AI VIETNAM
All-in-One Course

Selection of Functions for Different Targets

Quang-Vinh Dinh
PhD in Computer Science

Year 2023

AI VIETNAM
All-in-One Course

Reference

❖ **Read Chapter 1 for preparation**

Calculus Volume 1

SENIOR CONTRIBUTING AUTHORS
EDWIN "JED" HERMAN, UNIVERSITY OF WISCONSIN-STEVENS POINT
GILBERT STRANG, MASSACHUSETTS INSTITUTE OF TECHNOLOGY

openstax™

Link to download the book
https://www.dropbox.com/s/9njmu8wg09anex1/CalculusVolume1-OP.pdf?dl=0

Table of Contents

Outline

- Exponential Functions
- Logarithms
- Logarithm Applications
- Designing Functions for Evaluation
- Functions for Imbalanced Signals

Exponential Functions

❖ Definition

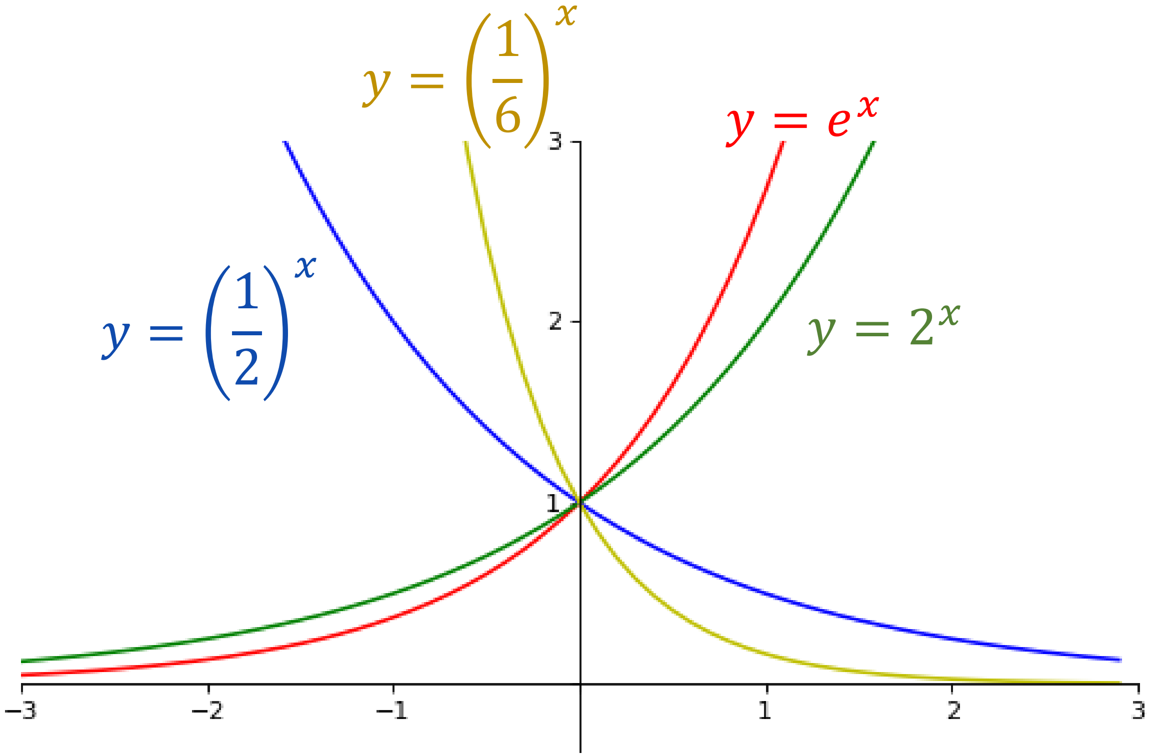
An exponential function can describe growth or decay

"Exponential functions model a relationship in which a constant change in the independent variable gives the same proportional change in the dependent variable."

https://en.wikipedia.org/wiki/Exponential_function

$$n \in \mathbb{N}$$

$$a^n = \underbrace{\mathrm{a} * \mathrm{a} * \cdots * \mathrm{a}}_{\text{n times}}$$



Exponential Functions

❖ Laws of Exponent

For any constant a > 0, b > 0, and for all x and y

$$a^x a^y = a^{x+y}$$

$$(ab)^x = a^x b^x$$

$$(a^x)^y = a^{xy}$$

$$\frac{a^x}{a^y} = a^{x-y}$$

$$a^x a^y = \underbrace{(\mathrm{a} * \cdots * \mathrm{a})}_{\text{x times}} * \underbrace{(\mathrm{a} * \cdots * \mathrm{a})}_{\text{y times}}$$

$$= \underbrace{(\mathrm{a} * \cdots * \mathrm{a}) * (\mathrm{a} * \cdots * \mathrm{a})}_{\text{x+y times}}$$

$$= a^{x+y}$$

$$a^x b^x = \underbrace{(\mathrm{a} * \cdots * \mathrm{a})}_{\text{x times}} * \underbrace{(\mathrm{b} * \cdots * \mathrm{b})}_{\text{x times}}$$

$$= \underbrace{(\mathrm{ab}) * \cdots * (\mathrm{ab})}_{\text{x times}}$$

$$= (\mathrm{ab})^x$$

Exponential Functions

❖ Laws of Exponent

For any constant a > 0, b > 0, and for all x and y

$$a^x a^y = a^{x+y}$$

$$(ab)^x = a^x b^x$$

$$(a^x)^y = a^{xy}$$

$$\frac{a^x}{a^y} = a^{x-y}$$

$$(a^x)^y = \left.\begin{matrix} \boxed{(a * \cdots * a)} \quad x\text{ times} \\ * \\ \cdots \\ * \\ \boxed{(a * \cdots * a)} \quad x\text{ times} \end{matrix}\right\} y\text{ times}$$

$$= a^{xy}$$

$$\frac{a^x}{a^y} = \frac{\boxed{a * \cdots * a} \quad x\text{ times}}{\boxed{a * \cdots * a} \quad y\text{ times}}$$

. . .

Exponential Functions

❖ Laws of Exponent

For any constant a > 0, b > 0, and for all x and y; p, q $\in \mathbb{Z}, q \neq 0$

$$a^{p/q} = \sqrt[q]{a^p}$$

$$\frac{a^x}{b^x} = \left(\frac{a}{b}\right)^x$$

$$a^{-x} = \frac{1}{a^x}$$

$$a^0 = 1$$

$$a = a^{q/q} = \left(a^{1/q}\right)^q$$

$$\Rightarrow a^{1/q} = \sqrt[q]{a}$$

$$a^{\frac{p}{q}} = a^{\frac{1}{q}*p} = \left(a^{1/q}\right)^p$$

$$= (a^p)^{1/q} = \sqrt[q]{a^p}$$

$$\frac{a^x}{b^x} = \frac{\boxed{a * \cdots * a}}{\boxed{b * \cdots * b}} \quad \begin{matrix}\text{x times}\\ \\ \text{x times}\end{matrix} \rightarrow \frac{a^x}{b^x} = \left(\frac{a}{b}\right)^x$$

$$a^{-x} = a^{-x}\frac{a^x}{a^x} = \frac{a^{x-x}}{a^x} = \frac{1}{a^x}$$

$$a^0 = a^{x-x} = a^x a^{-x} = \frac{a^x}{a^x} = 1$$

Exponential Functions

❖ Example

Initial population P_0

Growth at an annual rate of 6%

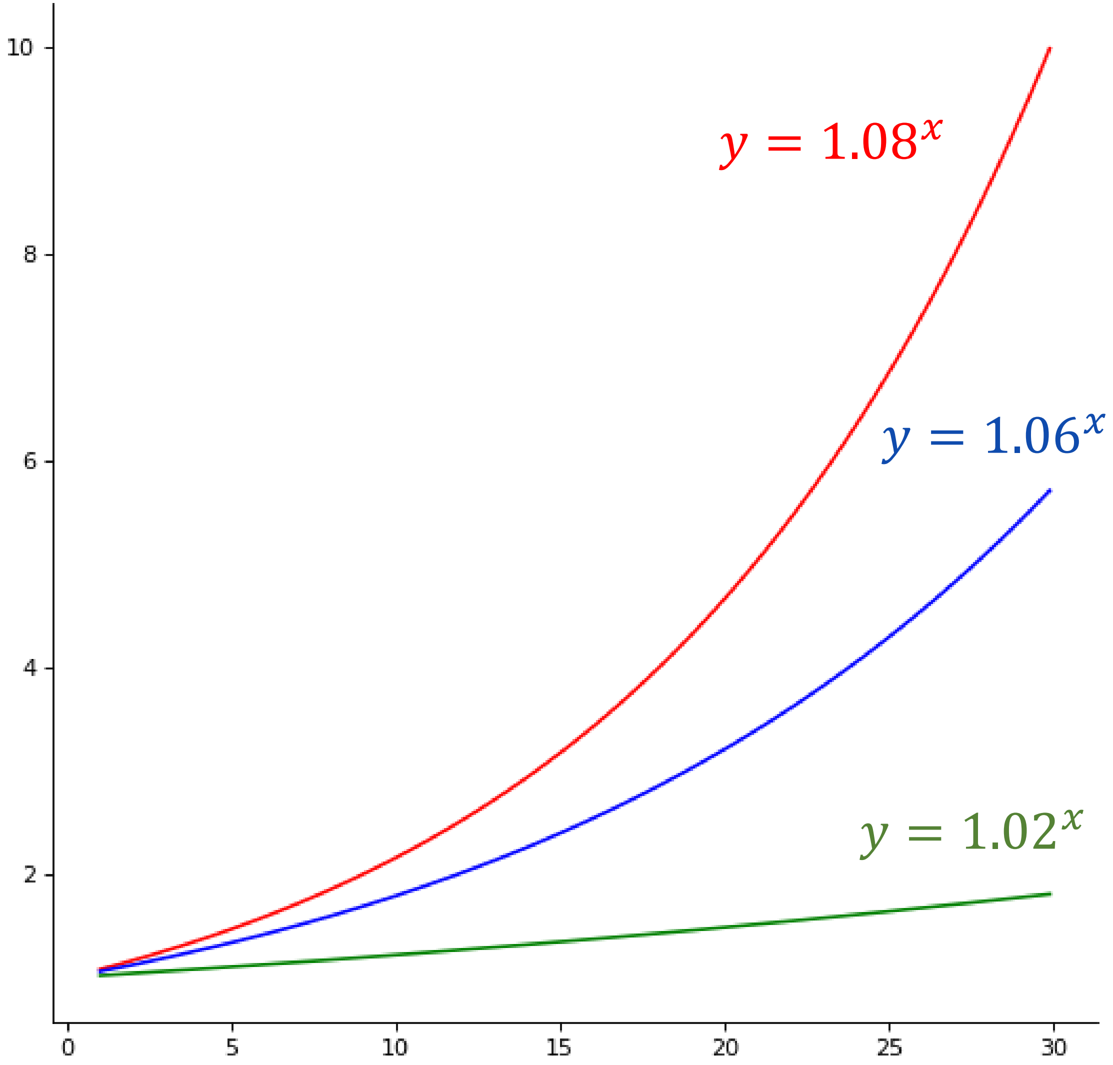
Population after 1 year is

$$P_1 = P_0 + 0.06P_0 = P_0(1 + 0.06) = P_0(1.06)$$

Population after 2 year is

$$P_2 = P_1 + 0.06P_1 = P_1(1.06) = P_0(1.06)^2$$

This example is from the reference book



Exponential Functions

❖ Example

This example is from the reference book

Suppose a particular population of bacteria is known to double in size every 4 hours. If a culture starts with 1000 bacteria, then

The number of bacteria after 4 hours = ?

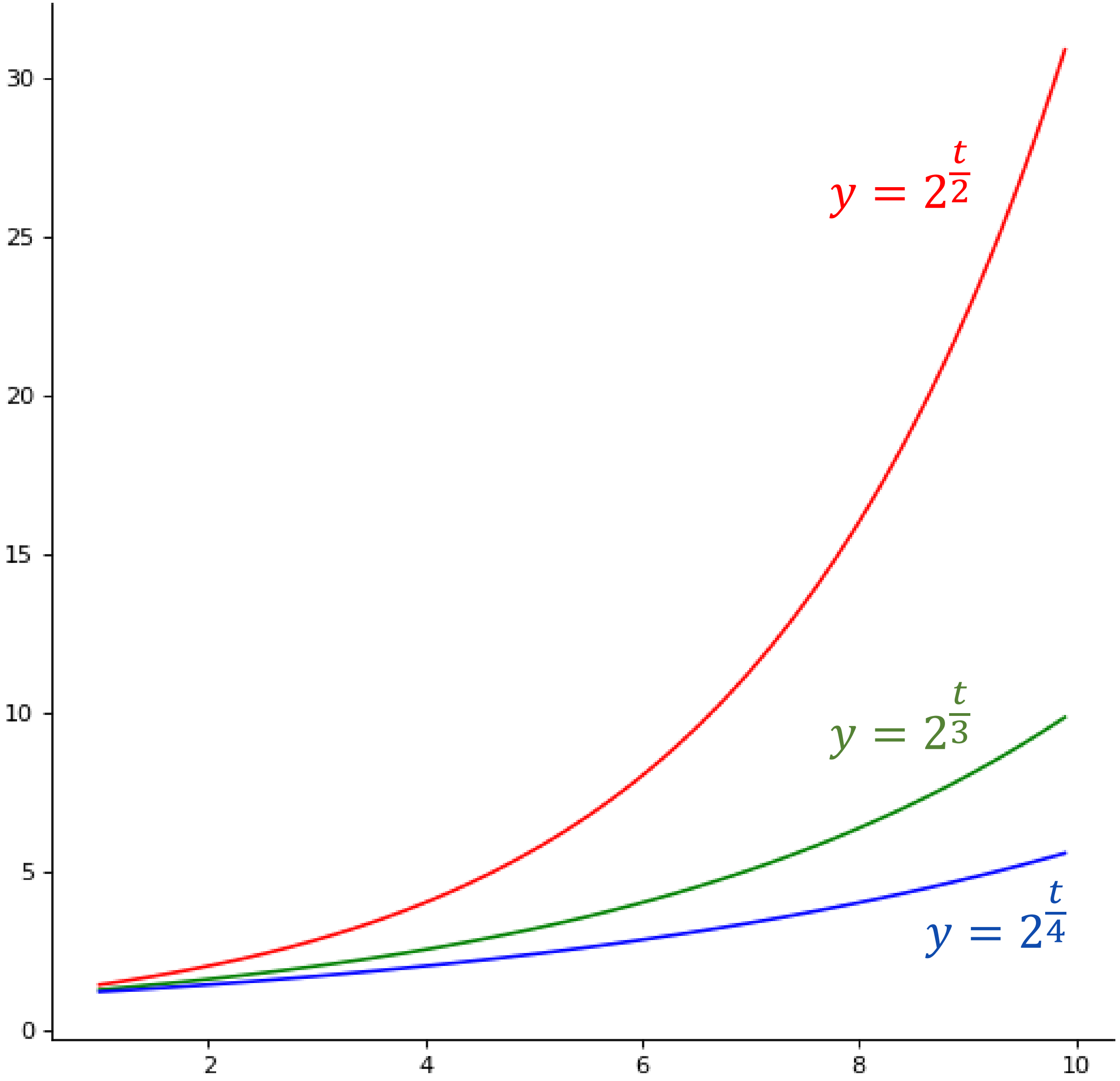
The number of bacteria after 8 hours = ?

The number of bacteria after 1 hours = ?

The number of bacteria after 6 hours = ?

The number of bacteria after 6 hours is

$$N = 1000 \times 2^{\frac{6}{4}} = 2828$$



Exponential Functions

❖ Example

Suppose a person invests *P* dollars in a savings account with an annual interest rate *r*, compounded annually. The amount of money after 1 year is

$$A_1 = A + \mathrm{r}A = A(1 + r)$$

The amount of money after 2 years is

$$A_2 = A_1 + \mathrm{r}A_1 = A(1 + r) + rA(1 + r) = A(1 + r)^2$$

More generally, the amount after t years is

$$A_t = A(1 + r)^t$$

If the money is compounded 2 times per year, the amount of money after half a year is

$$A_{1/2} = A + \frac{r}{2}A = A\left(1 + \frac{r}{2}\right)$$

The amount of money after 1 year is

$$A_1 = A\left(1 + \frac{r}{2}\right) + \frac{r}{2}A\left(1 + \frac{r}{2}\right) = A\left(1 + \frac{r}{2}\right)^2$$

After t years, the amount of money in the account is

$$A_t = A\left(1 + \frac{r}{2}\right)^{2t}$$

This example is from the reference book

Exponential Functions

❖ Example

This example is from the reference book

After t years, the amount of money in the account is

$$A_t = A\left(1 + \frac{r}{2}\right)^{2t}$$

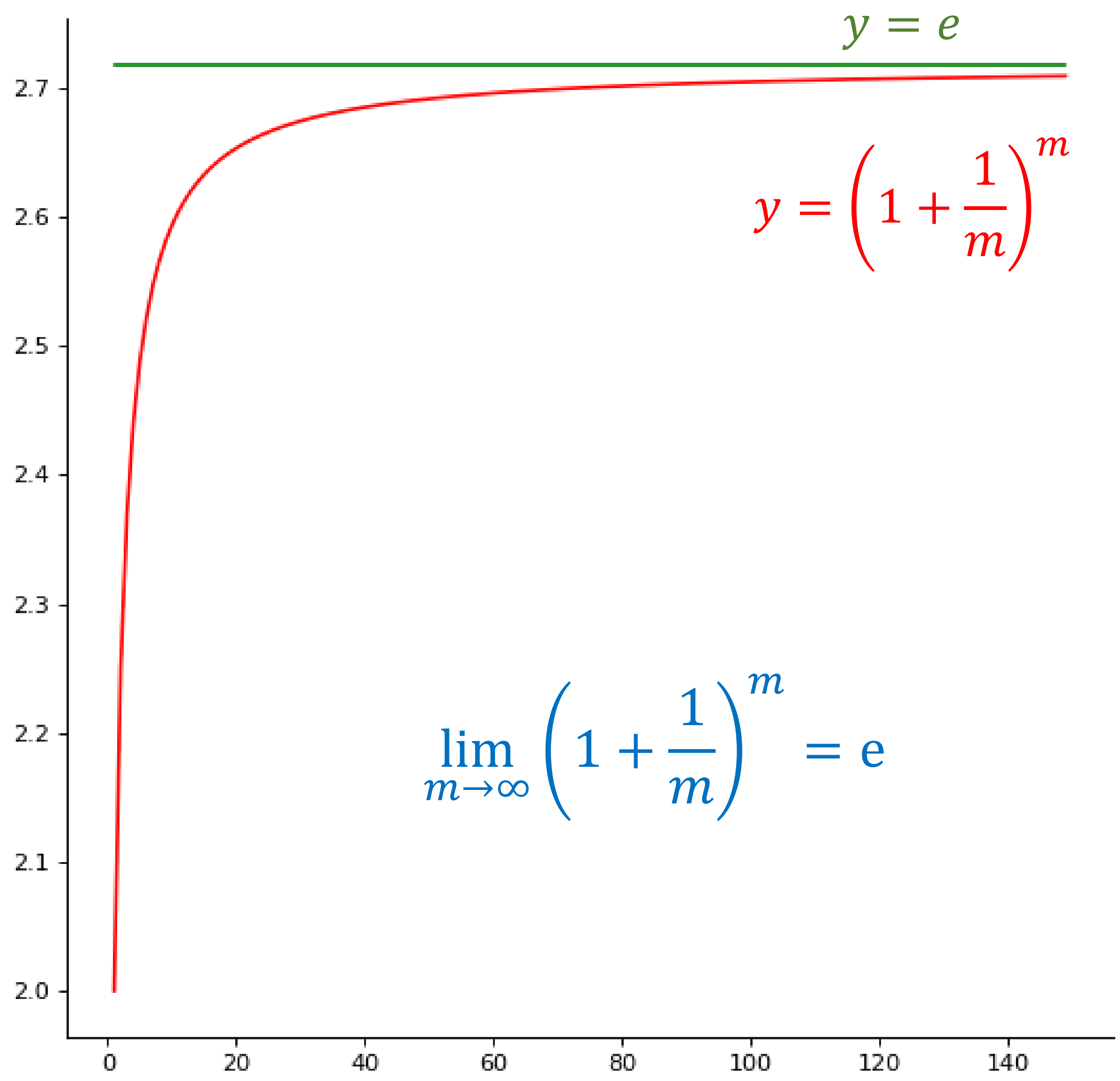
More generally, if the money is compounded n times per year, the amount of money in the account after t years is given by the function

$$A_t = A\left(1 + \frac{r}{n}\right)^{nt}$$

set $m = \frac{n}{r}$

$$A_t = A\left(1 + \frac{1}{m}\right)^{mrt}$$

$$A_t = Ae^{rt}$$



$$\lim_{m \to \infty}\left(1 + \frac{1}{m}\right)^m = e$$

Exponential Functions

❖ Example

This example is from the reference book

After t years, the amount of money in the account is

$$A_t = A\left(1 + \frac{r}{2}\right)^{2t}$$

More generally, if the money is compounded n times per year, the amount of money in the account after t years is given by the function

$$A_t = A\left(1 + \frac{r}{n}\right)^{nt}$$

set $m = \frac{n}{r}$

$$A_t = A\left(1 + \frac{1}{m}\right)^{mrt}$$

$$A_t = Ae^{rt}$$

Suppose \$5000 is invested in an account at an annual interest rate of r = 6%, compounded continuously.

After 30 years

$$A_t = Ae^{rt}$$

$$= 5000e^{0.06*30}$$

$$\approx 30248$$

Exponential Functions

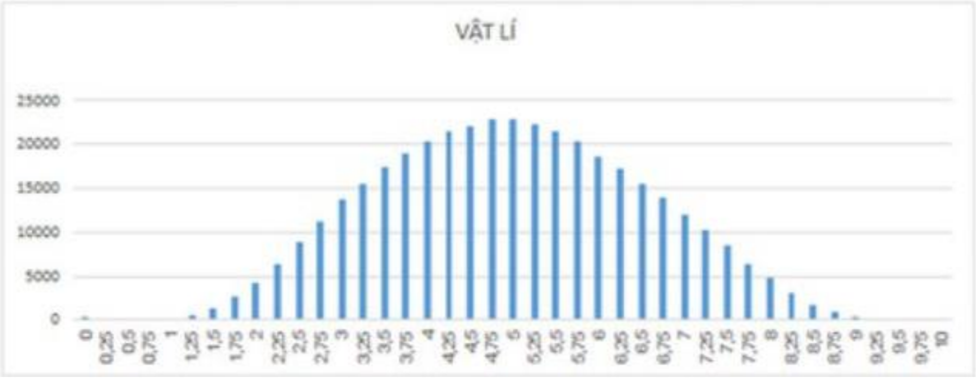
❖ Example

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$$

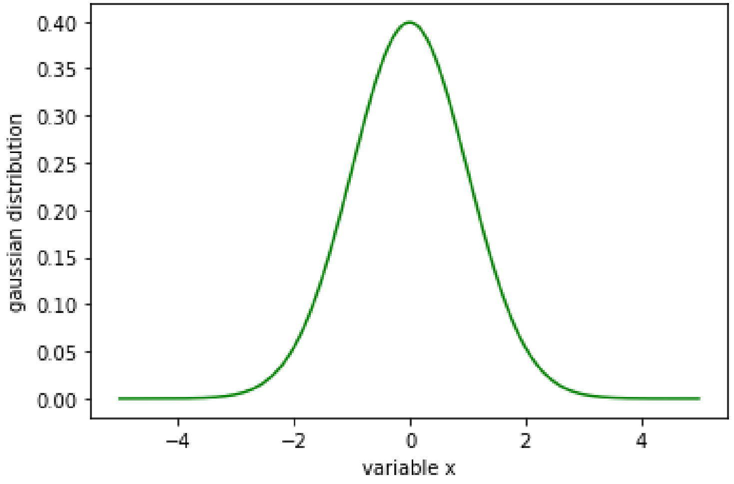
$$-\infty < x < \infty$$

μ: mean

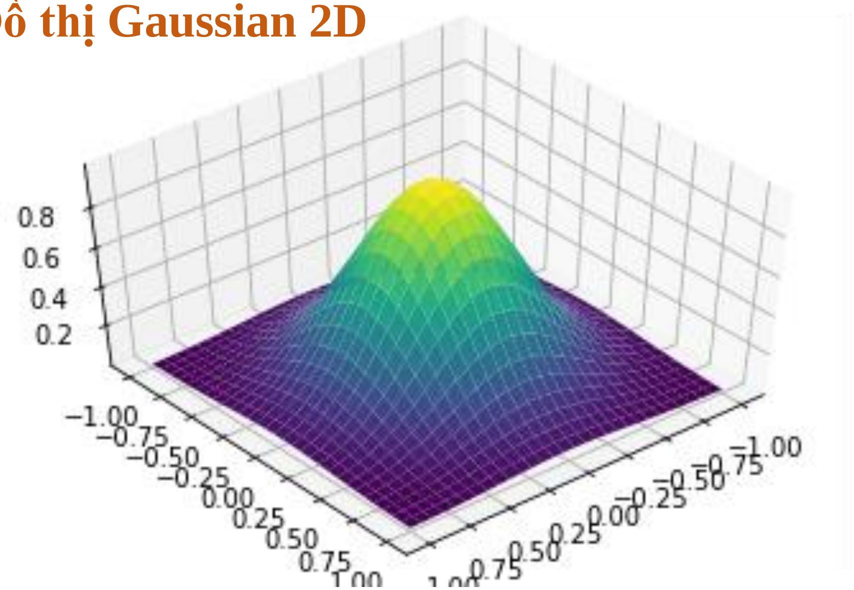
σ^2: variance



Đồ thị Gaussian 1D



Đồ thị Gaussian 2D



Bilateral filter

$$k(x) = \int_W c(\xi, x) s(f(\xi), f(x)) d\xi$$

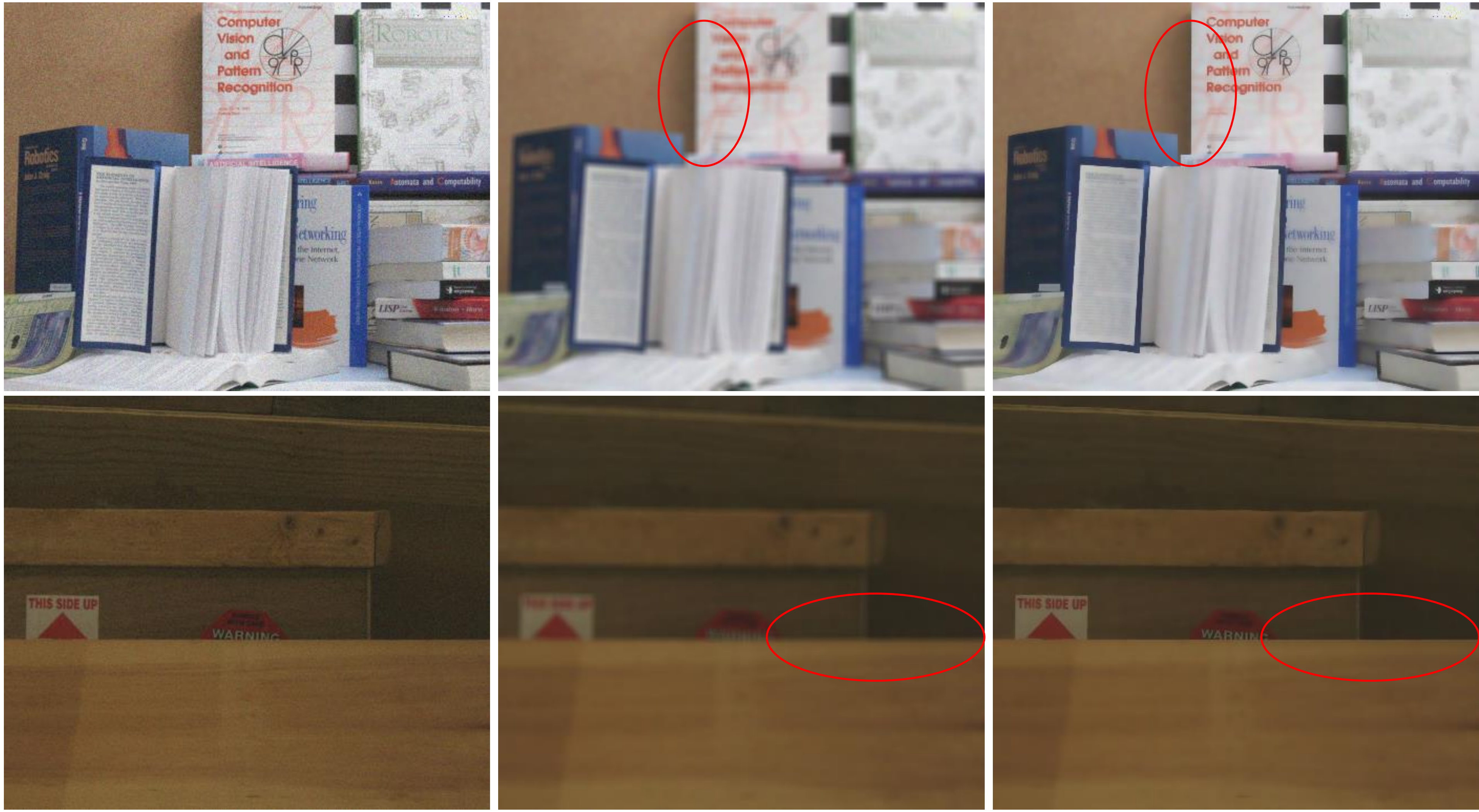
where W is the filter window. The
are the weight functions, essential
tions of the form:

$$c(\xi, x) = e^{-\frac{1}{2}(\frac{|\xi - x|}{\sigma_d})^2},$$

$$s(f(\xi), f(x)) = e^{-\frac{1}{2}(\frac{|f(\xi) - f(x)|}{\sigma_r})^2},$$

where $f(x)$ is an intensity value
σ_d are the scale parameters of th
components, respectively.

from my paper



Ảnh gốc bị nhiễu

Smoothing với trọng số Gausian

Smoothing với trọng số Gausian+color

Sigmoid Function

Sigmoid function

$$y = \sigma(u) = \frac{1}{1 + e^{-u}}$$

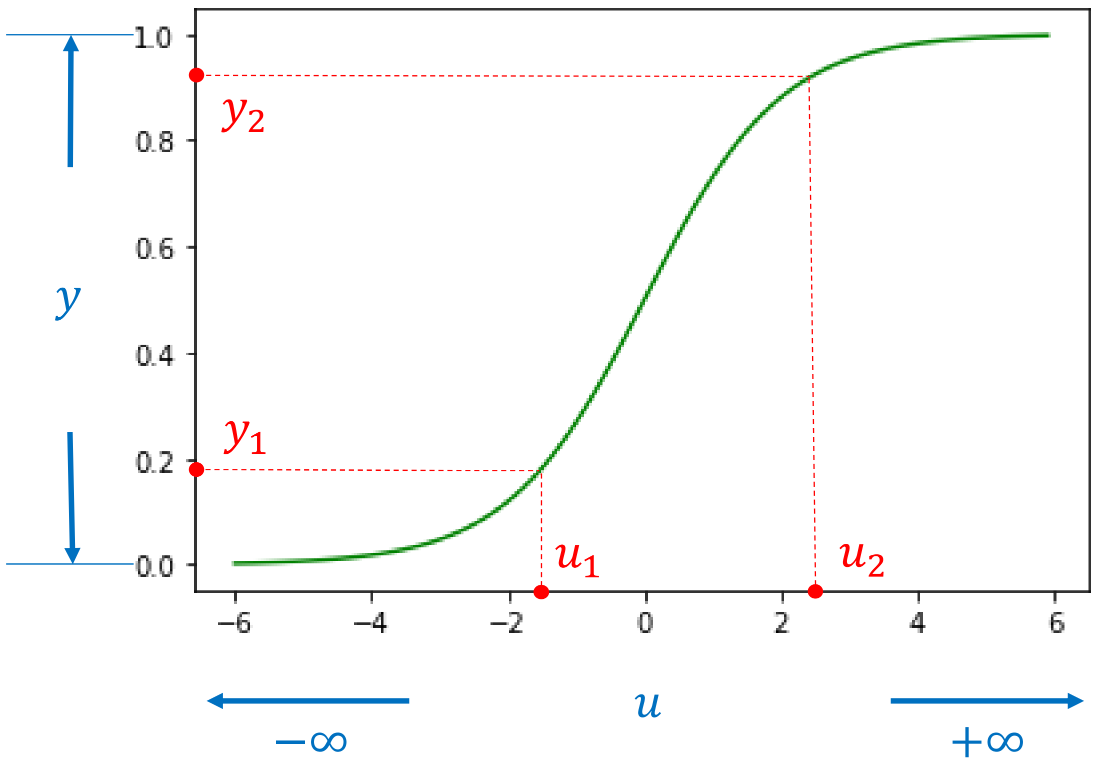
$$u \in (-\infty \quad +\infty)$$

$$y \in (0 \quad 1)$$

Property

$$\forall u_1 u_2 \in [a \ \ b] \text{ và } u_1 \leq u_2$$

$$\rightarrow \sigma(u_1) \leq \sigma(u_1)$$

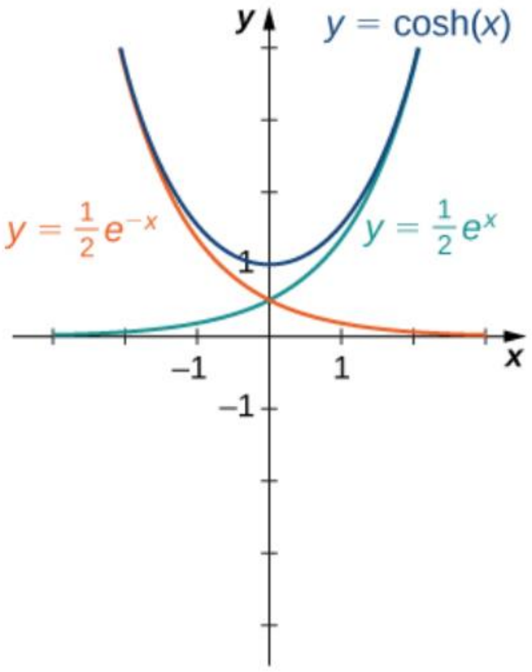


Hyperbolic Functions

❖ Definition

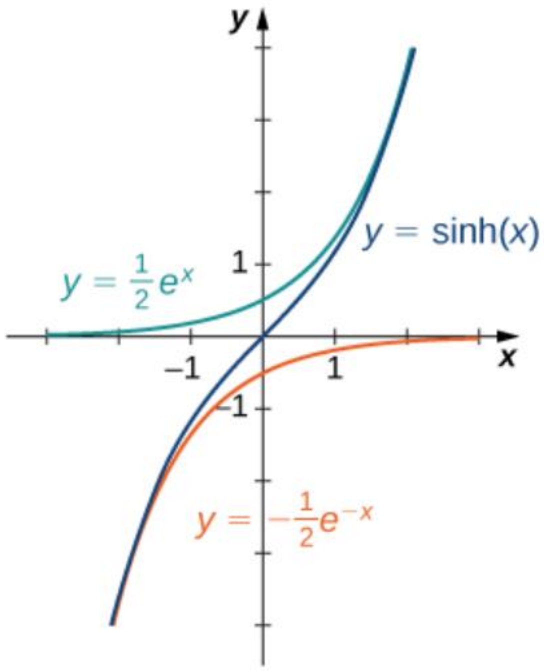
Defined in terms of certain combinations of e^x and e^{-x}

Three figures are from the reference book



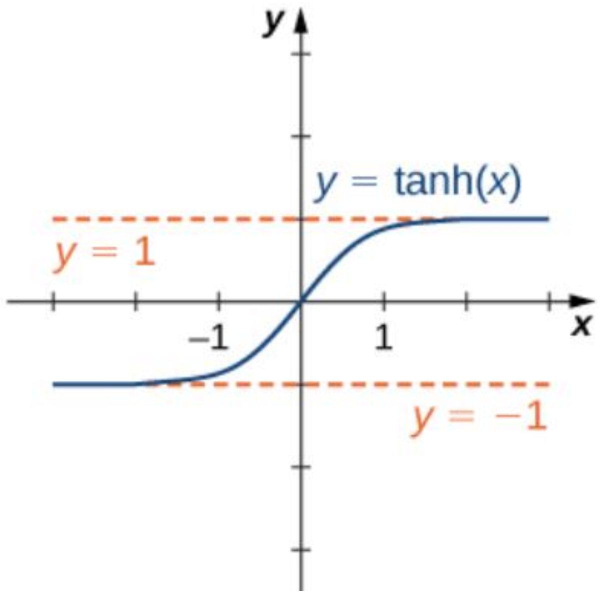
Hyperbolic cosine

$$cosh(x) = \frac{e^x + e^{-x}}{2}$$



Hyperbolic sine

$$sinh(x) = \frac{e^x - e^{-x}}{2}$$



Hyperbolic tangent

$$tanh(x) = \frac{sinh(x)}{cosh(x)} = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

Implementation (straightforward)

Softmax function

Chuyển các giá trị của một vector thành các giá trị xác suất

Formula

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

$$0 \leq f(x_i) \leq 1$$

$$\sum_i f(x_i) = 1$$

Input

$x_1 = 1.0$

$x_2 = 2.0$

$x_3 = 3.0$

→ **Softmax** →

Probability

$f(x_1) = 0.09$

$f(x_2) = 0.24$

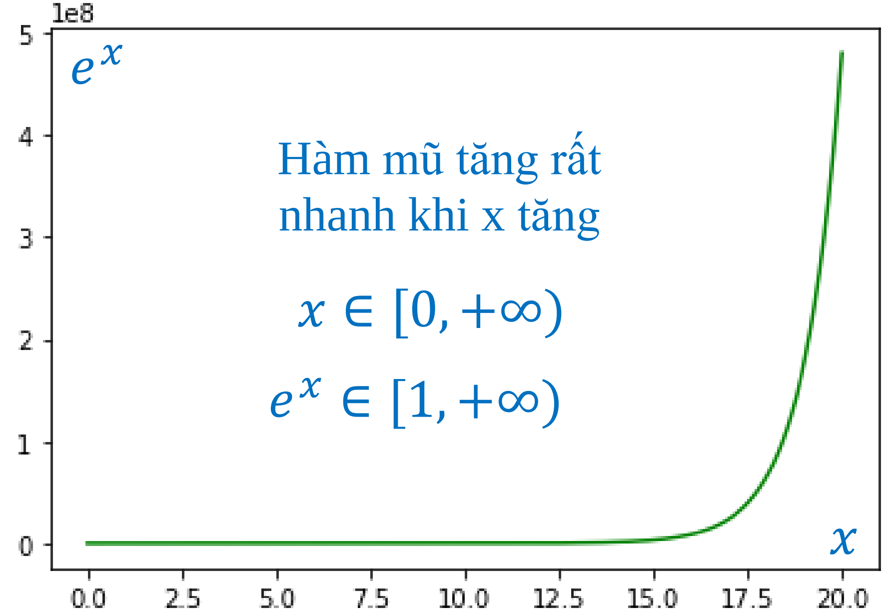
$f(x_3) = 0.67$

Giá trị nan vì e^x vượt giới hạn lưu trữ của biến

```
def naive_softmax(data):
    exp_data = [math.exp(x) for x in data]
    sum_exp_data = sum(exp_data)
    result = [x/sum_exp_data for x in exp_data]
    return result
```

```
data1 = [1.0, 2.0, 3.0]
print(naive_softmax(data1))
```

```
[0.09003057317038046, 0.24472847105479767, 0.6652409557748219]
```



Softmax function

Implementation (straightforward)

Chuyển các giá trị của một vector thành các giá trị xác suất

Formula

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

$$0 \le f(x_i) \le 1$$

$$\sum_i f(x_i) = 1$$

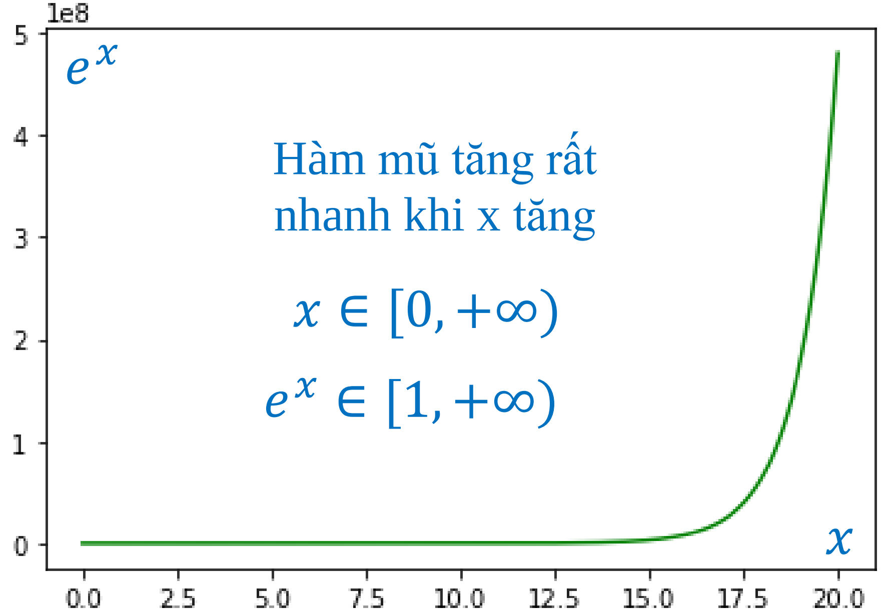
Input		Probability
$x_1 = 1.0$		$f(x_1) = 0.09$
$x_2 = 2.0$	→ **Softmax** →	$f(x_2) = 0.24$
$x_3 = 3.0$		$f(x_3) = 0.67$

Giá trị nan vì e^x vượt giới hạn lưu trữ của biến

```python
def naive_softmax(data):
    exp_data = [math.exp(x) for x in data]
    sum_exp_data = sum(exp_data)
    result = [x/sum_exp_data for x in exp_data]
    return result
```

```python
data2 = [1000.0, 1001.0, 1002.0]
print(naive_softmax(data2))
```

```
---------------------------------------------------------------------------
OverflowError                             Traceback (most recent call last)
Cell In [8], line 2
```



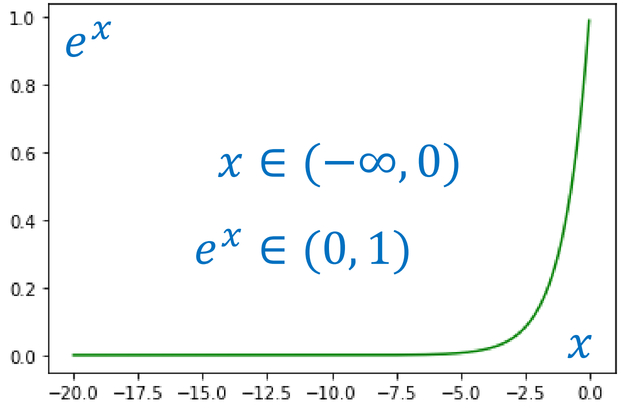
Softmax function (stable)

Formula

$$m = \max(\boldsymbol{x})$$

$$f(x_i) = \frac{e^{(x_i - m)}}{\sum_j e^{(x_j - m)}}$$

X	X-m		Probability
$x_1 = 1.0$	$x_1 = -2.0$		$f(x_1) = 0.09$
$x_2 = 2.0$	$x_2 = -1.0$	Softmax	$f(x_2) = 0.24$
$x_3 = 3.0$	$x_3 = 0$		$f(x_3) = 0.67$




```python
import math

def robust_softmax(data):
    max_value = max(data)
    data = [x-max_value for x in data]

    exp_data = [math.exp(x) for x in data]
    sum_exp_data = sum(exp_data)
    result = [x/sum_exp_data for x in exp_data]

    return result
```

```python
data1 = [1.0, 2.0, 3.0]
print(robust_softmax(data1))
# [0.0900305, 0.2447284, 0.6652409]
```

```python
data2 = [1000.0, 1001.0, 1002.0]
print(robust_softmax(data2))
# [0.0900305, 0.2447284, 0.6652409]
```

```python
data3 = [1.0, 1001.0, 1002.0]
print(robust_softmax(data3))
# [0.0, 0.26894142, 0.73105857]
```

Outline

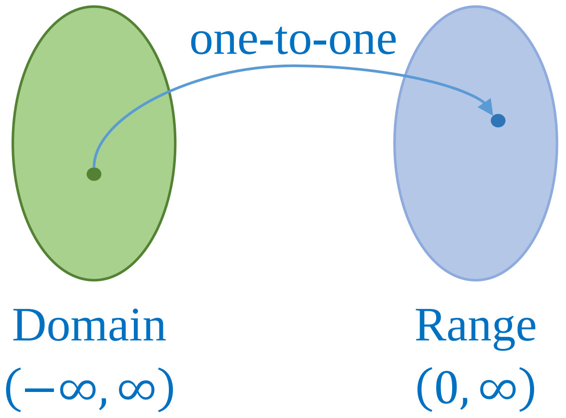
- Exponential Functions
- Logarithms
- Logarithm Applications
- Designing Functions for Evaluation
- Functions for Imbalanced Signals

Logarithmic Functions

❖ Definition

Exponential function

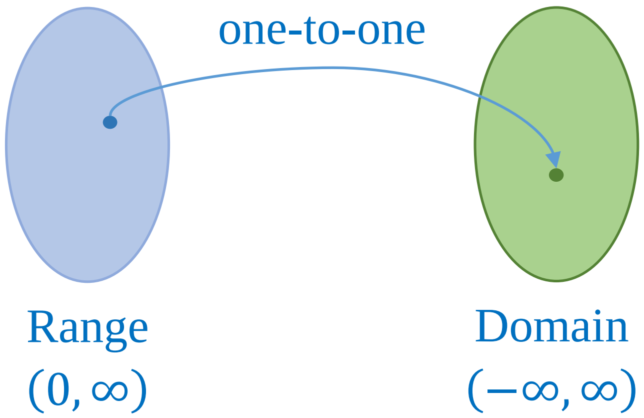
$$f(x) = a^x$$



For any constant a > 0, a ≠ 1

The logarithmic function with base b is

$$log_a(x) = y \quad \Leftrightarrow \quad a^y = x$$



Logarithmic Functions

❖ Properties of Logarithms

If a, $b, c > 0$, a $\neq 1$, and
r is any real number

$$log_a(a^x) = x$$

$$a^{log_a(x)} = x$$

$$log_a(bc) = log_a(b) + log_a(c)$$

$$log_a\left(\frac{b}{c}\right) = log_a(b) - log_a(c)$$

$$log_a(b^r) = r log_a(b)$$

Using definition

prove $log_a(bc) = log_a(b) + log_a(c)$

set $m = log_a(b) \Rightarrow a^m = b$ (1)

set $n = log_a(c) \Rightarrow a^n = c$ (2)

$(1) * (2) \Rightarrow bc = a^n a^m = a^{n+m}$

$\Leftrightarrow log_a(bc) = log_a(a^{n+m})$

$\Leftrightarrow log_a(bc) = n + m$

$\Leftrightarrow log_a(bc) = log_a(b) + log_a(c)$

Logarithmic Functions

❖ Properties of Logarithms

If a, $b, c > 0$, a $\neq 1$, and
r is any real number

$$\log_a(a^x) = x$$

$$a^{\log_a(x)} = x$$

$$\log_a(bc) = \log_a(b) + \log_a(c)$$

$$\log_a\left(\frac{b}{c}\right) = \log_a(b) - \log_a(c)$$

$$\log_a(b^r) = r\log_a(b)$$

prove $\log_a(b^r) = r\log_a(b)$

set $m = \log_a(b) \Rightarrow a^m = b$ (1)

raise (1) to the r power

$$b^r = (a^m)^r = a^{mr} \quad (2)$$

Perform logarithms with base a to (2)

$$\log_a(b^r) = \log_a(a^{mr})$$

$$\Rightarrow \log_a(b^r) = mr$$

$$\Rightarrow \log_a(b^r) = r\log_a(b)$$

Logarithmic Functions

❖ Natural logarithm

$ln(x)$ mean $log_e(x)$

$$ln(e) = log_e(e) = 1$$

$$ln(e^5) = log_e(e^5) = 5$$

$$ln(1) = log_e(1) = 0$$

❖ Change base

Let a, $b > 0$, and a $\neq 1$, b $\neq 1$

$$log_a(x) = \frac{log_b(x)}{log_b(a)}$$

for any real number $x > 0$

Logarithm

Logarithm trả lời câu hỏi:

Nhân bao nhiêu lần cho một số để bằng một số khác

Ví dụ: Nhân bao nhiêu lần cho số 2 để được 8
Trả lời: 2x2x2 = 8 → nhân 3 lần số 2 để được 8

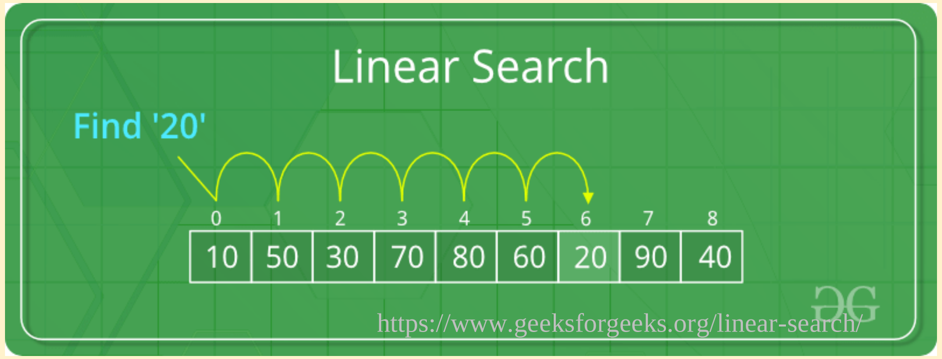
Ví dụ: Nhân bao nhiêu lần số 2 để được 16
Trả lời: 2x2x2x2 = 16 → nhân 4 lần số 2 để được 8

https://www.mathsisfun.com/algebra/logarithms.html

Logarithm trong Big O

Big O: Tính độ phức tạp của thuật toán khi problem size tăng lên

Với linear search như hình bên dưới



trường hợp tệ nhất là khi tìm '40'. Lúc này, tất cả N phần tử đều phải được duyệt trước khi tìm ra '40'
→ độ phức tạp của linear search là O(N): trong trương hợp tệ nhất cần N bước để tìm ra một phần tử.

Binary search (như hình bên trái) chia đôi problem size ở mỗi bước. Do đó, trường hợp tệ nhất (như tìm key=9), chúng ta cần 3 bước với problem size N=8.
→ độ phức tạp của binary search là O(log(N)): trong trương hợp tệ nhất cần log(N) bước để tìm ra một phần tử.

Trong Computer Science, nếu không được nhắc đến, giá trị base mặc định của log là 2.

Logarithm

Công thức phổ biến

$$\log_a a = 1$$

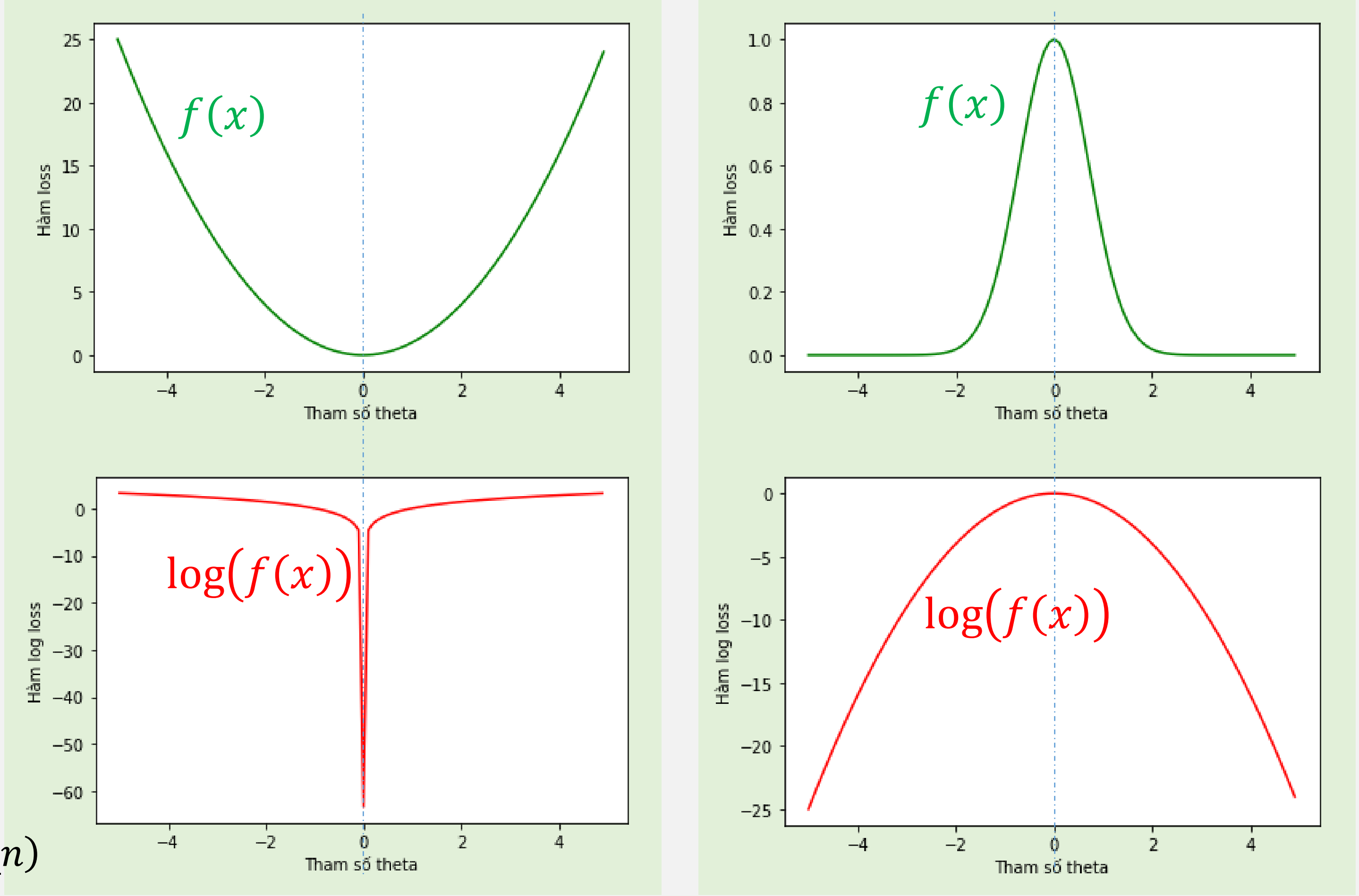
$$\log_a xy = \log_a x + \log_a y$$

Hàm log là hàm đơn điệu (~thứ tự không thay đổi)

$$\forall x_1 x_2 \in [a \;\; b] \text{ và } x_1 \leq x_2$$

$$\rightarrow \log(x_1) \leq \log(x_1)$$

Ứng dụng trong Machine Learning



Ví trí cực đại của hàm $f(\theta)$ và $\log f(\theta)$ không thay đổi

Tìm bộ tham số **θ** cho một model sao cho model mô tả được dữ liệu training

$$\underset{\theta}{\operatorname{argmax}} f(\theta) = \underset{\theta}{\operatorname{argmax}} P_\theta(\text{training data})$$

Với data sample được thu nhập độc lập với nhau

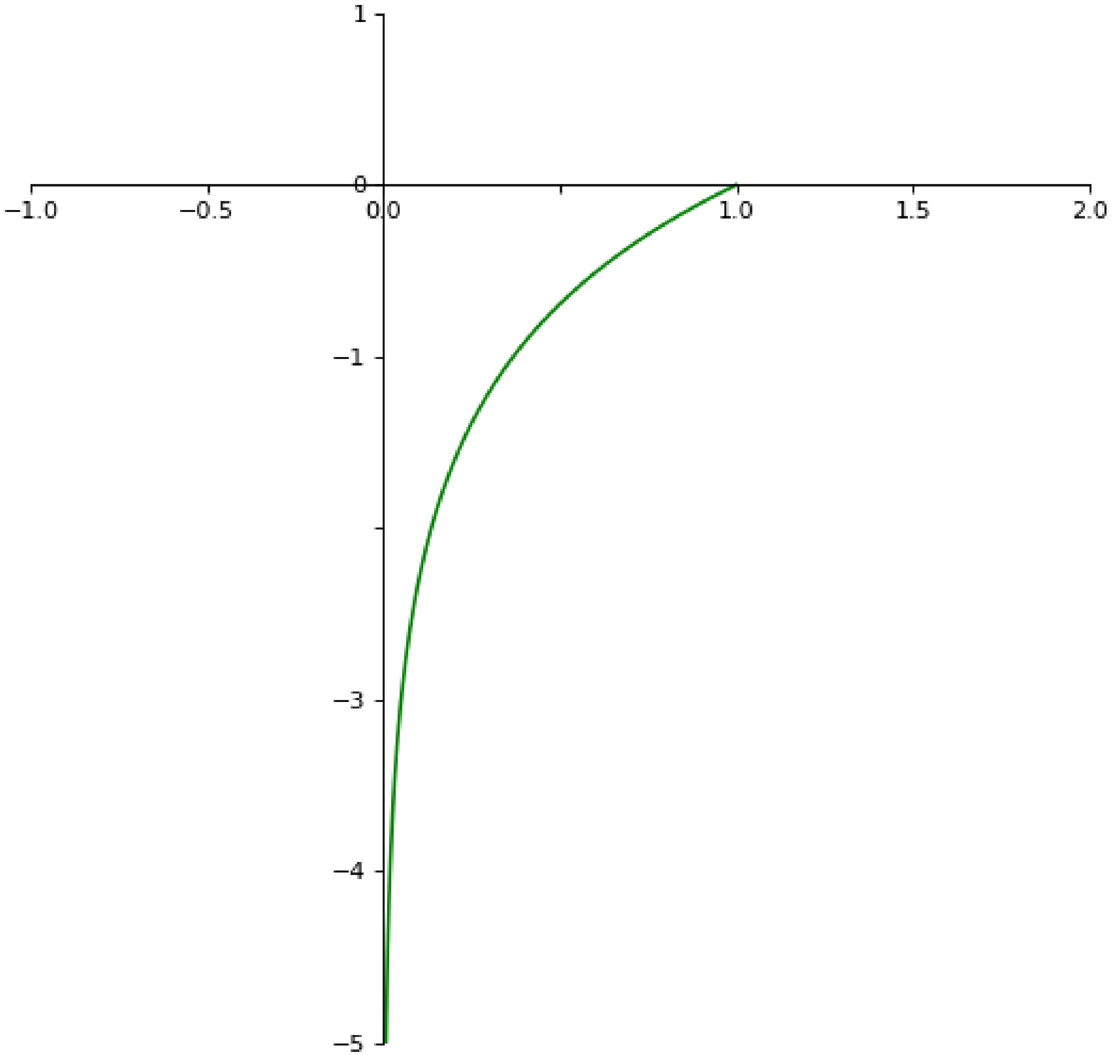
$$\underset{\theta}{\operatorname{argmax}} f(\theta) = \underset{\theta}{\operatorname{argmax}} P_\theta(\text{sample\_1}) * \cdots * P_\theta(\text{sample\_}n)$$

Dùng hàm log

$$\underset{\theta}{\operatorname{argmax}} \log f(\theta) = \underset{\theta}{\operatorname{argmax}} [\log P_\theta(\text{sample\_1}) + \cdots + \log P_\theta(\text{sample\_}n)]$$

Logarithm and Small Numbers

❖ Relative order of numbers is preserved




```python
v1 = 0.0004
v2 = 0.0003
v = v1*v2

m1 = 0.001
m2 = 0.0007
m = m1*m2

print(v)
print(m)
```

```
1.2e-07
7e-07
```

```python
import math

v1 = 0.0004
v2 = 0.0003
v = math.log(v1) + math.log(v2)

m1 = 0.001
m2 = 0.0007
m = math.log(m1) + math.log(m2)

print(v)
print(m)
```

```
-15.935774094164366
-14.172185501903005
```

Outline

- Exponential Functions
- Logarithms
- Logarithm Applications
- Designing Functions for Evaluation
- Functions for Imbalanced Signals

Designing a Function

❖ Lie/Truth classification

Feature | Label

Input	Label
…	Lie
…	Lie
…	Lie
…	Lie
…	Truth
…	Truth
…	Truth
…	Truth

Category 0

Category 1

Assign numbers to categories

Feature | Label

Input	Label
…	0
…	0
…	0
…	0
…	1
…	1
…	1
…	1

Category 0

Category 1

a sample
$x = \cdots, y = 0$

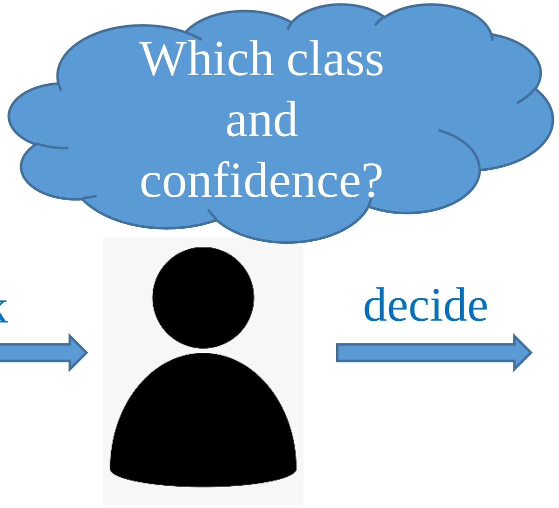
input 1

input 2

…

input n

ask



decide

output 1: $\hat{y} = 0$ (80%)

output 2: $\hat{y} = 1$ (65%)

…

output n: $\hat{y} = 0$ (70%)

evaluate

label 1: $y = 0$

label 2: $y = 0$

…

label n: $y = 1$

Designing a Function

❖ Lie/Truth classification

Feature	Label
Input	Label
…	Lie
…	Lie
…	Lie
…	Lie
…	Truth
…	Truth
…	Truth
…	Truth

Category 0

Category 1

Assign numbers to categories

Feature	Label
Input	Label
…	0
…	0
…	0
…	0
…	1
…	1
…	1
…	1

Category 0

Category 1

a sample
$x = \cdots, y = 0$

output 1: $\hat{y} = 0.2$

output 2: $\hat{y} = 0.65$

input 1

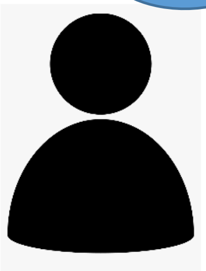
input 2

…

input n

ask

Which class and confidence?



decide

output 1: $\hat{y} = 0$ (80%)

output 2: $\hat{y} = 1$ (65%)

…

output n: $\hat{y} = 0$ (70%)

evaluate

label 1: $y = 0$

label 2: $y = 0$

…

label n: $y = 1$

AI VIETNAM
All-in-One Course

Designing a Function

❖ Lie/Truth classification

Input	Label
…	Lie
…	Lie
…	Lie
…	Lie
…	Truth
…	Truth
…	Truth
…	Truth

Feature | Label

Category 0

Category 1

Assign numbers to categories

Input	Label
…	0
…	0
…	0
…	0
…	1
…	1
…	1
…	1

Feature | Label

Category 0

Category 1

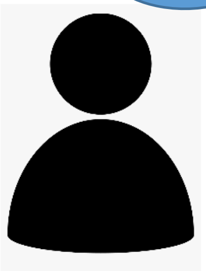
a sample
$x = \cdots, y = 0$

How to measure his/her performance (accuracy)?

input 1
input 2
…
input n

ask

Which class and confidence?



decide

output 1: $\hat{y} = 0.2$
output 2: $\hat{y} = 0.65$
…
output n: $\hat{y} = 0.3$

evaluate

label 1: $y = 0$
label 2: $y = 0$
…
label n: $y = 1$

Designing a Function

❖ Lie/Truth classification

Feature | Output | Label

Input	Output	Label
…	0.3	0
…	0.8	0
…	0.7	0
…	0.4	0
…	0.6	1
…	0.8	1
…	0.9	1
…	0.2	1

Category 0

Category 1

with y = 0

with y = 1

$f(\hat{y}, y) = 1$ either (y = 0 and $\hat{y} \leq 0.5$)
or (y = 1 and $\hat{y} > 0.5$)

$f(\hat{y}, y) = 0$ otherwise

➡ $\text{accuracy} = \frac{5}{8}$

weakness?

Designing a Function

❖ Lie/Truth classification

Feature Output Label

Input	Output	Label
…	0.3	0
…	0.8	0
…	0.7	0
…	0.4	0
…	0.6	1
…	0.8	1
…	0.9	1
…	0.2	1

Category 0

Category 1

Feature Output Label

Input	Output	Label
…	0.1	0
…	0.9	0
…	0.9	0
…	0.2	0
…	0.9	1
…	0.8	1
…	0.9	1
…	0.1	1

Category 0

Category 1

$f(\hat{y}, y) = 1$ either (y $=$ 0 and $\hat{y} \leq 0.5$)
or (y $=$ 1 and $\hat{y} > 0.5$)

$f(\hat{y}, y) = 0$ otherwise

➡ $\text{accuracy} = \frac{5}{8}$

weakness?

Designing a Function

❖ Lie/Truth classification

Feature | Output | Label

Input	Output	Label	
…	0.3	0	Category 0
…	0.8	0	
…	0.7	0	
…	0.4	0	
…	0.6	1	Category 1
…	0.8	1	
…	0.9	1	
…	0.2	1	

Feature | Output | Label

Input	Output	Label	
…	0.1	0	Category 0
…	0.9	0	
…	0.9	0	
…	0.2	0	
…	0.9	1	Category 1
…	0.8	1	
…	0.9	1	
…	0.1	1	

if y = 0

$$f(\hat{y}, y) = 1 \quad \text{if } \hat{y} \leq 0.5$$

if y = 1

$$f(\hat{y}, y) = 1 \quad \text{if } \hat{y} > 0.5$$

otherwise

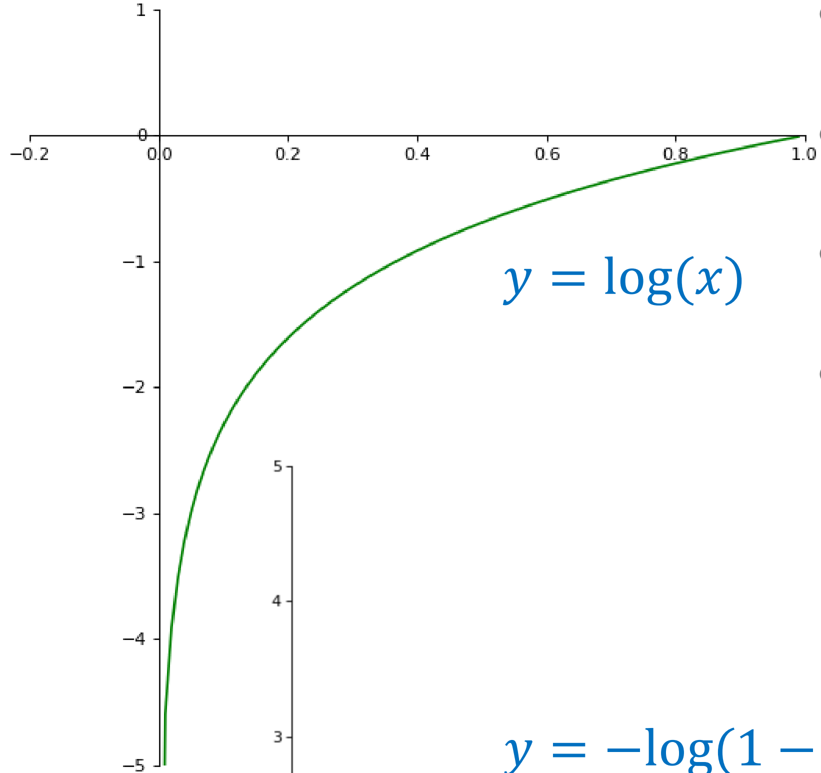
$$f(\hat{y}, y) = 0$$

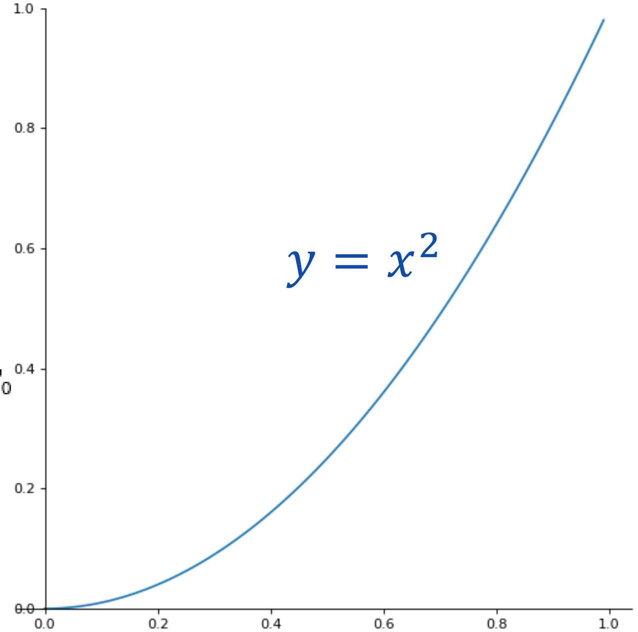
$$\Rightarrow \quad \text{accuracy} = \frac{5}{8}$$

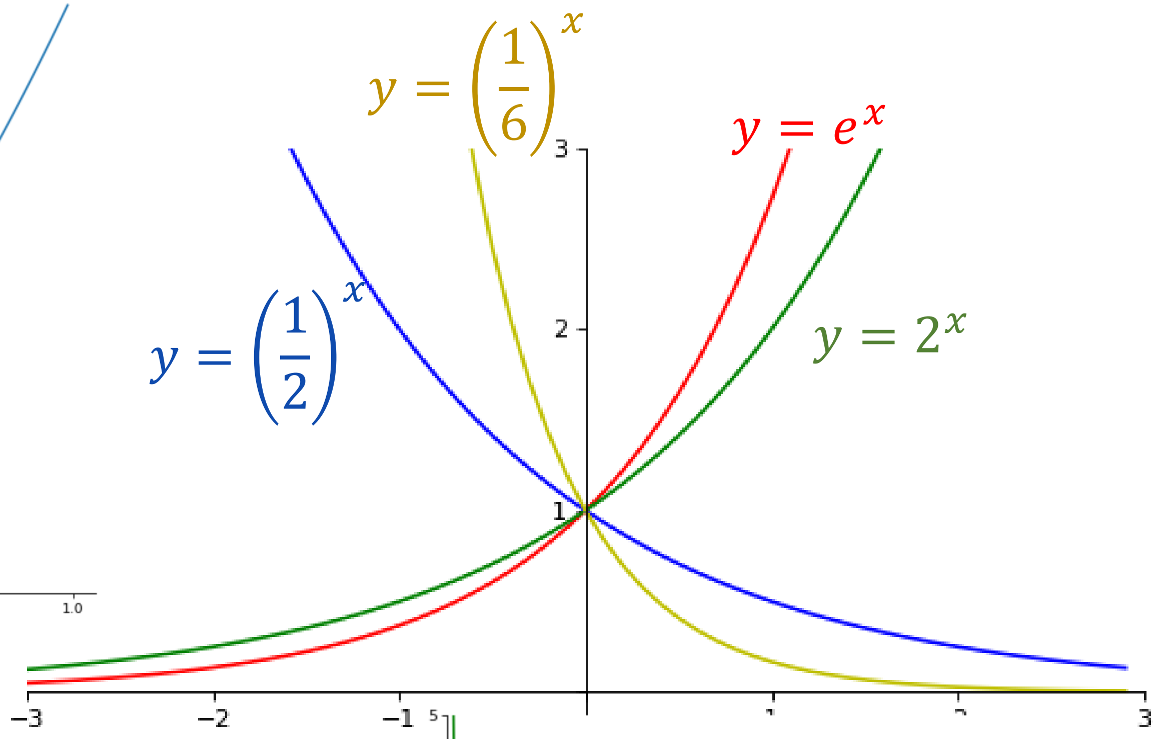
weakness?

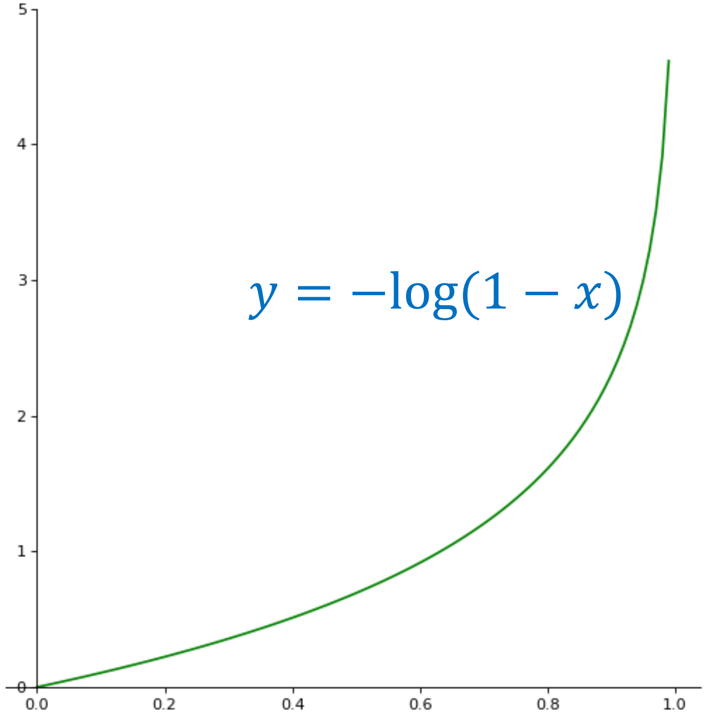
Wrong classification is a serious problem!!!

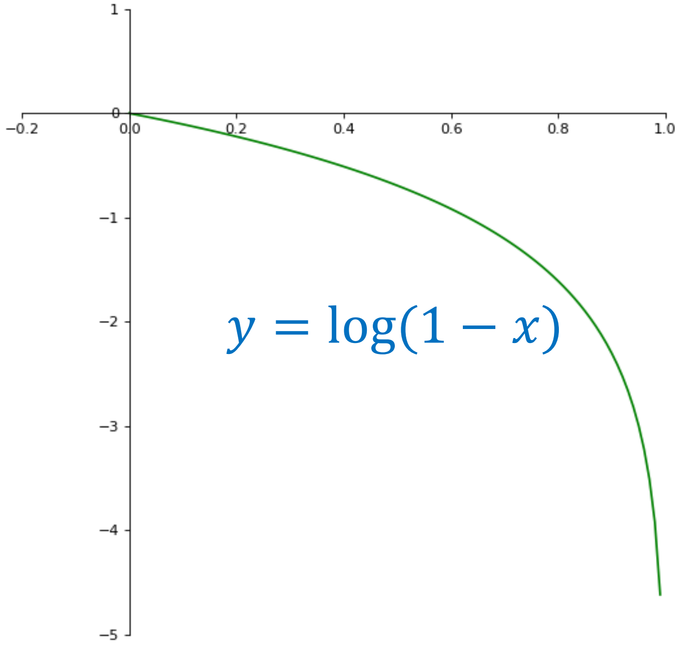
❖ Suggested Functions

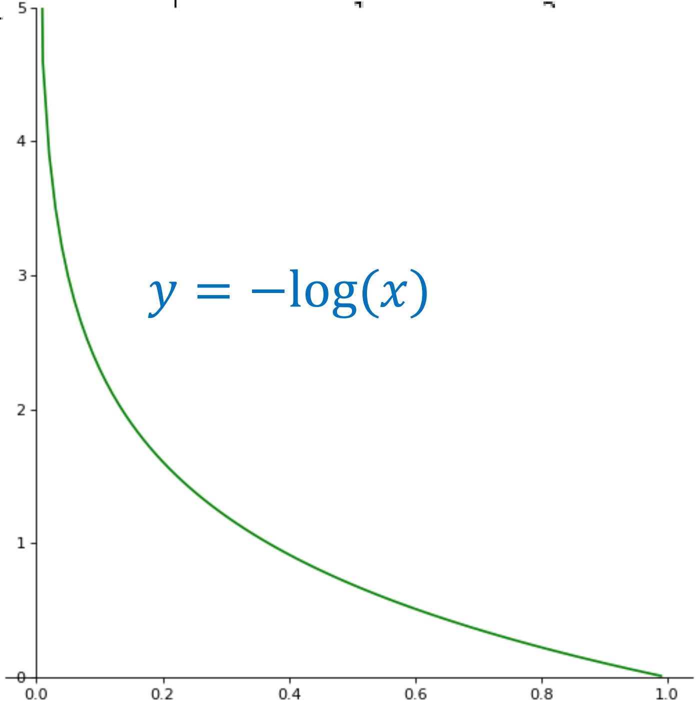












Designing a Function

❖ Lie/Truth classification

Feature | Output | Label

Input	Output	Label
…	0.3	0
…	0.8	0
…	0.7	0
…	0.4	0
…	0.6	1
…	0.8	1
…	0.9	1
…	0.2	1

Category 0

Category 1

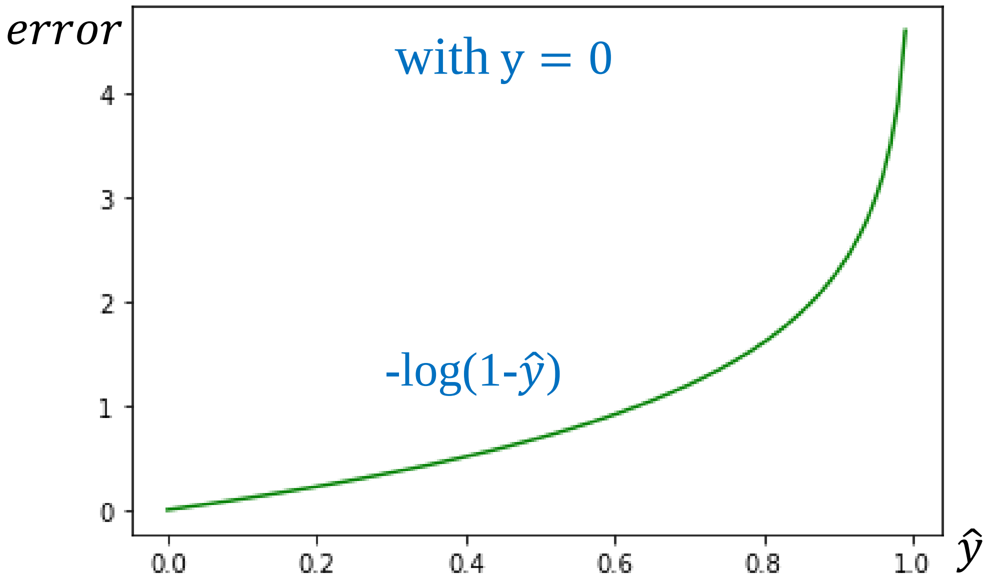
if y = 1

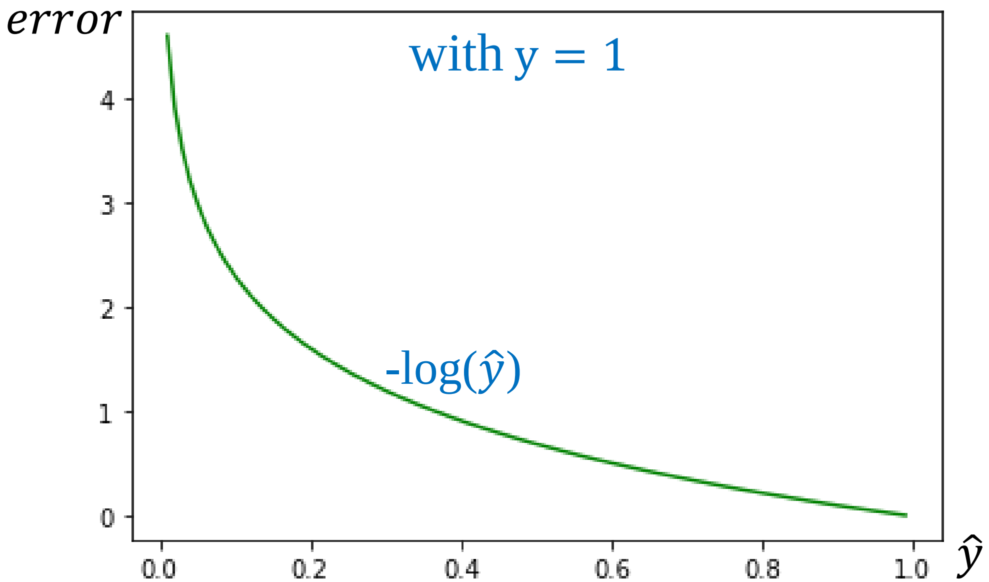
$$L(\hat{y}) = -\log(\hat{y})$$

if y = 0

$$L(\hat{y}) = -\log(1-\hat{y})$$

How to remove if?





Designing a Function

❖ Lie/Truth classification

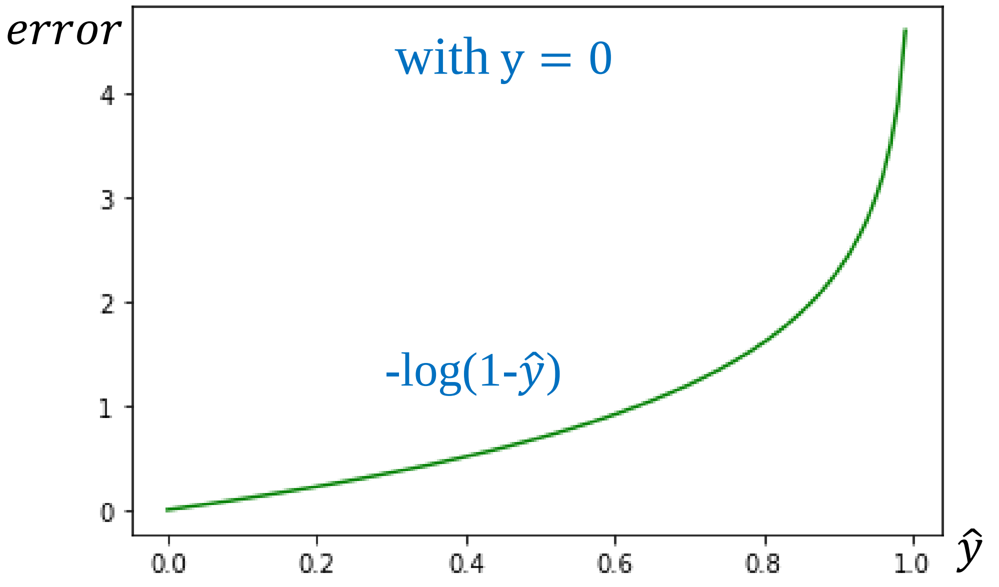
Feature	Output	Label
Input	**Output**	**Label**
…	0.3	0
…	0.8	0
…	0.7	0
…	0.4	0
…	0.6	1
…	0.8	1
…	0.9	1
…	0.2	1

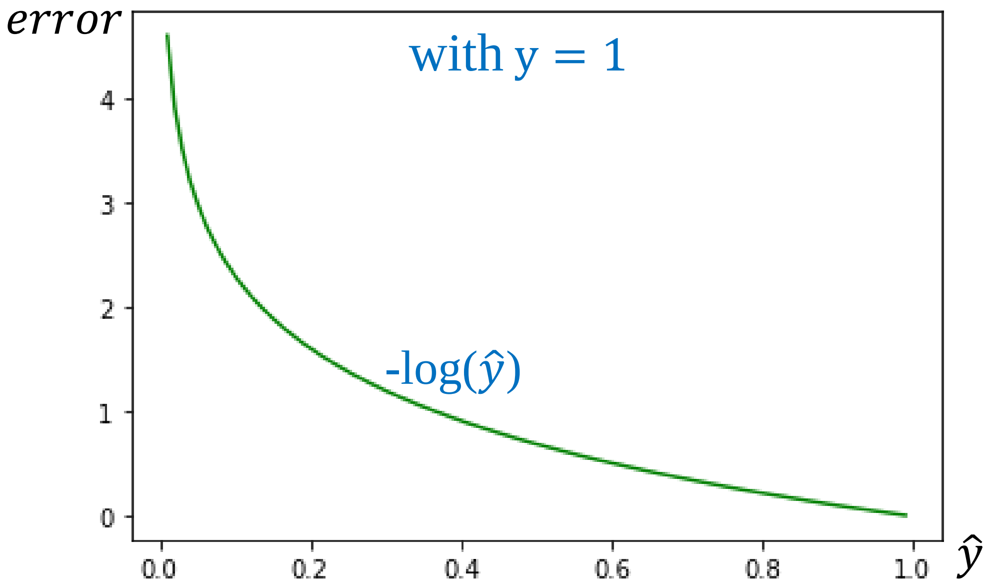
if $y = 0$

$$L(\hat{y}) = -\log(1 - \hat{y})$$

if $y = 1$

$$L(\hat{y}) = -\log(\hat{y})$$





Binary cross-entropy

$$L(y, \hat{y}) = -y\log\hat{y} - (1 - y)\log(1 - \hat{y})$$

Designing a Function

❖ Lie/Truth classification

Feature Output Label

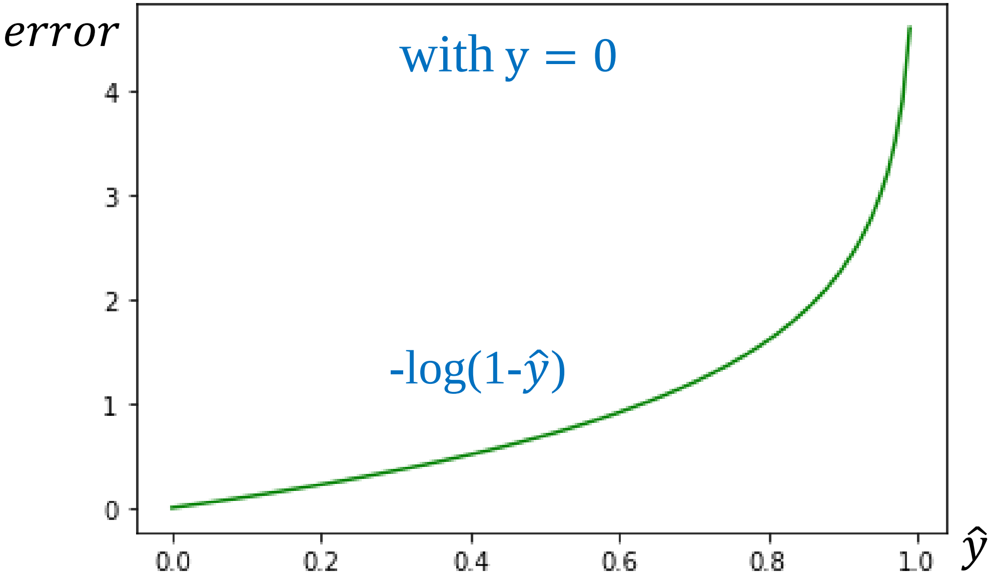
Input	Output	Label	Loss
…	0.3	0	0.356
…	0.8	0	1.609
…	0.7	0	1.203
…	0.4	0	0.511
…	0.6	1	0.511
…	0.8	1	0.223
…	0.9	1	0.105
…	0.2	1	1.609

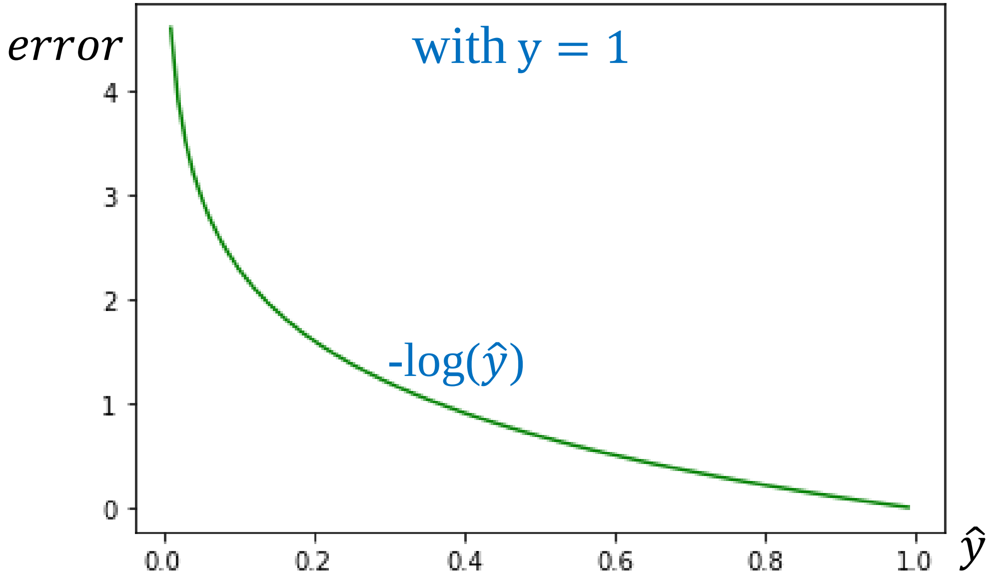
if y = 0

$$L(\hat{y}) = -\log(1-\hat{y})$$

if y = 1

$$L(\hat{y}) = -\log(\hat{y})$$





Binary cross-entropy

$$L(y,\hat{y}) = -y\log\hat{y} - (1-y)\log(1-\hat{y})$$

Designing a Function

❖ Lie/Truth classification

Input	Output	Label
…	0.2	0
…	0.8	0
…	0.3	0
…	0.1	0
…	0.8	1
…	0.8	1
…	0.9	1
…	0.9	1

Feature — Output — Label

After a period of time

if y = 0

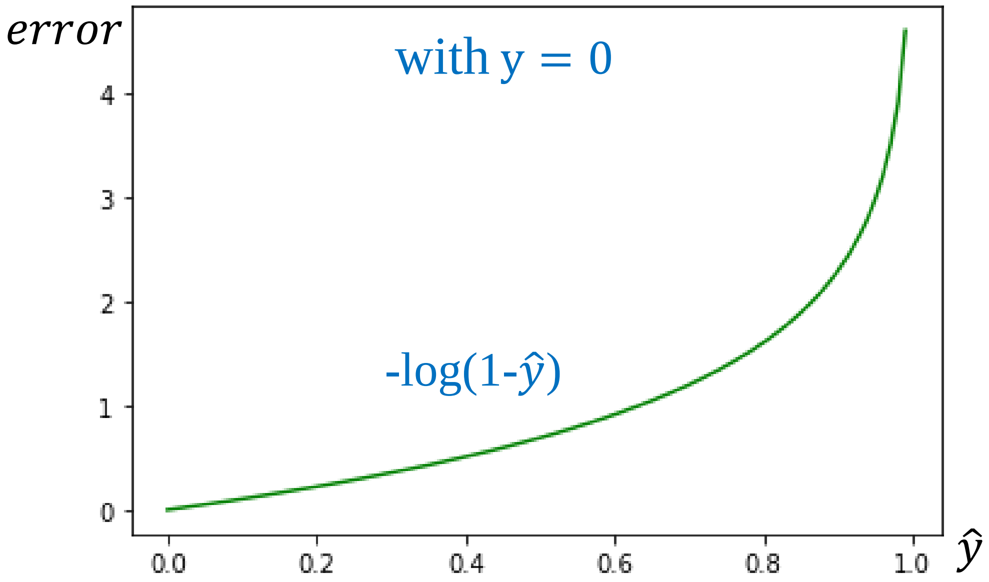
$$L(\hat{y}) = -\log(1-\hat{y})$$

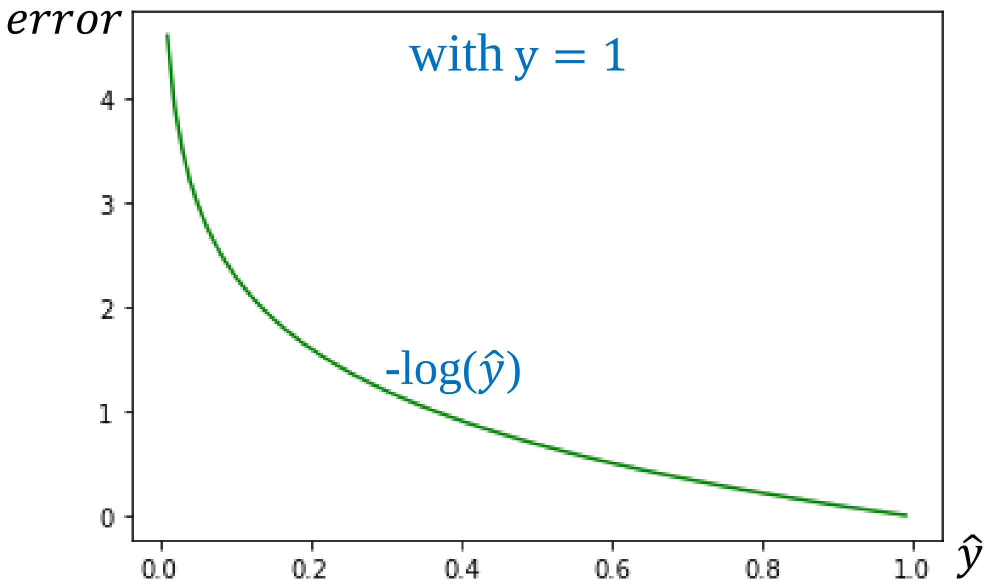
if y = 1

$$L(\hat{y}) = -\log(\hat{y})$$

Binary cross-entropy

$$\mathrm{L}(y,\hat{y}) = -y\log\hat{y} - (1-y)\log(1-\hat{y})$$





Designing a Function

❖ Lie/Truth classification

Feature | Output | Label

Input	Output	Label
…	0.2	0
…	0.7	0
…	0.7	1
…	0.8	1
…	0.8	1
…	0.8	1
…	0.9	1
…	0.9	1
…	0.8	1
…	0.7	1

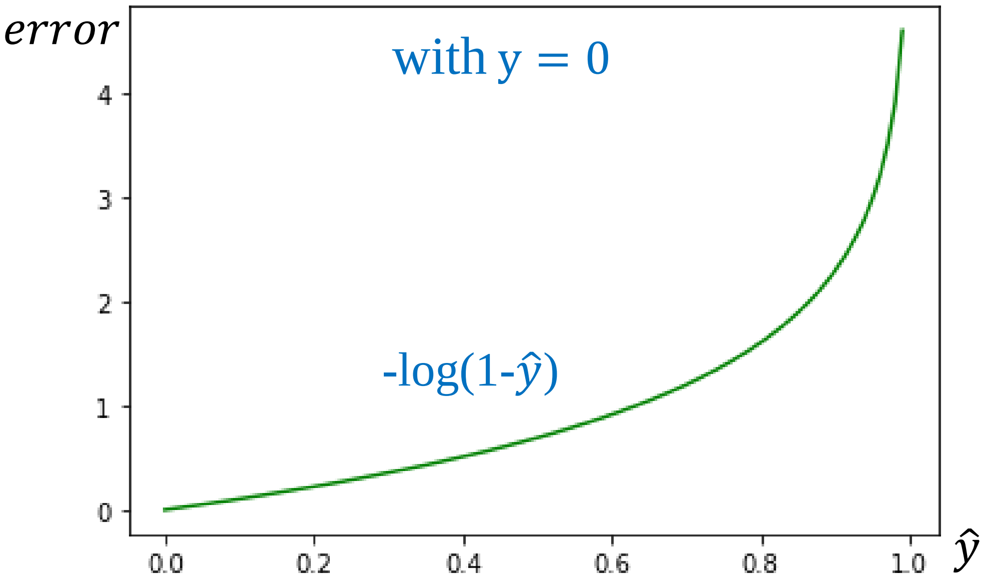
A special context

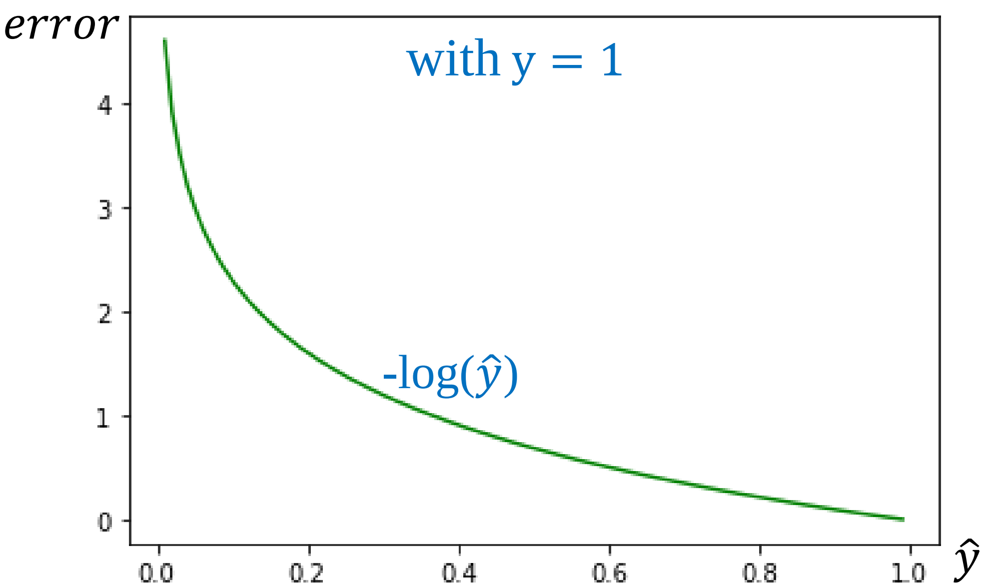
if y = 0

$$L(\hat{y}) = -\log(1-\hat{y})$$

if y = 1

$$L(\hat{y}) = -\log(\hat{y})$$





Binary cross-entropy

$$L(y,\hat{y}) = -y\log\hat{y} - (1-y)\log(1-\hat{y})$$

Designing a Function

❖ Lie/Truth classification

Feature Output Label

Input	Output	Label	Loss
…	0.7	0	1.204
…	0.8	1	0.223
…	0.7	1	0.356
…	0.8	1	0.223
…	0.8	1	0.223
…	0.8	1	0.223
…	0.9	1	0.105
…	0.9	1	0.105
…	0.8	1	0.223
…	0.7	1	0.356

a more severe context!

$$loss_{y=0} = 1.204$$

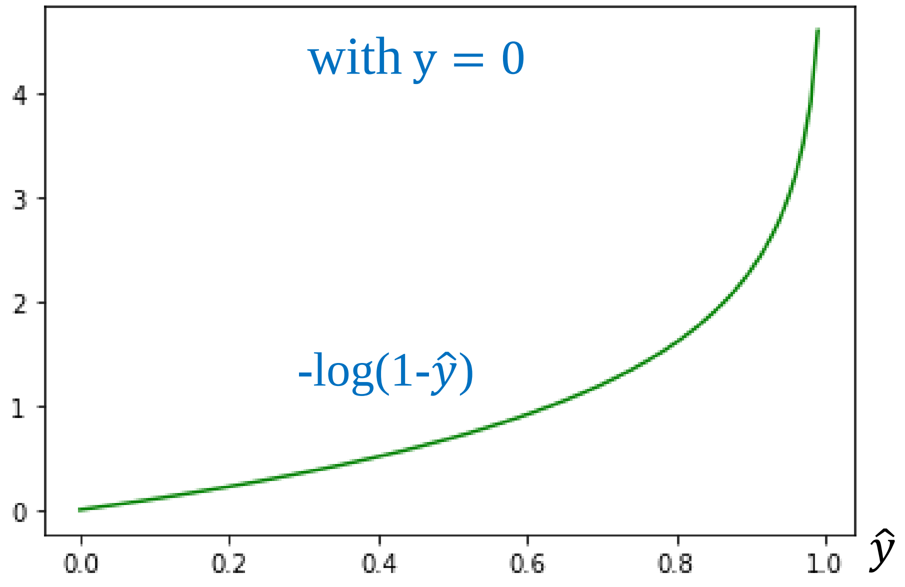
$$loss_{y=1} = 2.039$$

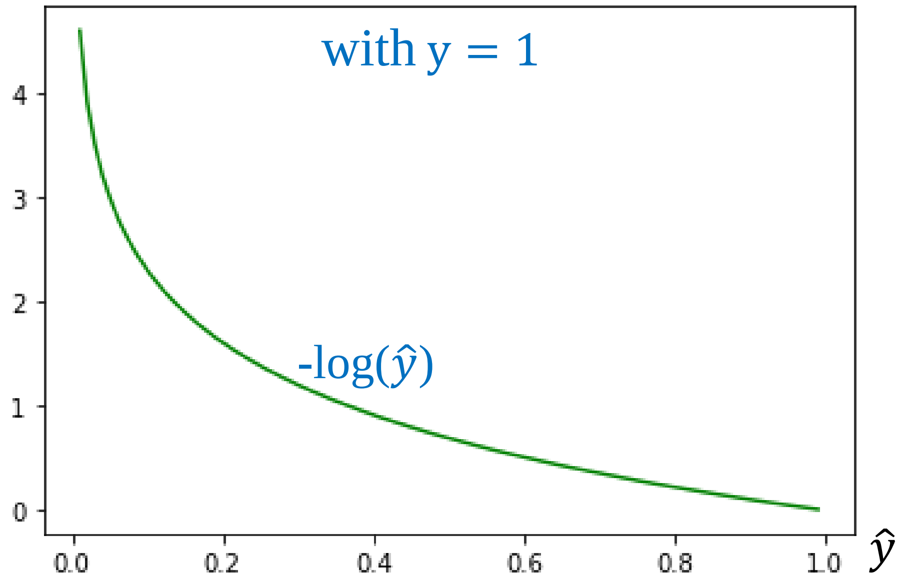
if y = 0

$$L(\hat{y}) = -\log(1-\hat{y})$$

if y = 1

$$L(\hat{y}) = -\log(\hat{y})$$



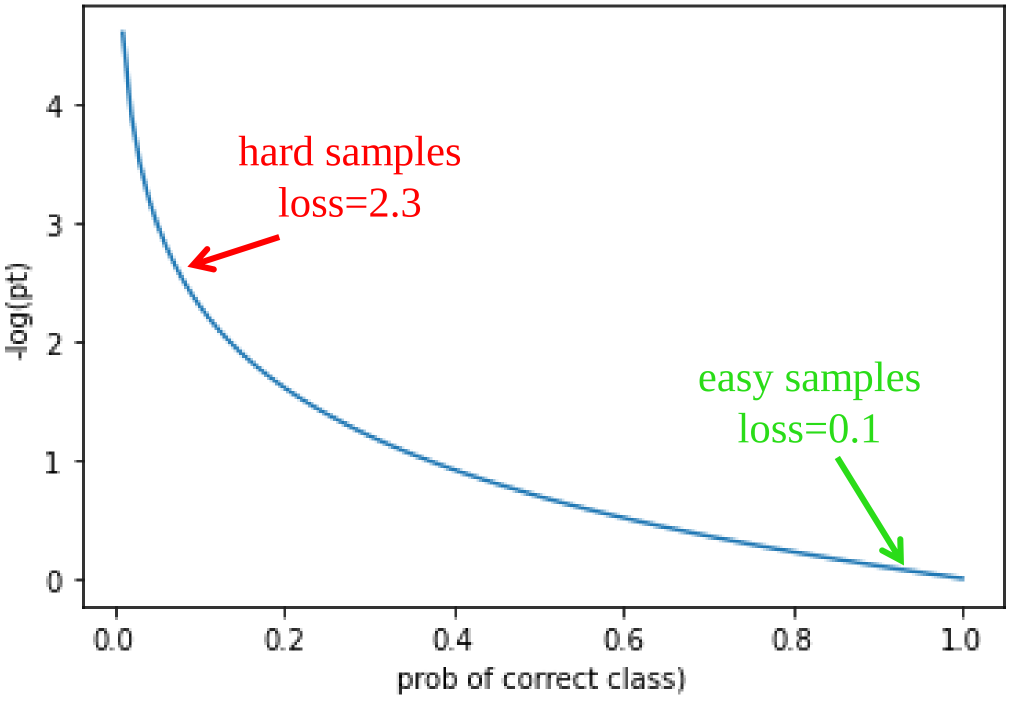


Binary cross-entropy

$$L(y, \hat{y}) = -y\log\hat{y} - (1-y)\log(1-\hat{y})$$

Designing a Function

❖ Imbalance data



Imbalance Case:
- 100000 easy samples vs 100 hard samples

Easy samples loss = 100000*0.1 = 10000
Hard samples loss = 100*2.3 = 230
Loss = Easy samples loss + Hard samples loss
Easy samples loss: Hard samples loss = 10000:230 ≈ 43

BCE không tốt cho trường hợp data bị imbalance nặng

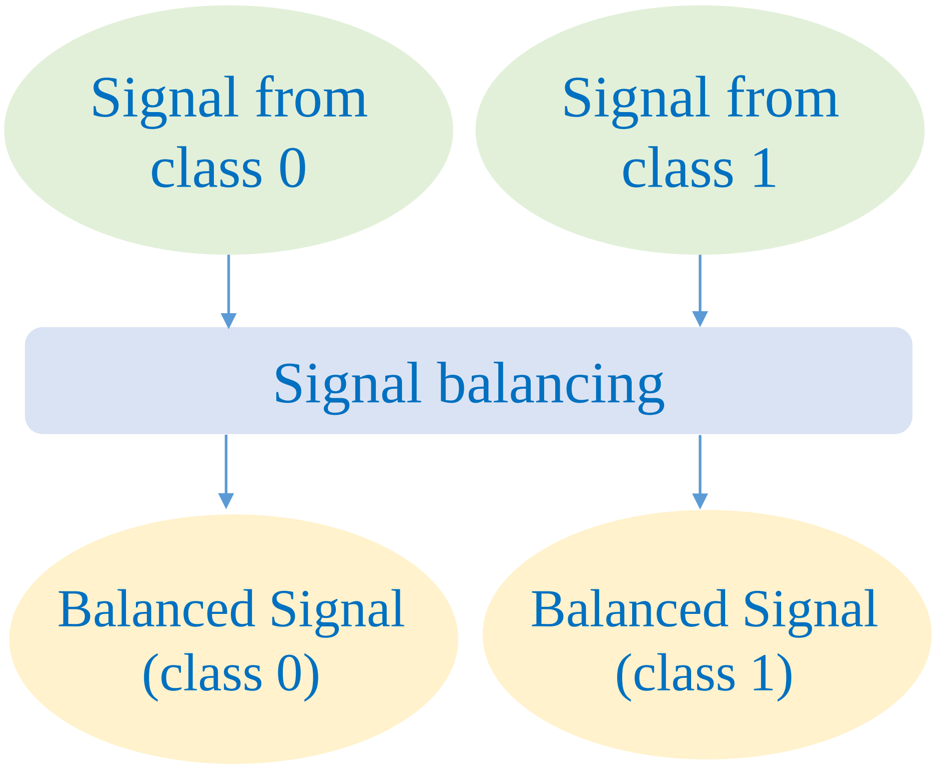
How to solve it!!!

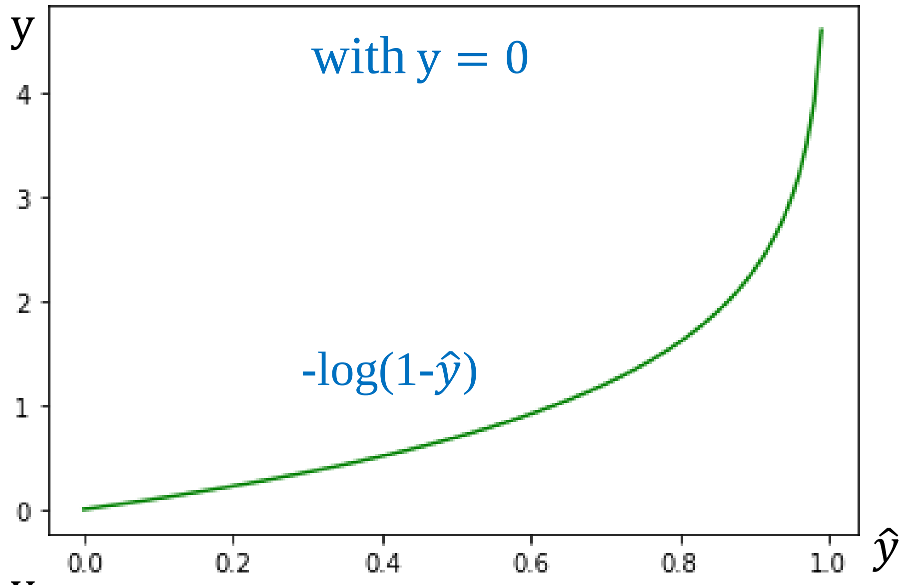
Designing a Function

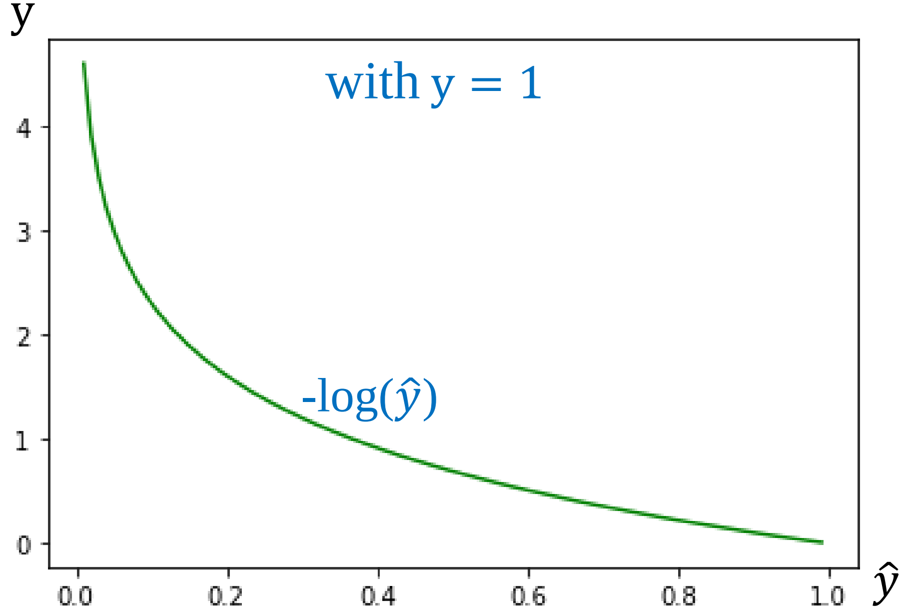
❖ Lie/Truth classification

Feature | Output | Label

Input	Output	Label
…	0.7	0
…	0.8	1
…	0.7	1
…	0.8	1
…	0.8	1
…	0.8	1
…	0.9	1
…	0.9	1
…	0.8	1
…	0.7	1



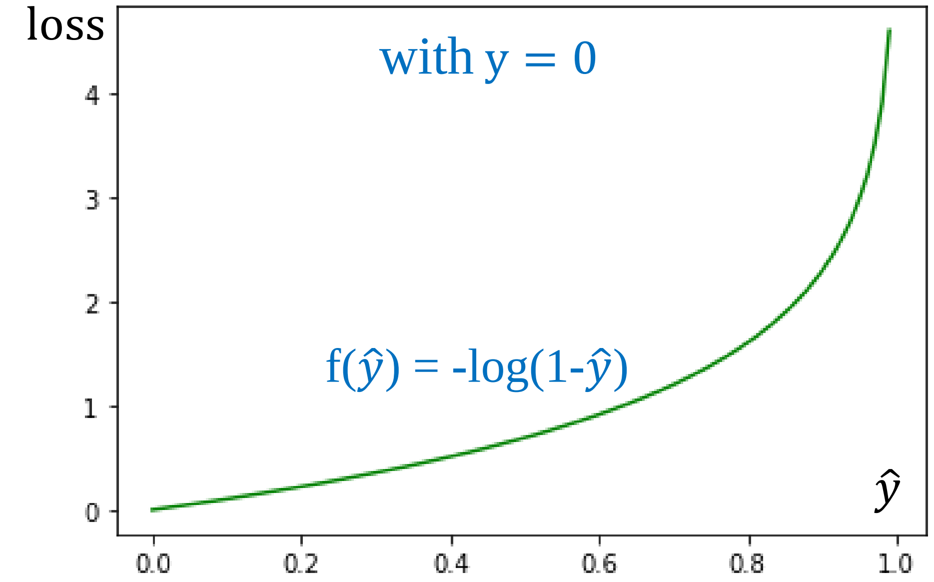


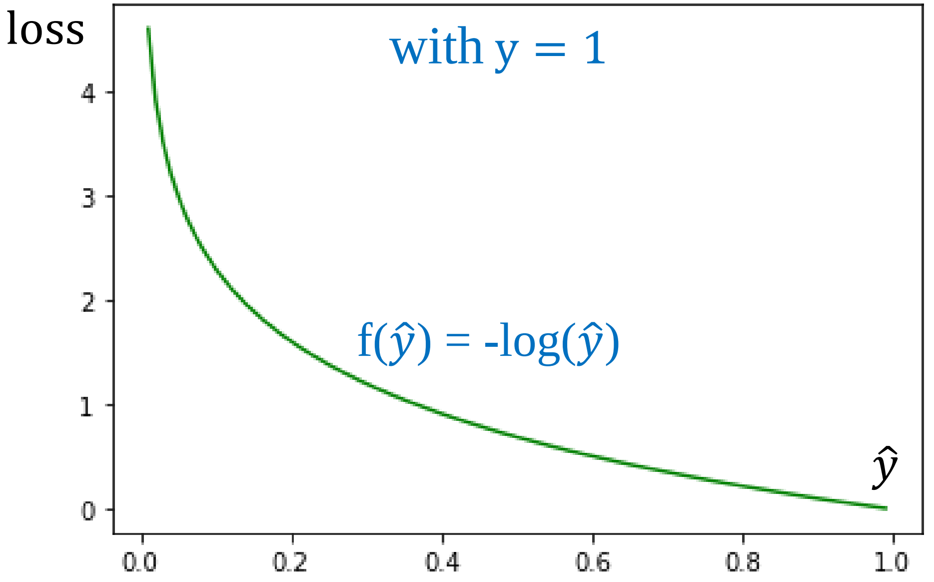


Binary cross-entropy

$$L(y, \hat{y}) = -y\log\hat{y} - (1 - y)\log(1 - \hat{y})$$

Designing a Function

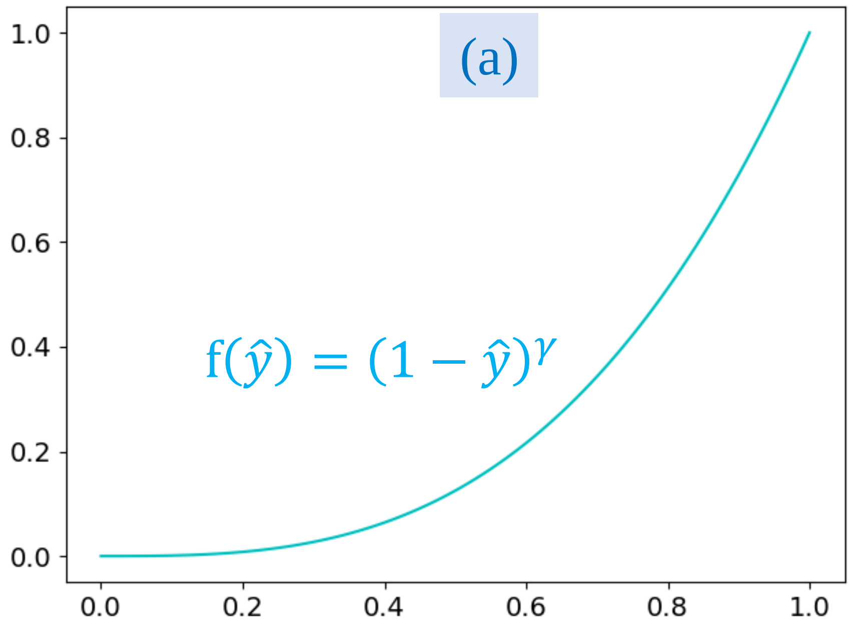


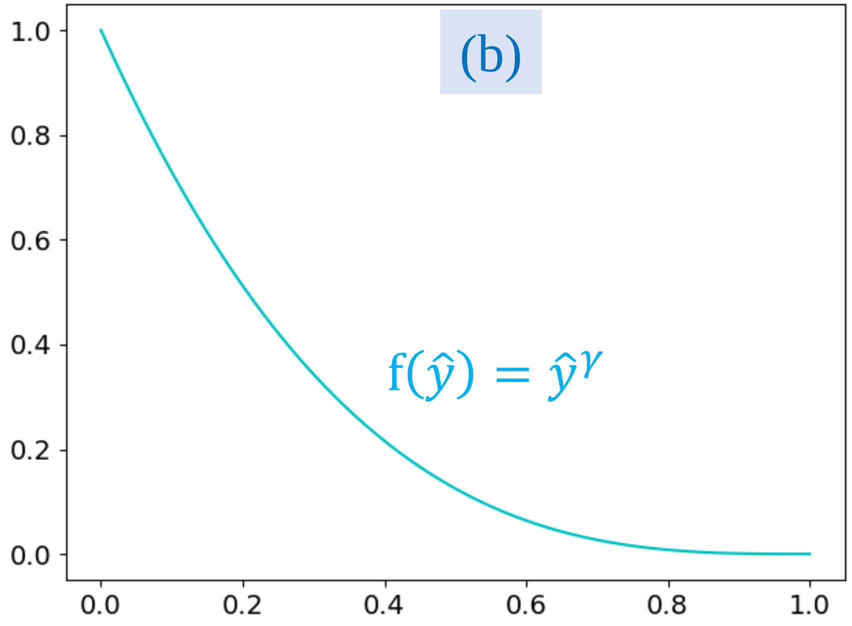


Given $0 \leq k \leq 1$

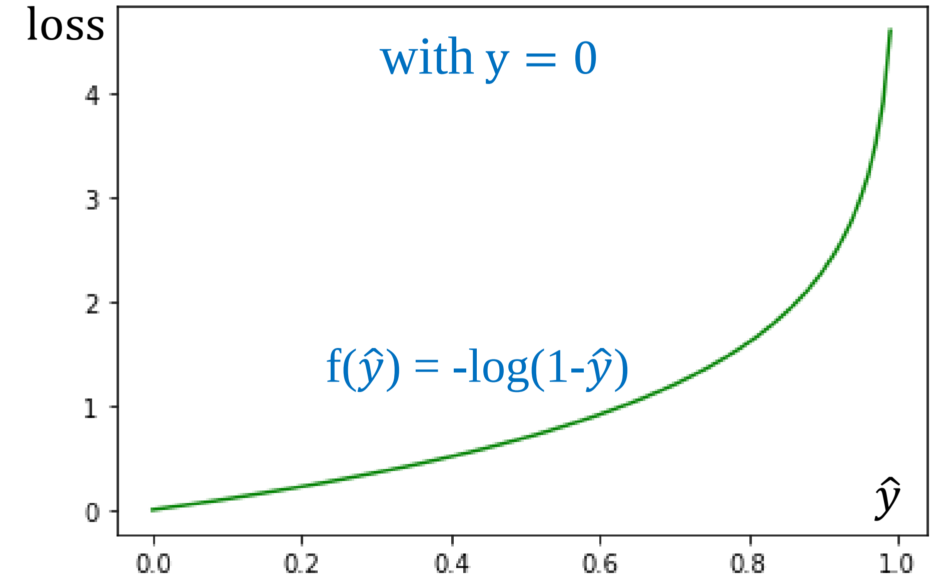
if $f(\hat{y}) * k$
where k approaches 1
→ $f(\hat{y}) * k$ reduces slightly

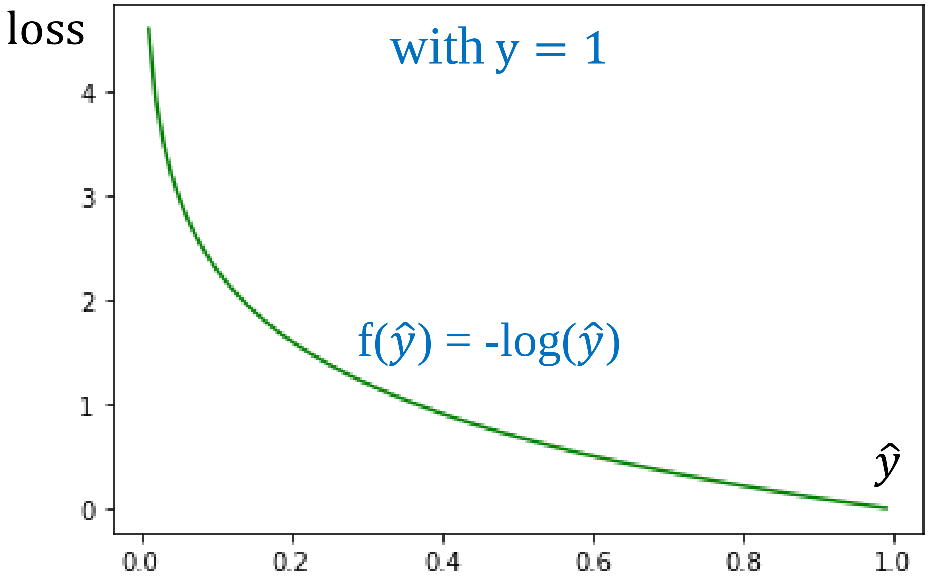
if $f(\hat{y}) * k$
where k approaches 0
→ $f(\hat{y}) * k$ reduces significantly





Designing a Function





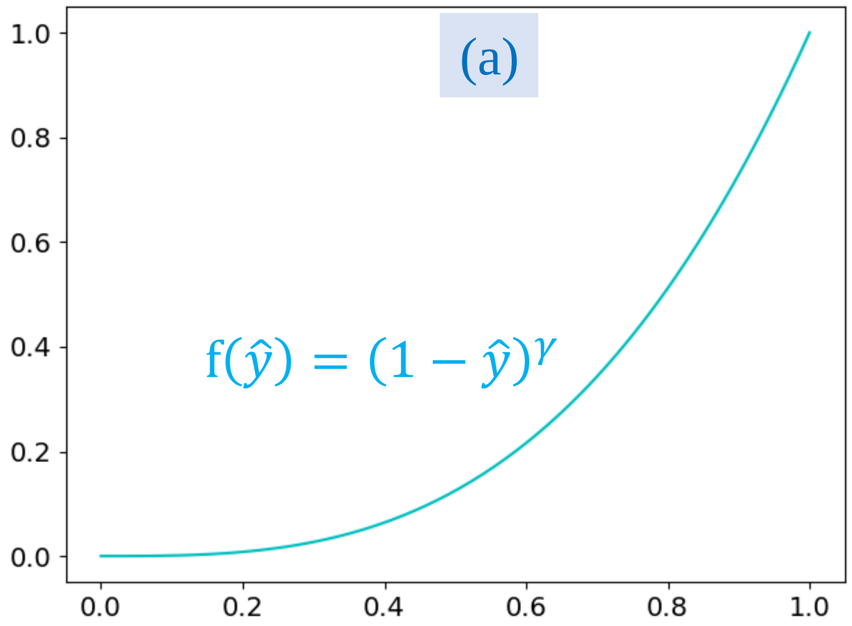
Given $0 \leq k \leq 1$

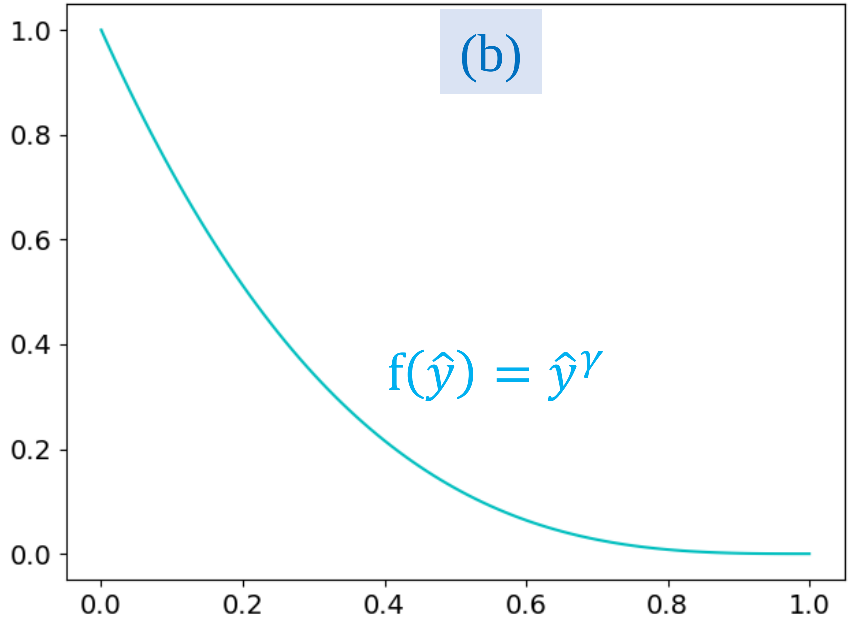
if $f(\hat{y}) * k$
where k approaches 1
$\rightarrow f(\hat{y}) * k$ reduces slightly

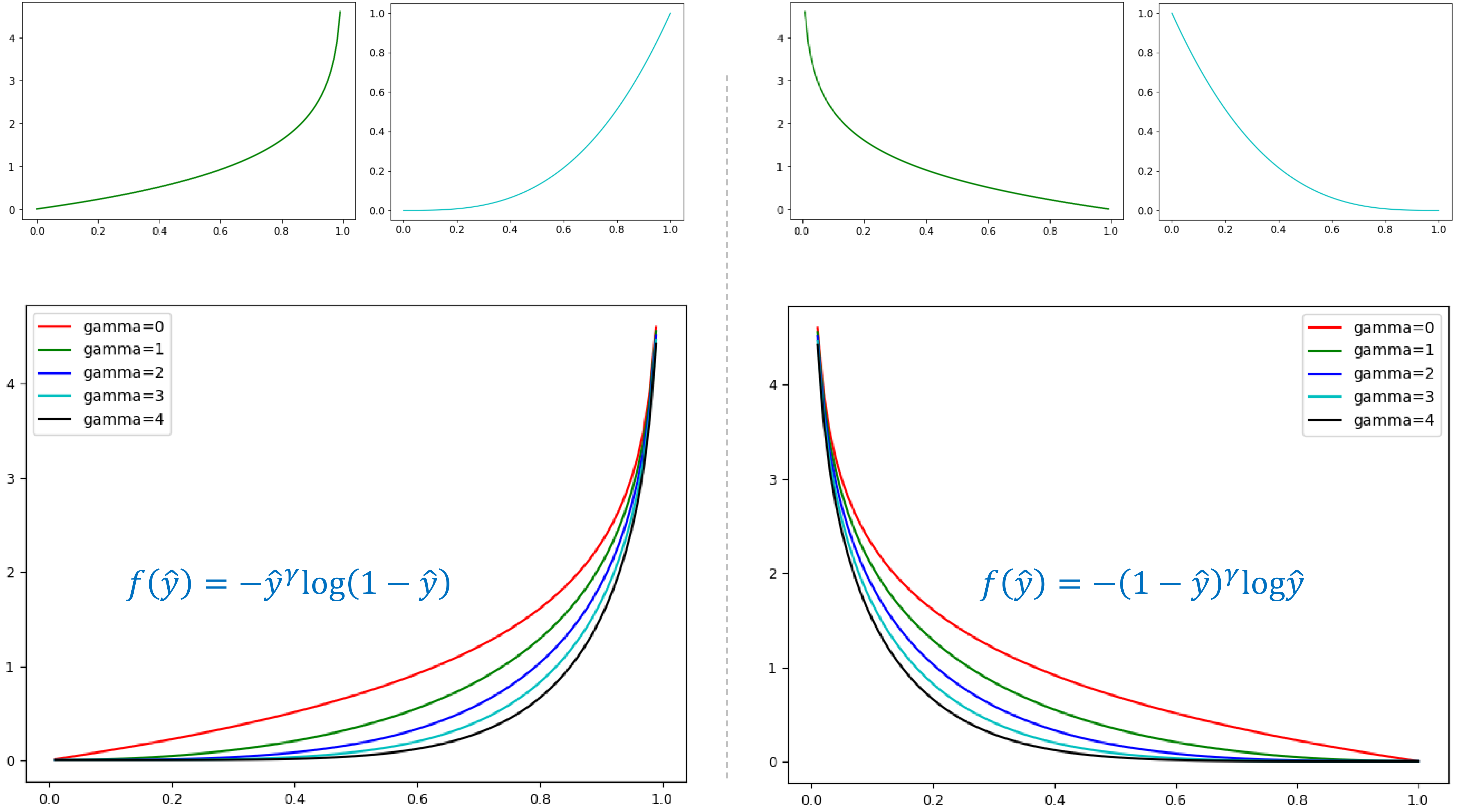
if $f(\hat{y}) * k$
where k approaches 0
$\rightarrow f(\hat{y}) * k$ reduces significantly

Reducing slightly for the correct part

Reducing significantly for the incorrect part







gamma=0
gamma=1
gamma=2
gamma=3
gamma=4
$f(\hat{y}) = -\hat{y}^{\gamma}\log(1-\hat{y})$
gamma=0
gamma=1
gamma=2
gamma=3
gamma=4
$f(\hat{y}) = -(1-\hat{y})^{\gamma}\log\hat{y}$

Applying to our Problem

$$L(y, \hat{y}, \gamma) = -y(1-\hat{y})^{\gamma}\log\hat{y} - (1-y)\hat{y}^{\gamma}\log(1-\hat{y})$$

Input	Output	Label	Gamma=0	Gamma=1	Gamma=2	Gamma=3	Gamma=4
…	0.7	0	1.204	0.842	0.589	0.412	0.289
…	0.8	1	0.223	0.044	0.008	0.001	0.0003
…	0.7	1	0.356	0.107	0.032	0.009	0.002
…	0.8	1	0.223	0.044	0.008	0.001	0.0003
…	0.8	1	0.223	0.044	0.008	0.001	0.0003
…	0.8	1	0.223	0.044	0.008	0.001	0.0003
…	0.9	1	0.105	0.011	0.001	0.0001	0.00001
…	0.9	1	0.105	0.011	0.001	0.0001	0.00001
…	0.8	1	0.223	0.044	0.008	0.001	0.0003
…	0.7	1	0.356	0.107	0.032	0.009	0.002

$loss_{y=1}$: 2.039 0.458 0.111 0.028 0.007

Focal Loss for Dense Object Detection, ICCV, 2017.