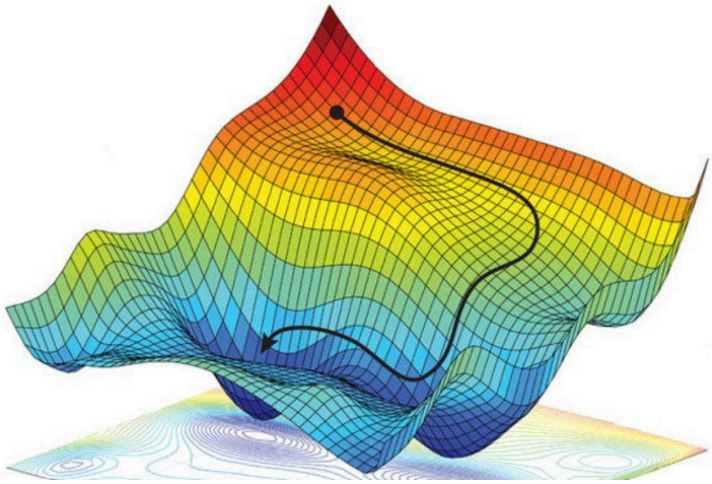
Calculus review

Topics

- Short review: Relevant mathematics (multi-variable calculus)
- Optimization fundamentals, Analytical optimization (pen and paper)
- Numerical optimization: Newton's method, secant method, finite difference, and Gradient descent
- Non-gradient based methods: random search, evolutionary, MC.



Motivation: Why should you care?

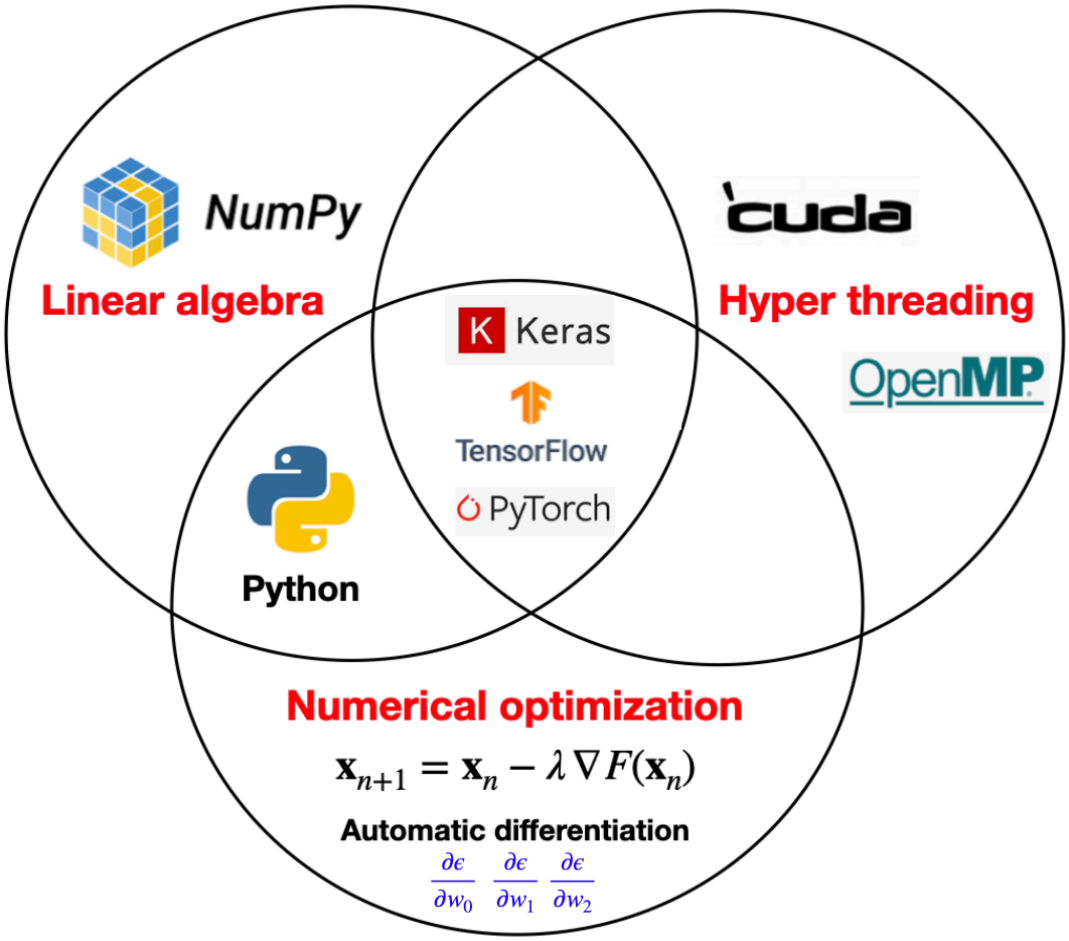
- The purpose of optimization is to achieve the "best" design relative to a set of prioritized criteria or constraints. These include maximizing factors such as productivity, strength, reliability, longevity, efficiency, and utilization. [1]
- Systematically finding optimal solutions to problems is better than "trial and error"
- The number of industries and companies utilizing mathematical optimization applications. According to a recent study conducted by Forrester Consulting, 37% of US managers (who are responsible for their company's data science strategy) say that they use mathematical optimization frequently in their work today.
- A HUGE amount of supervised machine learning can be condensed into a unified mathematical frame-work under the umbrella of mathematical optimization
- This is because the "training" (fitting) of all functional parametric models is essentially a mathematical optimization problem.

Motivation:

- Numerical optimization is the cornerstone of `modern deep learning`; linear and logistic regression, SVM, non-linear curve fitting, ALL artificial neural network models (MLP, CNN, RNN, LSTM, LLM, etc), most time-series

models (ARIMA, VAR, etc)

- DL libraries exist to do optimization (and linear algebra) as efficiently as possible.



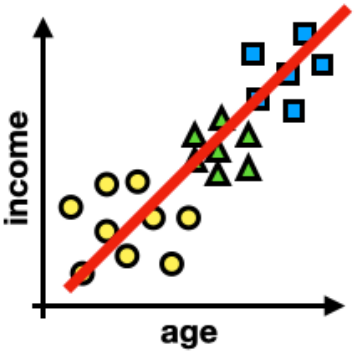
Single-variable functions

Review Content: Don't cover in class, students can review this at home.

Single variable functions: $y = f(x) \quad \mathbb{R}^1 \rightarrow \mathbb{R}^1$

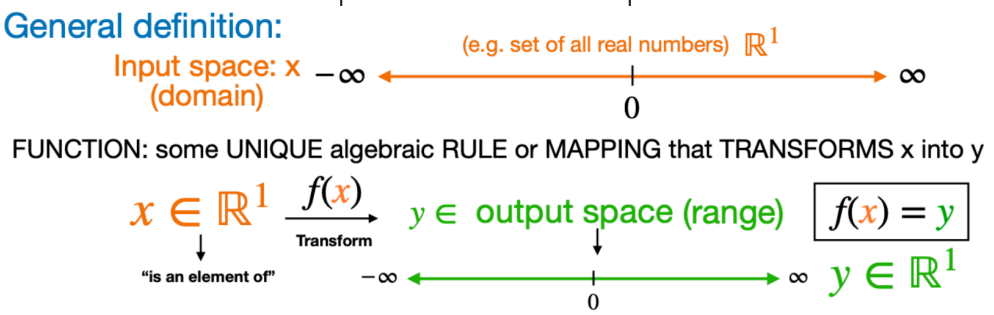
scalar input: x
scalar output: y

$$y = M(x \mid \mathbf{p}) = mx + b$$

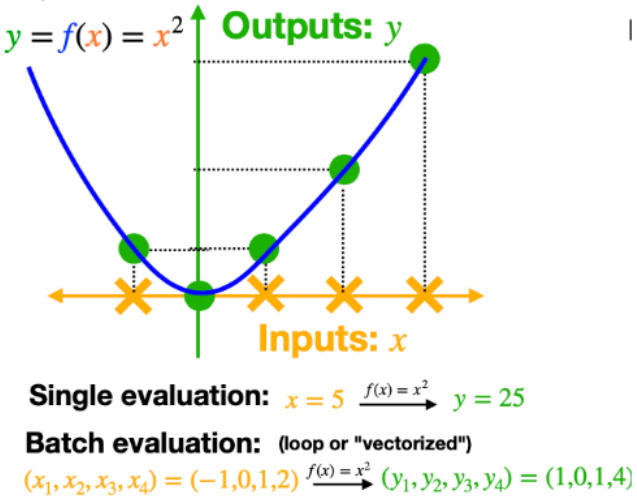


Single variable functions

- Functions are rules that map between numerical “spaces”



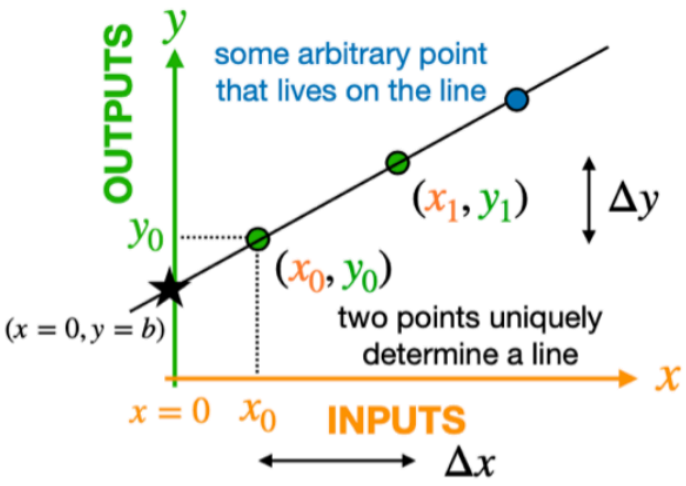
- In computers we can think of things in a “vectorized” fashion, i.e. compute an operation on all components in the vector at once, this is typically much more efficient



Important properties

- **single valued:** Is actually a function? i.e. does each input have only one output
- i.e does it pass the vertical line test
- **Linearity** Is it linear? If not what is the qualitative nature of it’s non-linearity
- Where is it defined? (domain=set of all x values)(range=set of possible y-values)
- **Periodicity:** Is it periodic (does it eventually just repeat itself)
- **Continuity:** Is it continuous (no breaks in the line)
- **Symmetries:** Are there any symmetries/invariant quantities
- Rotation, inflection, translation, etc • What is its slope as each point
- **Monotonicity:** Is it monotonic? i.e. only increasing or only decreasing
- **Concavity:** What is its Concavity: (convex or concave)
- **Differentiability:** Is it Differentiable? (does it have a well defined 1st, 2nd ... derivatives)

Linear functions



Two Parameter model:

slope $m = \frac{rise}{run} = \frac{\Delta y}{\Delta x} = \frac{y_1 - y_0}{x_1 - x_0}$

y-Intercept: $x = 0 \rightarrow f(0) = b$

Common forms:

slope-intercept form:

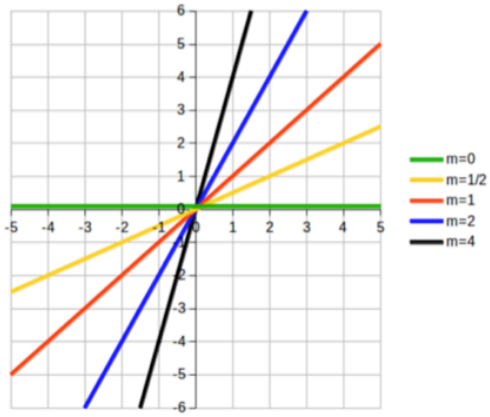
$$y = f(x) = mx + b \qquad \mathbf{P} = (m, b)$$

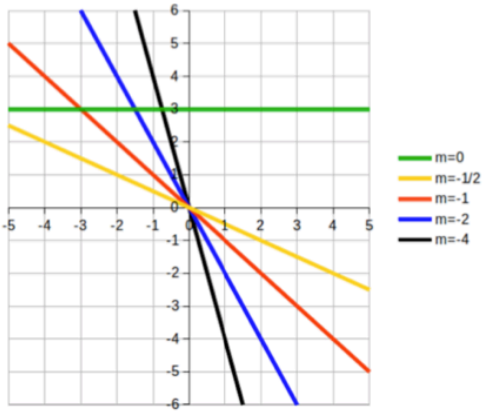
Point-slope form:

$$(y - y_0) = m(x - x_0) = \left(\frac{y_1 - y_0}{x_1 - x_0}\right)(x - x_0)$$

Standard form: $Ax + By = C$

Meaning of slope:





Function parameterization

- This is VERY important in parametric statistical learning.
- **FOR PARAMETRIC MODELS, THE FUNCTION PARAMETERS ARE WHAT WE LEARN!!**

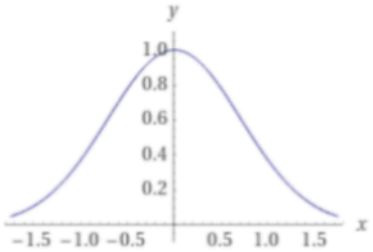
- Think of parameters as “knobs” we can turn to bend and shift the function “fit” the data

PARAMETRIC MODEL = Functional form + Parameterization

(underlying shape) (stretching and shifting)

$y = f(x) \rightarrow$ **"parent function"**

$$f(x) = e^{-x^2}$$



Parameterized form-1:

$$y = f(x \mid \mathbf{p}) = Af\left(\frac{x - x_o}{w}\right) + S$$

$$\mathbf{p} = (A, x_o, w, S)$$

$A =$ **verticle stretch (amplitude)**

$x_o =$ **horizontal shift (recentering parameter)**

$w =$ **horizontal stretch (width parameter)**

$S =$ **verticle shift (shift parameter)**

Parameterized form-2: $y = f(x \mid \mathbf{p}) = Af(w_1 x + b) + S$

(weights and bias) $w_1 = \frac{1}{w}$ **and** $b = -\frac{x_o}{w}$

Function transformations

Transformation Rules for Functions		
Function Notation	Type of Transformation	Change to Coordinate Point
f(x) + d	Vertical translation up d units	(x, y) → (x, y + d)
f(x) − d	Vertical translation down d units	(x, y) → (x, y − d)
f(x + c)	Horizontal translation left c units	(x, y) → (x − c, y)
f(x − c)	Horizontal translation right c units	(x, y) → (x + c, y)
−f(x)	Reflection over x-axis	(x, y) → (x, −y)
f(−x)	Reflection over y-axis	(x, y) → (−x, y)
af(x)	Vertical stretch for |a| > 1	(x, y) → (x, ay)
	Vertical compression for 0 < |a| < 1	
f(bx)	Horizontal compression for |b| > 1	$(x, y) \to \left(\frac{x}{b}, y\right)$
	Horizontal stretch for 0 < |b| < 1	

Rigid

Non-Rigid

https://www.onlinemathlearning.com/parent-functions.html

Parameterized form-1:

$$y = f(x \mid \mathbf{p}) = Af\left(\frac{x - x_o}{w}\right) + S$$

$$\mathbf{p} = (A, x_o, w, S)$$

A = verticle stretch (amplitude)

x_o = horizontal shift (recentering parameter)

w = horizontal stretch (width parameter)

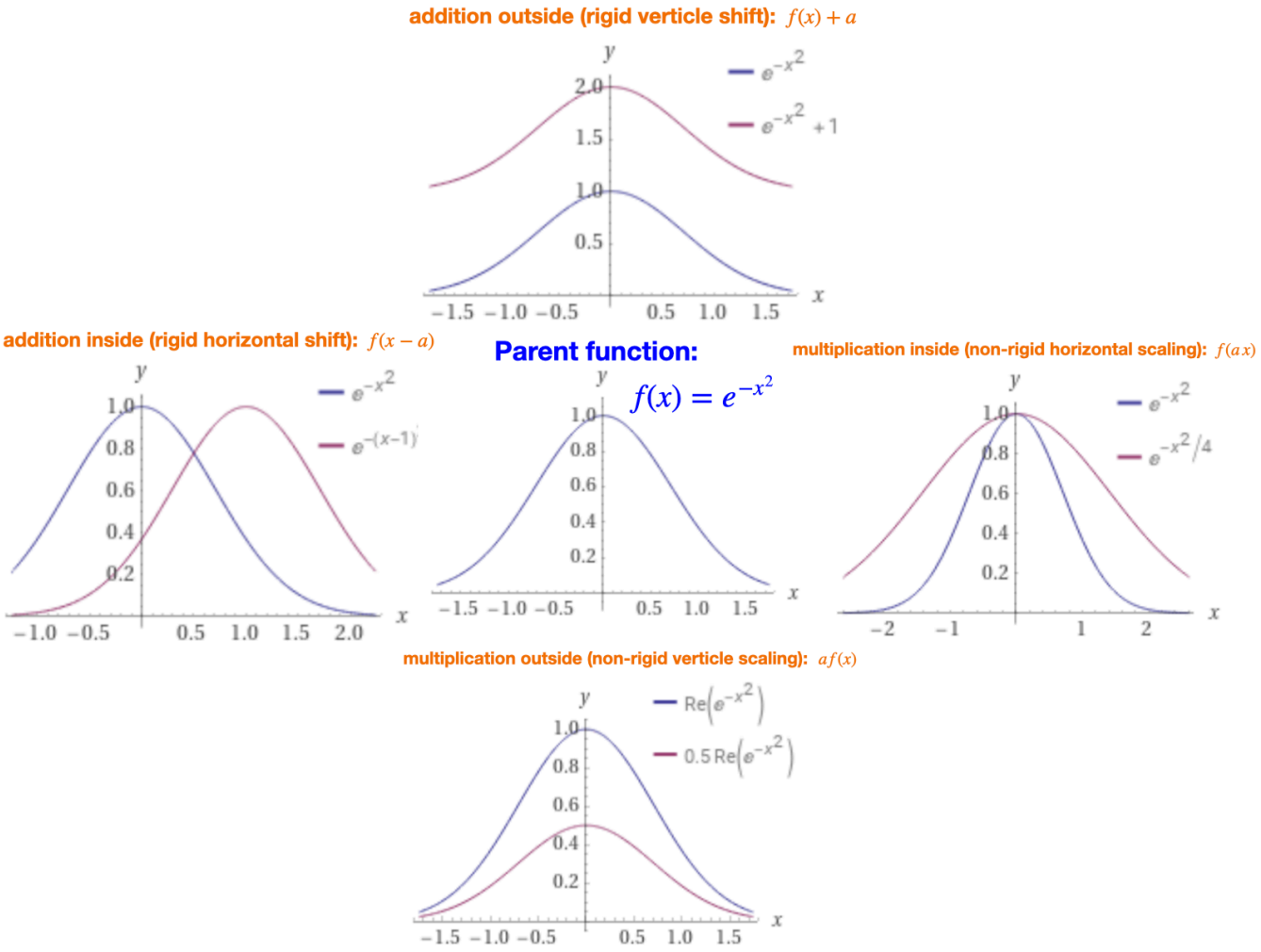
S = verticle shift (shift parameter)

Parameterized form-2: $y = f(x \mid \mathbf{p}) = Af(w_1 x + b) + S$

(weights and bias) $w_1 = \frac{1}{w}$ and $b = -\frac{x_o}{w}$

Example

- Demonstration of a function getting bent and shifted via parametrization



Additional Examples

- The following are common single variable parent functions

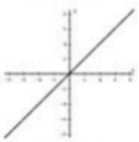
(underlying shape) (stretching and shifting) $\mathbf{p} = (A, x_o, w, S)$

$y = f(x) \rightarrow$ "parent function"

$$y = f(x \mid \mathbf{p}) = Af\left(\frac{x - x_o}{w}\right) + S$$

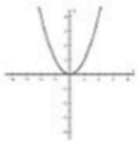
Linear

$$y = f(x \mid \mathbf{p}) = mx + b \quad \mathbf{p} = (m, b)$$



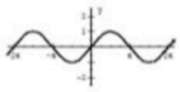
Quadratic

$$y = f(x \mid \mathbf{p}) = Ax^2 + Bx + C \quad \mathbf{p} = (A, B, C)$$



Sinusoidal

$$y = f(x \mid \mathbf{p}) = A_o \sin(\omega t + \phi) + c \quad \mathbf{p} = (A_o, \omega, \phi, c)$$

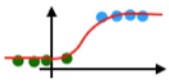


Exponential

$$y = f(x \mid \mathbf{p}) = ae^{\frac{(x - x_o)}{w}} + d \quad \mathbf{p} = (a, w, x_o, d)$$

$$y = f(x \mid \mathbf{p}) = \frac{A}{1 + e^{-\frac{(x - x_o)}{w}}} + s \quad \mathbf{p} = (A, w, x_o, S)$$

Logistic



$$y = f(x \mid \mathbf{p}) = A \cdot \exp^{-\left(\frac{x - x_o}{w}\right)^2} + S \quad \mathbf{p} = (A, x_o, w, S)$$

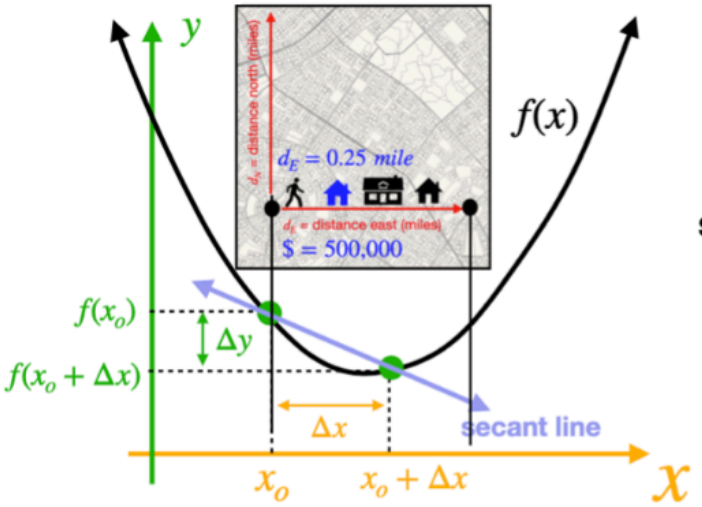
Gaussian

Single-variable Calculus

Refresher: Move through very quickly

Limit definition of derivative

- Differentiation finds the instantaneous slope of the function at a particular point x_0



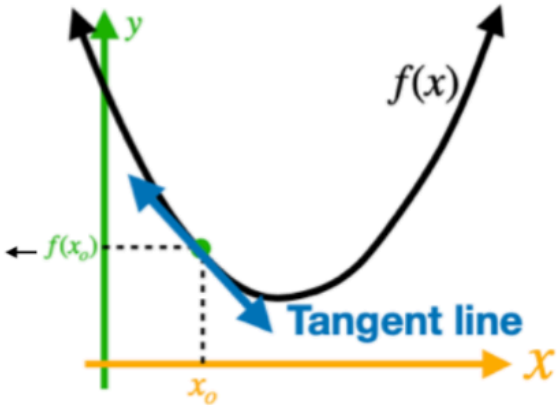
Slope of secant line

$$m = \frac{rise}{run} = \frac{\Delta y}{\Delta x} = \frac{f(x_o + \Delta x) - f(x_o)}{\Delta x}$$

↓

strink Δx till it's infinitally small

Secant line → Tangent line



Slope of tangent line

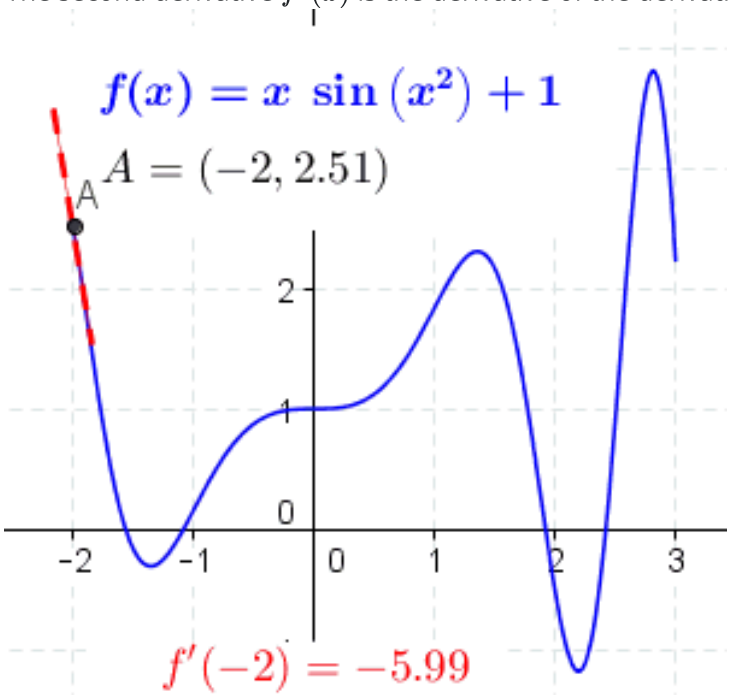
$$f'(x_o) = \frac{df}{dx} = \lim_{\Delta x \to 0} \frac{f(x_o + \Delta x) - f(x_o)}{\Delta x}$$

(Limit definition of derivative)

$f'(x_o)$ **= instantaneous rate of change of f(x) at** x_o

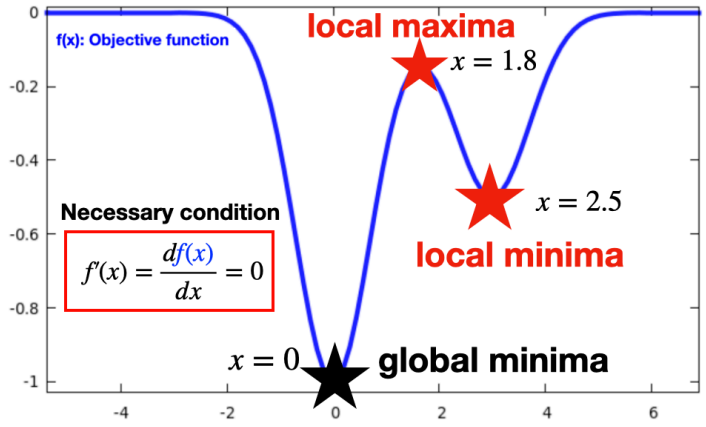
The derivative function

- The derivative $f'(x) = \frac{df}{dx}$ is a new function describing the slope of the first.
- It provides the slope of the tangent line of the original function at each point
  - I.e. you give me a some x_0 and I'll evaluate $f'(x_0)$ to give you the slope at x_0
- The second derivative $f''(x)$ is the derivative of the derivative



Stationary points

- Many properties of f can be determined by the derivatives f' and f''
- Stationary points, i.e. minima, maxima, and inflection points, refer to special points where the function does not change $f' = 0$ (i.e. horizontal tangent line).
- These can be either global (lowest or highest point) or local



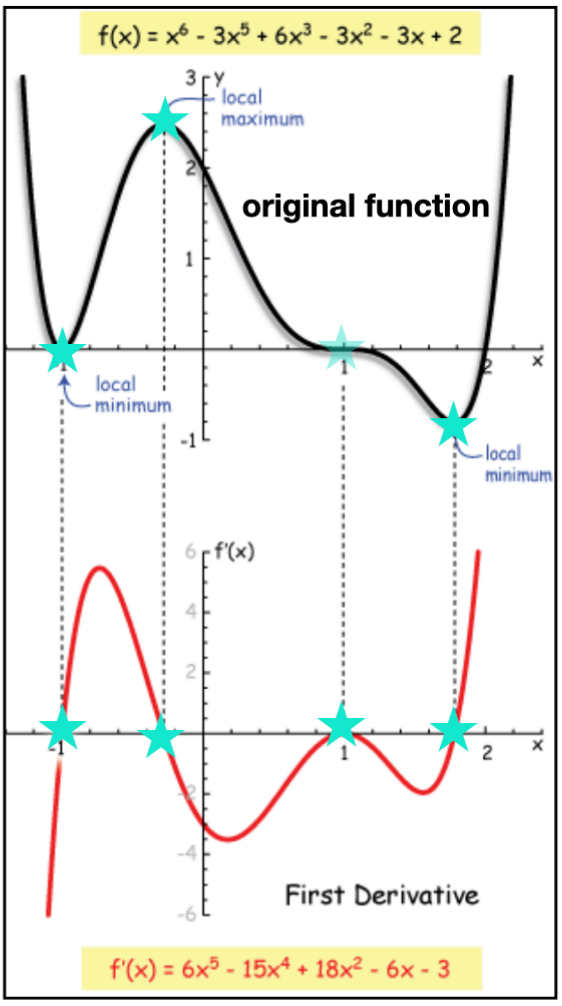
- **Optimization** attempts to find minima or maxima of some **objective function**.
- **IMPORTANT:** In regression, the objective function is known as the loss function L (or loss surface).

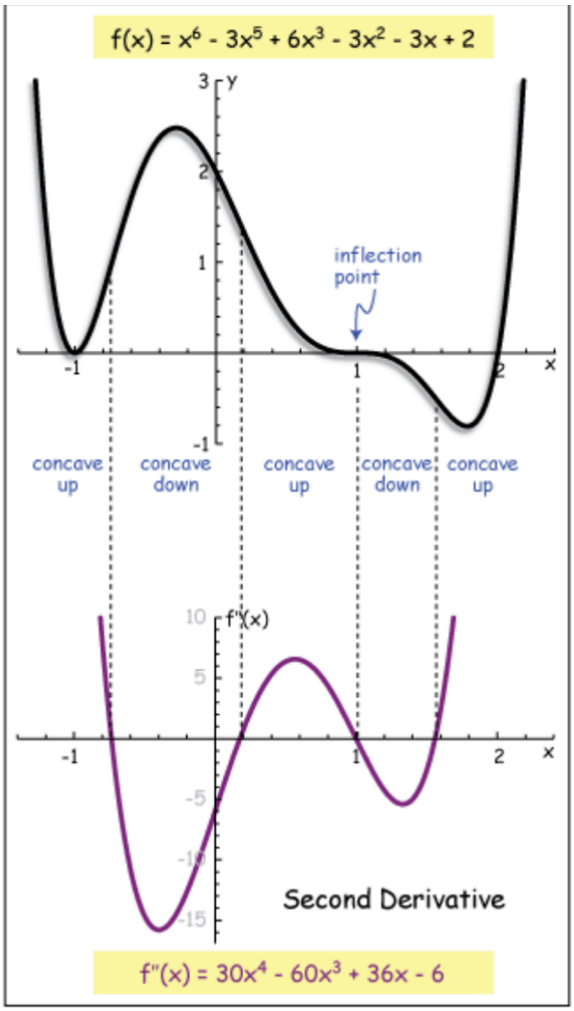
Stationary points

Stationary points

$$\frac{df}{dx} = 0$$

minima
maxima
constants





https://courses.lumenlearning.com/businesscalc1/chapter/reading-curve-sketching/

Concavity

- The property of concavity is particularly important in optimization

A concave function: no line segment joining two points on the graph lies above the graph at any point

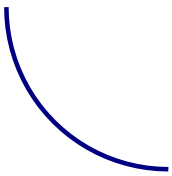
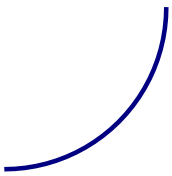
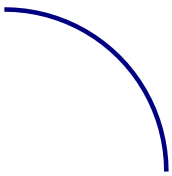
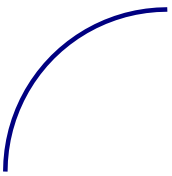
A convex function: no line segment joining two points on the graph lies below the graph at any point

A function that is neither concave nor convex: the line segment shown lies above the graph at some points and below it at others

- If f is strictly convex, it will have a single global minimum.
- If f is strictly concave, it has a single global maximum.
- In regression, the loss surface *typically* has a mix of convex and concave regions
- I.e. there are many local minima and maxima, which is what makes `training` difficult

Concavity and derivatives

- Demonstration of the relation between concavity and a functions derivative

	$f'(x) < 0$	$0 < f'(x)$
$0 < f''(x)$	 Here $f(x)$ is decreasing, while the rate of change of $f(x)$ is increasing. In this case the curve is **concave up**.	 Here $f(x)$ is increasing, while the rate of change of $f(x)$ is increasing. In this case the curve is **concave up**.
$f''(x) < 0$	 Here $f(x)$ is decreasing, while the rate of change of $f(x)$ is decreasing. In this case the curve is **concave down**.	 Here $f(x)$ is increasing, while the rate of change of $f(x)$ is decreasing. In this case the curve is **concave down**.

Multi-variable calculus

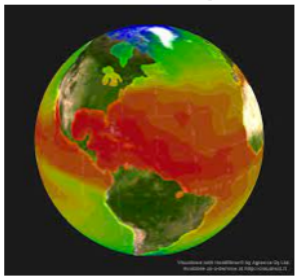
Refresher: Move through very quickly

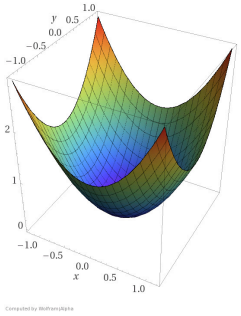
Multi-variable function types

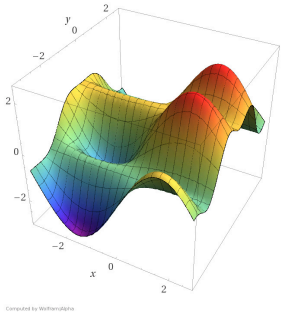
- **Scalar fields:** $y = f(\mathbf{x}) = f(x_1, x_2, \ldots, x_N) \quad \mathbb{R}^N \to \mathbb{R}^1$.
  - **IMPORTANT:** In regression, the objective function is called the loss function L (or loss surface). For parametric modeling, the loss surface is a scalar field over the model parameters. (e.g. $L(\mathbf{p})$ or $L(\mathbf{w})$ depending on the parameter notation).

vector input: x
scalar output: y

$T = T(\theta, \phi)$

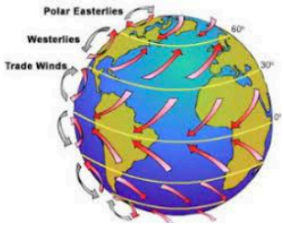


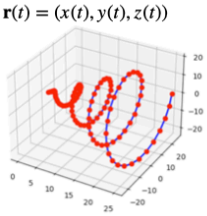




- **Vector fields:** $\mathbf{y} = f(\mathbf{x}) = \mathbf{f}(x_1, x_2, \ldots, x_N) = (y_1, y_2, \ldots, y_M) \quad \mathbb{R}^N \to \mathbb{R}^M$.
  - **IMPORTANT:** The loss function's gradient $\nabla L(\mathbf{w})$ in regression is a vector field.

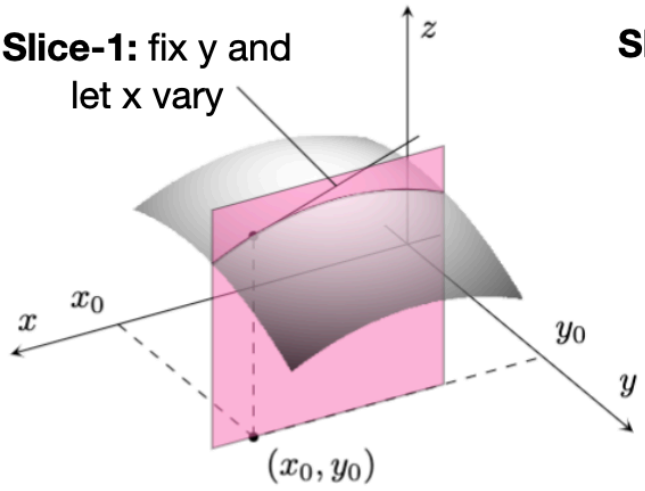
vector or scalar input: x
vector output: y

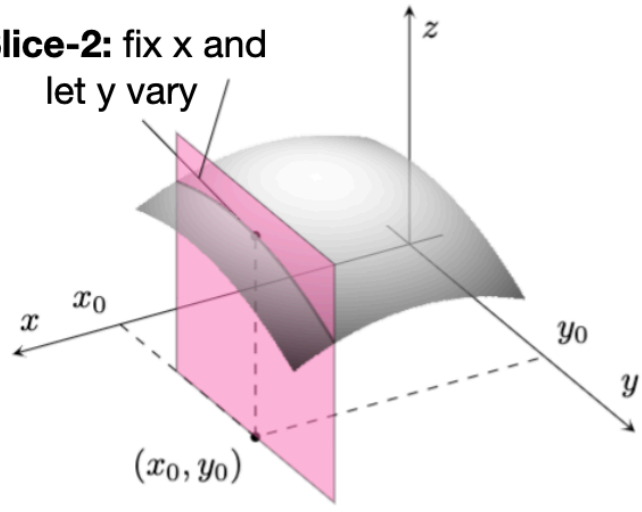




"Slicing" a multi-variable function

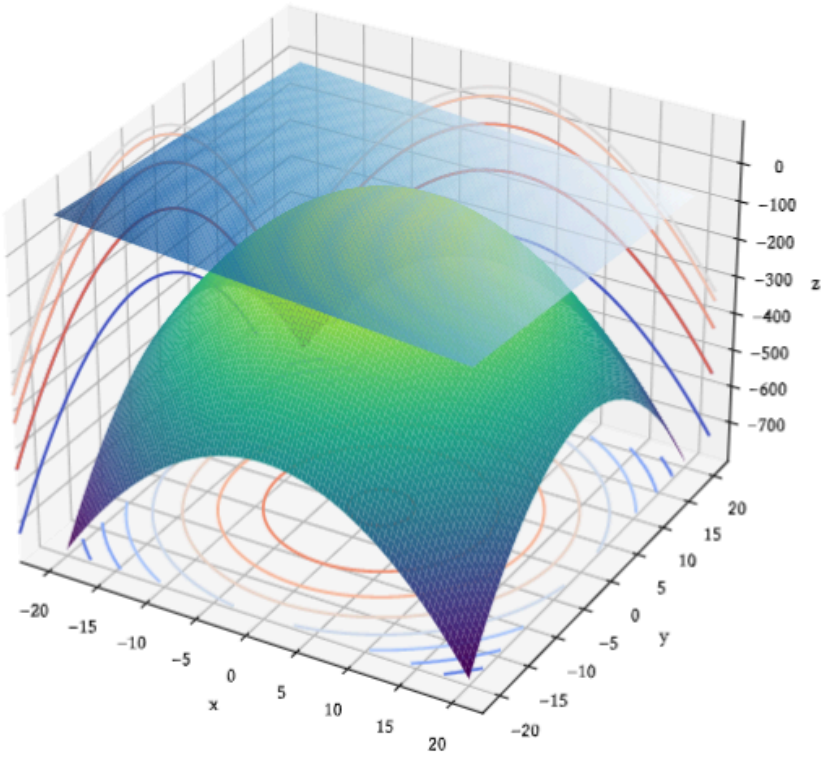
- Slicing is a very important concept in multi-variable calculus. It is when you "freeze" ALL variables, except one, and allow this variable to change independently
- This process converts a multi-variable scalar field $f(\mathbf{x}) = f(x_0, x_1 \ldots x_N)$, into a collection of single variable functions $f_1(x_1), f_2(x_2), \ldots f_N(x_N)$, in the neighborhood of some point in the multi-dimensional space $\mathbf{x}_0$. The various functions are single-variable because all other variables are "frozen" during their evaluation.





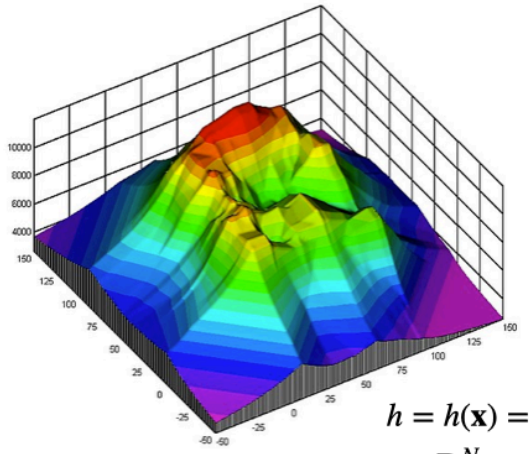
Contour plots-1

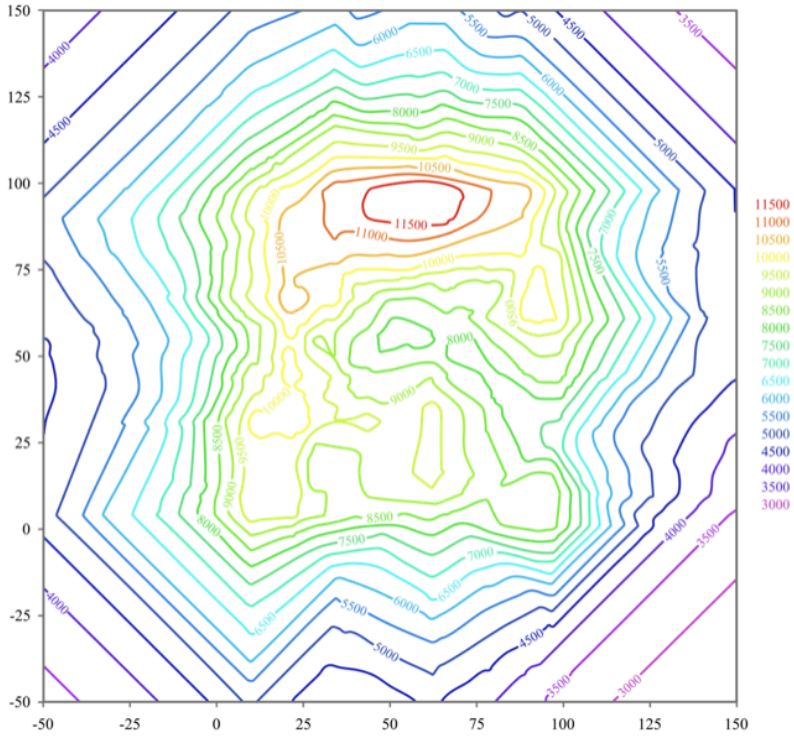
- Contour plots use slicing to create reduced dimensionality representations of higher dimensional data
- In this case, you freeze ONE variable, and visualize the effect of the variation in the others.



Contour plots-2

- Example of a scalar field with many local minima and maxima





$$h = h(\mathbf{x}) = h(x, y)$$

$$\mathbb{R}^N \rightarrow \mathbb{R}^1$$

Contour plot

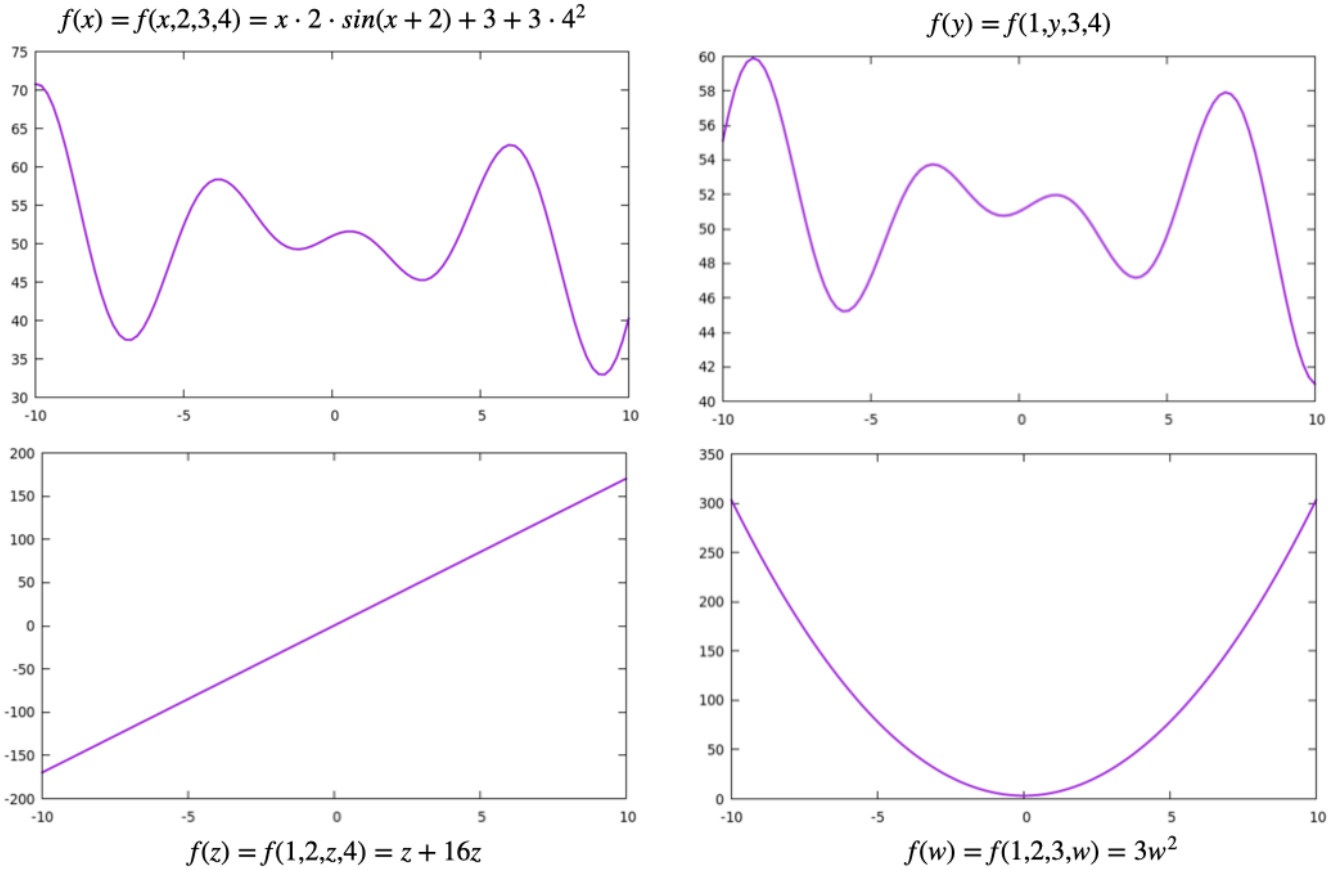
Fix h at constant intervals and let x and y vary

Example: slicing

- For example, consider a function describing the mapping from $\mathbb{R}^4 \rightarrow \mathbb{R}^1$
  $f(\mathbf{x}) = f(x, y, z, w) = x \cdot y \cdot sin(x + y) + z + z \cdot w^2$ with $\mathbf{x} \in \mathbb{R}^4$
- This can be "sliced" into FOUR functions about the point $\mathbf{x} = (x, y, z, w) = (1, 2, 3, 4)$
  - $f_1(x) = f(x, 2, 3, 4) = x \cdot 2 \cdot sin(x + 2) + 3 + 3 \cdot 4^2$
  - $f_2(y) = f(1, y, 3, 4) = 1 \cdot y \cdot sin(1 + y) + 3 + 3 \cdot 4^2$
  - $f_3(z) = f(1, 2, z, 4) = 1 \cdot 2 \cdot sin(1 + 2) + z + z \cdot 4^2$
  - $f_4(w) = f(1, 2, 3, w) = 1 \cdot 2 \cdot sin(1 + 2) + 3 + 3 \cdot w^2$
- Imagine "standing" on a "mountain" $y = f(\mathbf{x})$ in a 4D space at $\mathbf{x}_0$. The slices are like "looking" in each coordinate direction and observing the change in the mountain.

Example: slicing

- Below is a visualization of the slicing example from the previous slide.
- **Note:** This is vary powerful, it allows us to visualize a function of arbitrary dimensionality N, as a collection of N single variable functions

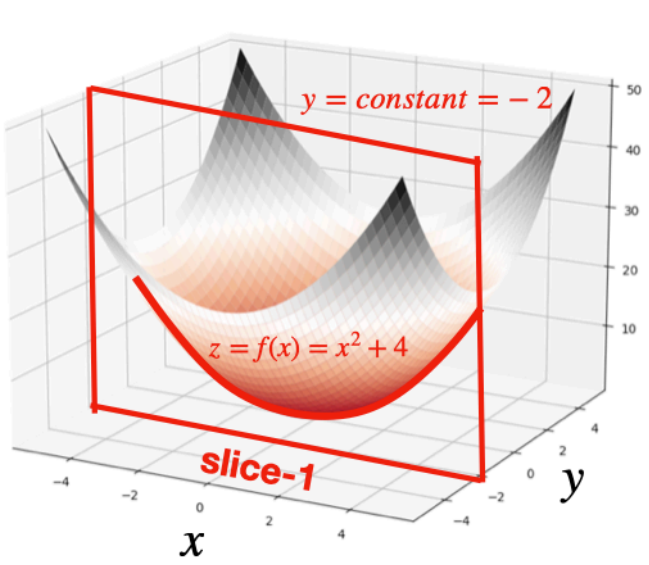


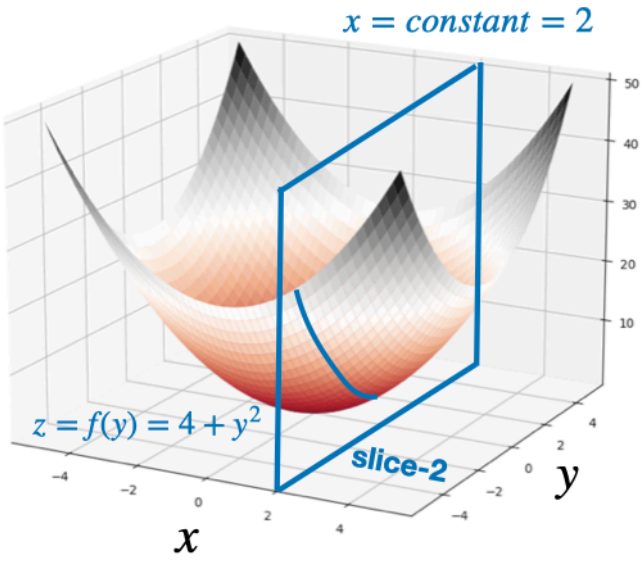
Partial derivative

- Slicing allows us to differentiate an N-dimensional scalar field in each of the coordinate directions about some point $\mathbf{x}_0$. Resulting in a collection of N partial-derivatives
- Partial differentiation applies single-variable calculus to each function from the slice
- In practice, we just treat the other variables as constant during the differentiation
- $f(\mathbf{x}) = f(x,y,z,w) = x \cdot y \cdot sin(x+y) + z + z \cdot w^2$
  - $f_x(\mathbf{x}) = \frac{\partial f}{\partial x} = y\,(xcos(x+y) + sin(x+y))$
  - $f_y(\mathbf{x}) = \frac{\partial f}{\partial y} = x\,(ycos(x+y) + sin(x+y))$
  - $f_z(\mathbf{x}) = \frac{\partial f}{\partial z} = 1 + w^2$
  - $f_w(\mathbf{x}) = \frac{\partial f}{\partial w} = 2z^2w$
  - Where we used the product rule $(uv)' = u'v + v'u$ on the first two
- These derivatives evaluated at $\mathbf{x} = (x,y,z,w) = (1,2,3,4)$ have the meaning of the slope of the tangent lines around $(1,2,3,4)$ on the four plots from the previous slide

Example: Partial differentiation

$z = f(x, y) = x^2 + y^2$ **consider point:** $(x, y) = (2, -2)$





Partial derivative: x
hold y as constant and take derivative with respect to x

$$f_x(x, -2) = \frac{\partial f}{\partial x} = \frac{\partial}{\partial x} f(x, -2)$$

$$= \frac{\partial}{\partial x}\left(x^2 + 4\right) = 2x + 0 = 2x$$

Partial derivative: y
hold x as constant and take derivative with respect to y

$$f_y(2, y) = \frac{\partial f}{\partial y} = \frac{\partial}{\partial y} f(2, y)$$

$$= \frac{\partial}{\partial y}\left(4 + y^2\right) = 0 + 2y = 2y$$

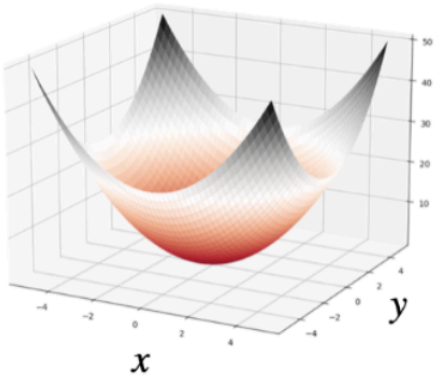
Partial derivatives at x=2 , y=-2

$$f_x(2, -2) = 2x\Big|_{x=2} = 4 \qquad f_y(2, -2) = 2y\Big|_{y=-2} = -4$$

The gradient

- The gradient is the vector of partial derivatives $\nabla f(\mathbf{x}) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}, \frac{\partial f}{\partial w}\right)$

$z = f(x, y) = x^2 + y^2$

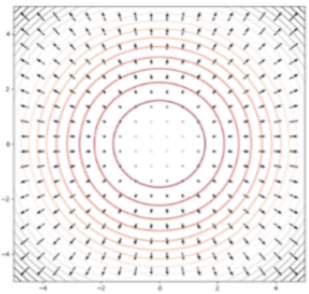


Partial derivatives can be calculated for any arbitrary x and y points

$$f_x(x, y) = \frac{\partial f}{\partial x} = \frac{\partial}{\partial x}\left(x^2 + y^2\right) = 2x + 0 = 2x$$

$$f_y(x, y) = \frac{\partial f}{\partial y} = \frac{\partial}{\partial y}\left(x^2 + y^2\right) = 0 + 2y = 2y$$

(Gradient Vector field)



Gradient —> vector of partial derivatives:

$$\nabla f(x, y) = \left(f_x(x, y), f_y(x, y)\right) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right) = (2x, 2y)$$

- **The gradient vector around a point $\mathbf{x}_0$ points in the direction of maximum increase of the function. This makes it incredibly useful for optimization**
- Gradient's magnitude is the rate of change in the direction of maximal increase

- The gradient can be thought of as an "operator" that acts on a function.

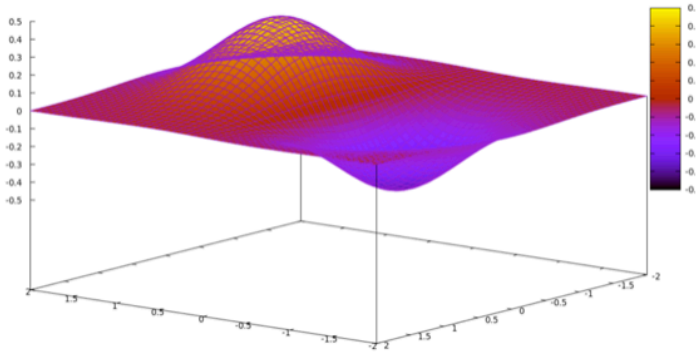
Gradient operator $\nabla = \begin{bmatrix} \frac{\partial}{\partial x_1} \\ \vdots \\ \frac{\partial}{\partial x_n} \end{bmatrix}$ → Acts on a scalar field $y = f(\mathbf{x}) = f(x_1, x_2, \ldots, x_N)$ → Gradient (Vector field) $\nabla f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$

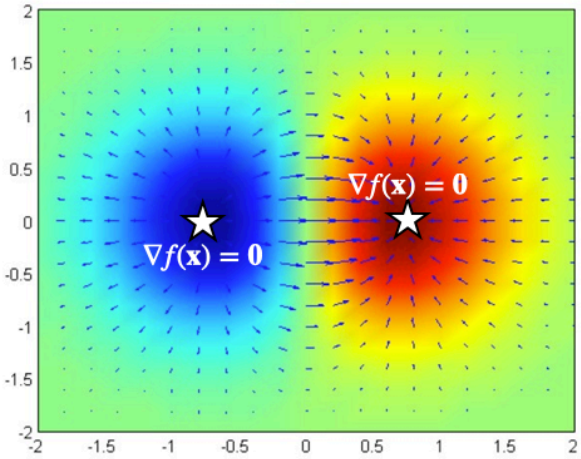
Another example

- Minima or maxima correspond to special $\mathbf{x}$ where **ALL gradient components are zero**
- This is a necessary condition for find stationary points $\nabla f(\mathbf{x}) = \mathbf{0} = (0, 0, \ldots, 0)$

$$f(x, y) = xe^{-(x^2+y^2)}$$

$$\nabla f(\mathbf{x}) = \Big(f_x(x, y), f_y(x, y)\Big) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)$$





(Gradient Vector field)

$$f_x(y, y) = \frac{\partial}{\partial x} f(x, y) = e^{-(x^2+y^2)}(1 - x\,(2x))$$

$$f_y(x, y) = \frac{\partial}{\partial y} f(x, y) = x\,(-2y)\,e^{-(x^2+y^2)}$$

product rule: $(u \cdot v)' = u' \cdot v + u \cdot v'$

Multi-variable Taylor Series

- To linearize general nonlinear systems, we will use the Taylor Series expansion of functions.
- Consider a function $f(x)$ of a single variable x, and suppose that $\bar{x}$ is a point such that $f(\bar{x}) = 0$. In this case, the point $\bar{x}$ is called an equilibrium point of the system $\dot{x} = f(x)$ since we have $\dot{x} = 0$ when $x = \bar{x}$ (i.e., the system reaches an equilibrium at $\bar{x}$).
- Taylor Series expansion of $f(x)$ around the point $\bar{x}$ is given by:

$$\mathcal{T}^1_{x_1} f(x) = f(\bar{x}) + (x_1 - x_2)^T \nabla f(x)$$

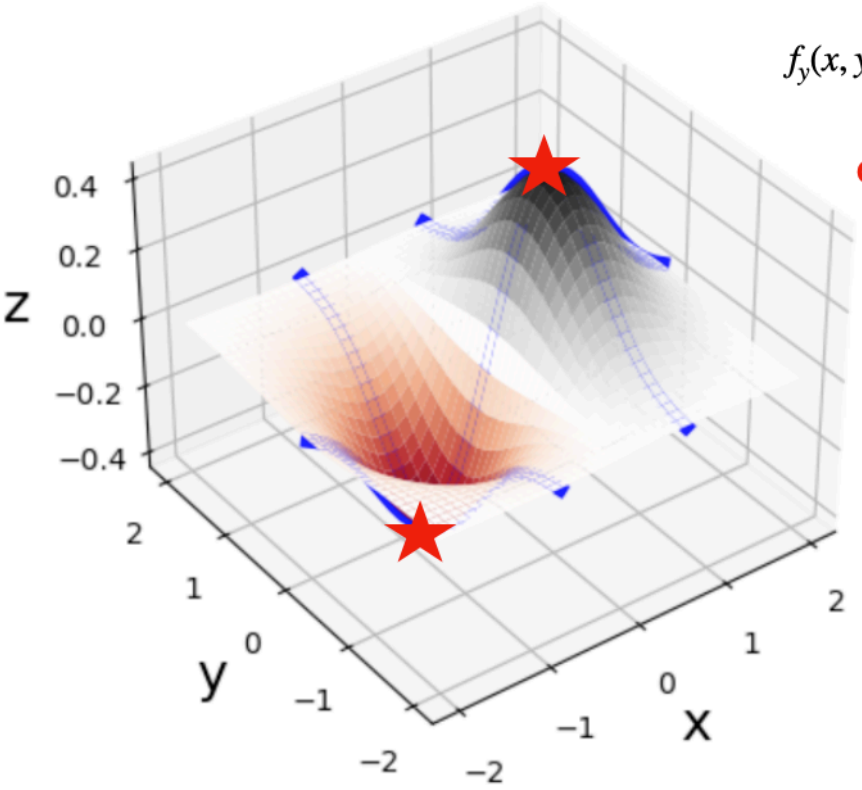
- Direction of the steepest descent is given by:

$$-\frac{\nabla f(x)}{\|\nabla f(x)\|_2}$$

- The steepest descent is in the direction of the negative gradient, $-\nabla f(x)$.

Multi-variable stationary points

$$f(x,y) = xe^{-(x^2+y^2)}$$



$$f_x(x,y) = \frac{\partial}{\partial x} f(x,y) = e^{-(x^2+y^2)}(1 - x\,(2x))$$

$$f_y(x,y) = \frac{\partial}{\partial y} f(x,y) = x\,(-2y)\,e^{-(x^2+y^2)}$$

Condition for minima and maxima

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

Both terms of gradient need to be zero

Find the roots Of a system of EQ

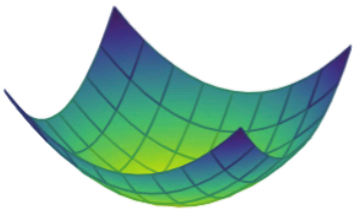
$$f_x(x,y) = 0$$

$$f_y(x,y) = 0$$

Multi-variable concavity

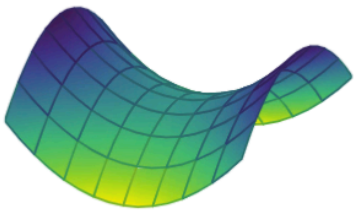
- Similar to the 1D case, the function's concavity control the type of stationary point

Strictly Convex



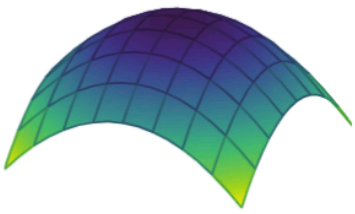
Global Minimum

Neither Concave nor Convex



Possibly a ***Saddle-Point*** or a Local Minimum / Maximum

Strictly Concave



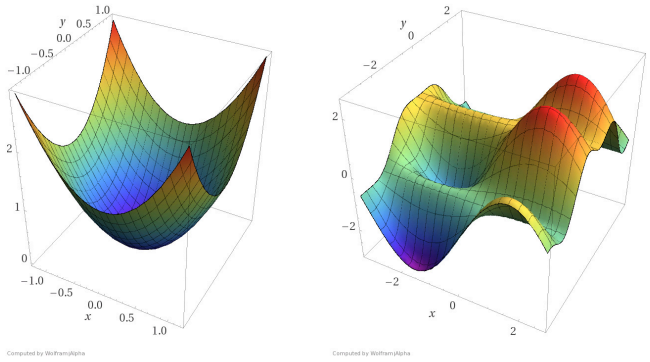
Global Maximum

Connection with regression

- Training a supervised parametric regression model is essentially the multi-variable optimization of a scalar field, known as the loss function.

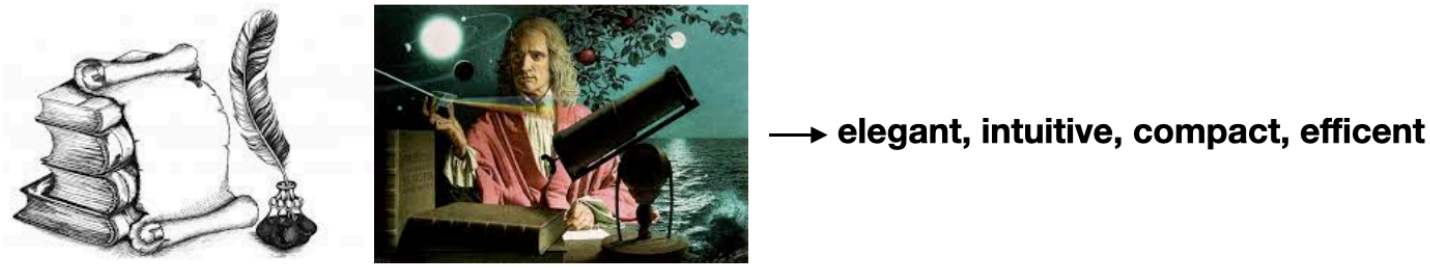
$$L = L(\mathbf{w}) = L(w_0, w_1, w_2 \dots w_N)$$

- We will use “w” as the dependent variable, because the inputs to the loss functions of the model parameters, often called weights (w) of the model
- The local or global minima or maxima of L satisfy the condition
  $\nabla L(\mathbf{w}) = \nabla L(\mathbf{w}) = (\frac{\partial L}{\partial w_0}, \frac{\partial L}{\partial w_1} \dots \frac{\partial L}{\partial w_N}) = 0$ (∇ = the vector of partial derivatives)
- **Possible loss surfaces**: for two parameter models

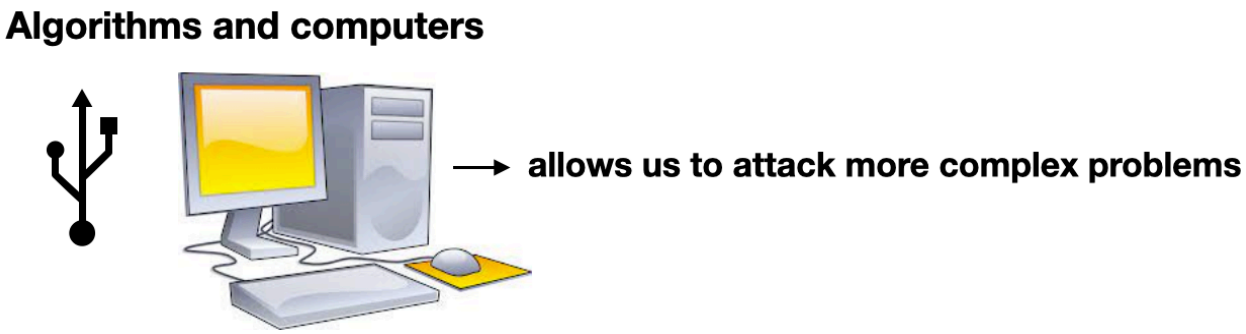


Mathematical paradigms

- Typically there are two options when it comes to doing mathematics
- **Analytical methods**: In this case means “by hand”, typically algebra or calculus

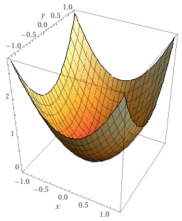


- **Numerical methods**: In this case means “by computer”, typically iterative “recipes”



Example: Analytic optimization

- Find the value of $\mathbf{x}_0$ that minimizes $f = f(\mathbf{x}) = f(x, y) = 5 + (x - 10)^2 + y^2$.



- This is the functional form of the loss function for single variable linear regression.
- It's such a simple function, we can infer the solution $\mathbf{x}_0 = (x_0, y_0) = (10, 0) \rightarrow f = 5$
- However, It can easily be proven using the condition $\nabla f(\mathbf{x}) = \mathbf{0} = (0, 0)$
  - $0 = \frac{\partial f}{\partial x} = 2(x - 10) \rightarrow x = 10 \qquad 0 = \frac{\partial f}{\partial y} = 2y \rightarrow y = 0$
- **This is a simple example, but the process is general**
  - (1): Compute the gradient and set it equal to zero
  - (2): Results in a system of N-equation and N-unknowns: $f_x(\mathbf{x}) = 0,\ f_y(\mathbf{x}) = 0 \ldots$
  - (3): `Solve` this system of equations to find the $\mathbf{x}_0$ (roots) which satisfies it

---

Footnotes

1. 2010, Optimization, an Important Stage of Engineering Design, Todd R. Kelley