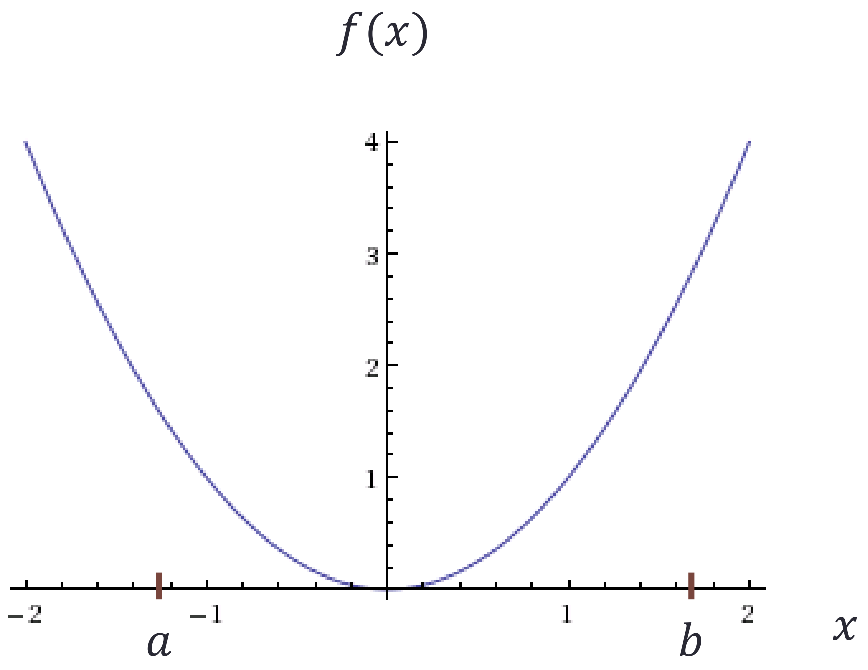
Convex Functions – Geometric Interpretation

- Consider the function $f(x) = x^2$



Convex Functions – Geometric Interpretation

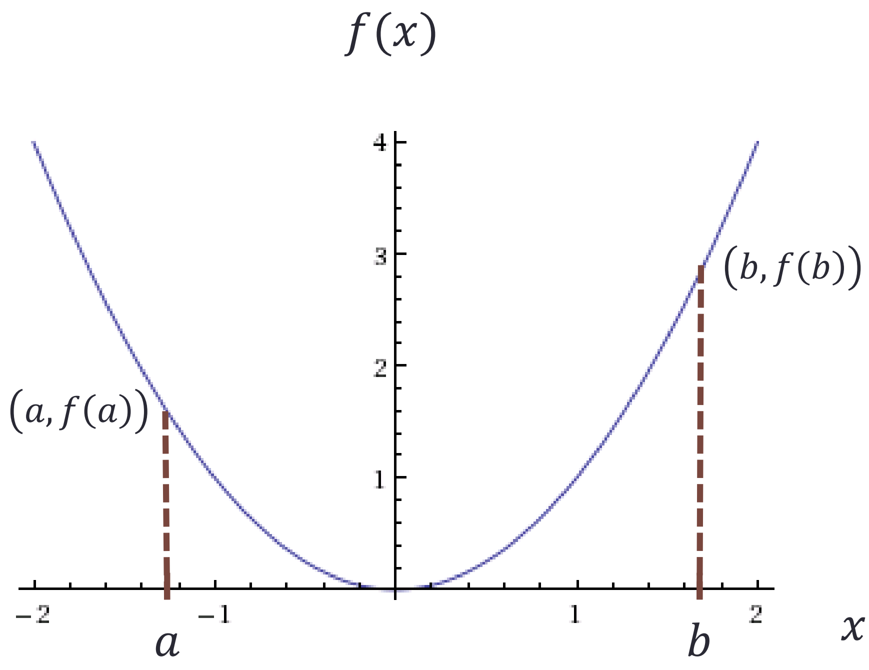
- Consider the function $f(x) = x^2$



Choose two points a and b (with $a < b$)

Convex Functions – Geometric Interpretation

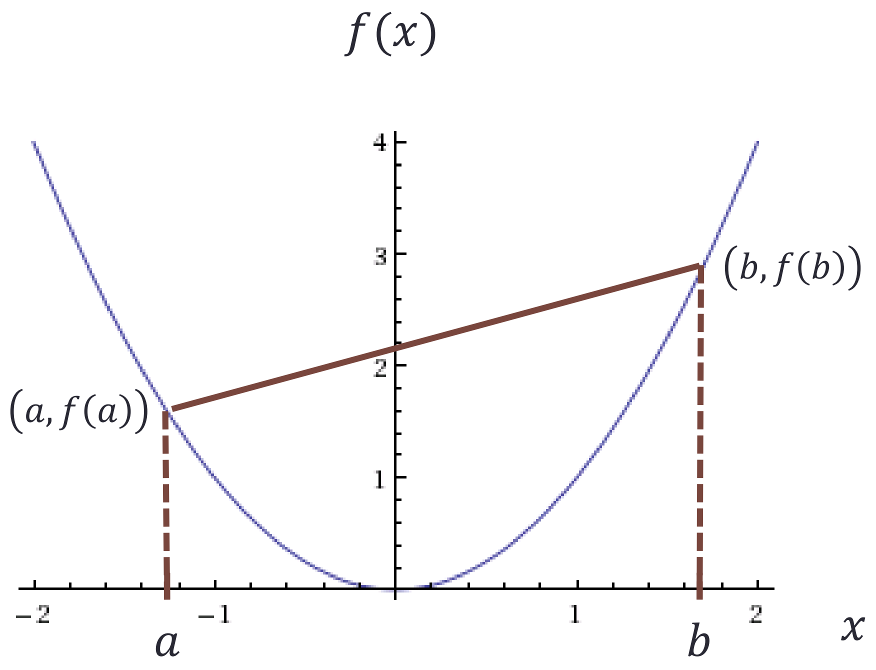
- Consider the function $f(x) = x^2$



Find the points $(a, f(a))$ and $(b, f(b))$

Convex Functions – Geometric Interpretation

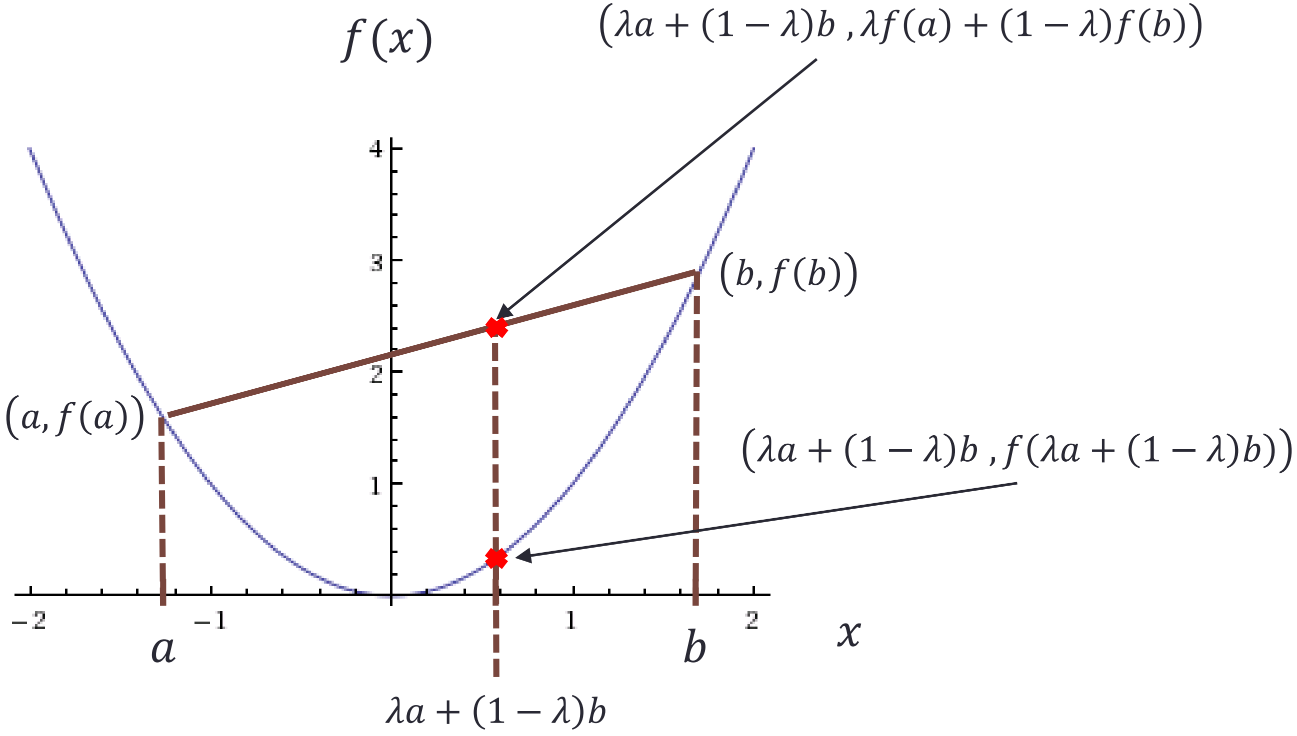
- Consider the function $f(x) = x^2$



The line joining $(a, f(a))$ and $(b, f(b))$ is never below the function value between a and b

Convex Functions – Geometric Interpretation

- Consider the function $f(x) = x^2$



The line joining $(a, f(a))$ and $(b, f(b))$ is never below the function value between a and b

Convex Functions – Geometric Interpretation

- Consider any two points a and b (with $a < b$), with $0 < \lambda < 1$
  - $\lambda a + (1 - \lambda)b$ is a point between a and b, weighted by λ
  - Suppose $\lambda = 0$, then it is point b
  - Suppose $\lambda = 1$, then it is point a
  - As λ increases from 0 to 1, the point given by $\lambda a + (1 - \lambda)b$ travels from b to a along a line

- For any value of $0 < \lambda < 1$, we have
  - $f(\lambda a + (1 - \lambda)b)$ is the value of $f(x)$ at this point
  - $\lambda f(a) + (1 - \lambda)f(b)$ is the weighted average of the values $f(a)$ and $f(b)$

- Convexity says that the weighted average is always at least as large as the function value
- Geometrically: The line joining the points $(a, f(a))$ and $(b, f(b))$ on the plane will never pass below $f(x)$ between a and b

Concave Functions

- Similarly:
  - For a concave function, the line joining the points $(a, f(a))$ and $(b, f(b))$ on the plane will never pass above $f(x)$

- Note: $f(x) = ax + b$ is concave and convex (but not strict)

Local vs Global Minima

- **Local Minimum**:
  - A point is a local minimum if there are no better (feasible) solutions "nearby"
  - Formally: A point $\boldsymbol{x}^*$ is a local minimum if there exists some $\epsilon > 0$ such that $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all feasible $\boldsymbol{x}$ such that $\|\boldsymbol{x} - \boldsymbol{x}^*\| < \epsilon$

- **Global Minimum**:
  - A point is a global minimum if there are no better (feasible) solutions anywhere
  - Formally: A point $\boldsymbol{x}^*$ is a global minimum if $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all feasible $\boldsymbol{x}$

Question: Suppose we know that $\boldsymbol{x}^*$ is a stationary point.
Can we tell whether it is a global minimum?

Convexity helps!

- If $f(\boldsymbol{x})$ is a *convex function*, then any stationary point is a global minimum
- If $f(\boldsymbol{x})$ is a *concave function*, then any stationary point is a global maximum

HOW TO VERIFY CONVEXITY?

- By definition (often hard)
- Derivative tests

How to verify convexity for single var functions?

- For a single variable twice differentiable function $f(x)$
  - If $f''(x) \geq 0$ for all x, then $f(x)$ is a convex function
  - If $f''(x) > 0$ for all x, then $f(x)$ is a strictly convex function
  - If $f''(x) \leq 0$ for all x, then $f(x)$ is a concave function
  - If $f''(x) < 0$ for all x, then $f(x)$ is a strictly concave function

Single Var Convex Functions

- Consider $f(x) = x^2$
- $f'(x) = 2x$
- $f''(x) = 2 \quad > 0$ for all x
- So $f(x)$ is convex

- Exercise: verify that for $0 \leq \lambda \leq 1$ and any two points a and b, we have
$$f(\lambda a + (1 - \lambda)b) \leq \lambda f(a) + (1 - \lambda)f(b)$$

How to verify convexity for multi var functions?

- For a single variable twice differentiable function $f(x)$

  If $f''(x) \geq 0$ for all x, then $f(x)$ is a convex function
  If $f''(x) > 0$ for all x, then $f(x)$ is a strictly convex function
  If $f''(x) \leq 0$ for all x, then $f(x)$ is a concave function
  If $f''(x) < 0$ for all x, then $f(x)$ is a strictly concave function

- A multi variable twice-differentiable function $f(\boldsymbol{x})$ is convex if all eigenvalues of the **Hessian matrix** $\boldsymbol{\nabla^2 f(x)}$ are non-negative
  - The i-th row, j-th col entry of the *Hessian matrix* is obtained by

$$[\boldsymbol{\nabla^2 f(x)}]_{i,j} := \frac{\partial^2 f}{\partial x_i \partial x_j}$$

  - Recall: Eigenvalues of a matrix M are the roots of the equation

$$\det(M - yI) = 0$$

Multivariable Convex Functions

$$f(x_1, x_2) = x_1^2 + x_2^2$$

- The Hessian Matrix is

$$\boldsymbol{\nabla}^2 \boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$

- The eigenvalues are 2 and 2
- So $f(\boldsymbol{x})$ is convex