KAIST 교육 9년 동안의 이야기

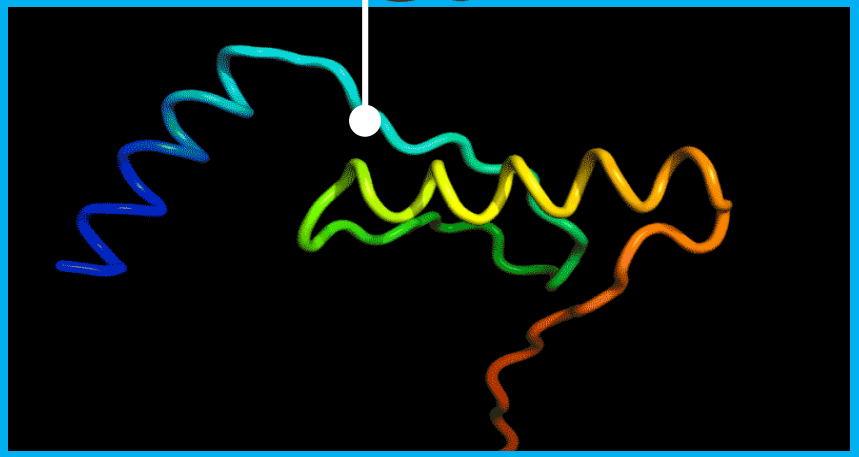
서창호
전기 및 전자공학부

2021년 5월 26일 교육혁신의 날 행사

Google assistant

AI

자율주행



단백질 구조 예측

AI fundamentals

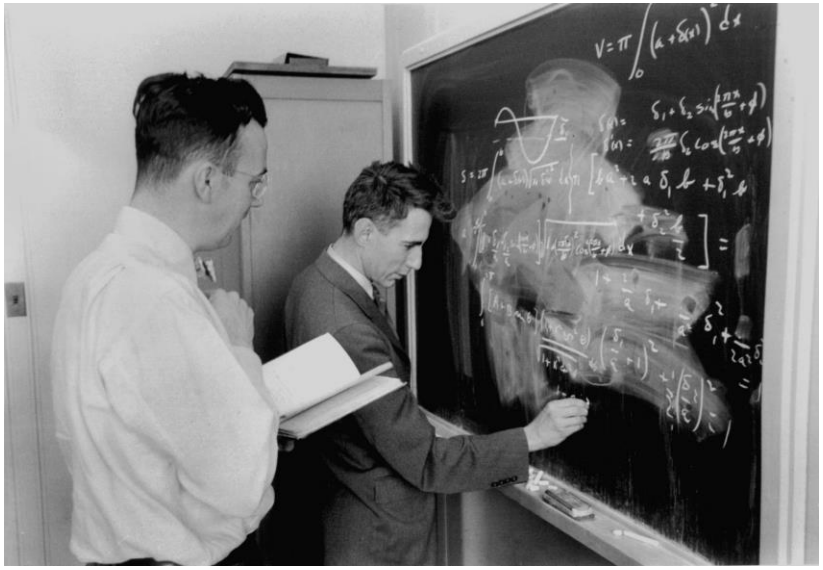
최적화

확률

정보이론

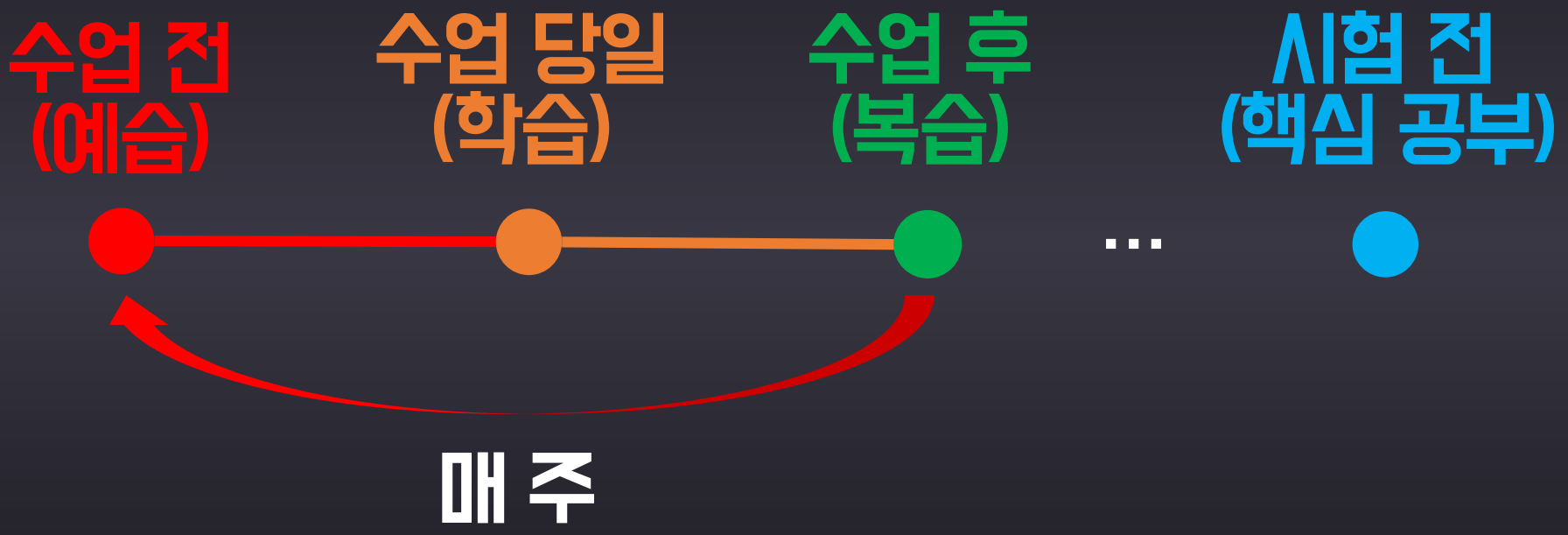
통신

융합 수업



이론

구현


수업 전
(예습)
수업 당일
(학습)
수업 후
(복습)
...
시험 전
(핵심 공부)
매 주

Course notes

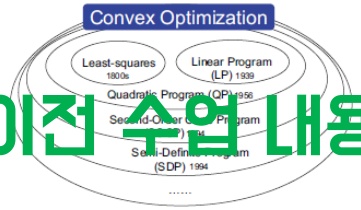
EE424 Introduction to Optimization
KAIST, Fall 2020

September 3, 2020
Changho Suh (chsuh@kaist.ac.kr)

Lecture 2: Definition of Convex Optimization

Recap

Last time, I told you a story of how the optimization theory was developed. There were two breakthroughs in the history of optimization. The first was made by the famous Gauss. In the process of solving an astronomy problem of figuring out the orbit of Ceres (which many astronomers were trying to address in the 1800s), he could develop an optimization problem, which is now known as the least-squares problem. The beauty of the least-squares problem is two-folded: (i) it has a closed form solution; (ii) there is an algorithm which enables computing the matrix inverse efficiently which is required to compute the solution. It turned out the beauty of the problem opened up the optimization field and has played a significant role in the field.



이전 수업 내용

Figure 1: A class of tractable optimization problems: Convex optimization.

The second breakthrough was made by Leonid Kantorovich. In the process of solving a military-related resource-allocation problem, he could formulate a problem which is now known as linear program (LP). The good thing of LP is that there developed an efficient algorithm which allows us to compute the optimal solution reliably and efficiently although the closed form solution is unknown. In other words, Kantorovich came up with the concept of *tractable* optimization problems which can be solved via an algorithm without the knowledge of the optimal solution form. This motivated many followers to mimic his approach, thereby coming up with a class of tractable optimization problems: convex optimization; see Fig. 1.

Today's lecture

The goal of today's lecture is to understand what convex optimization is. To this end, we will cover four [illegible]. First we will [illegible] standard form formulation of optimization problems. Terms about the definition of *convex optimization problems* as of our main interest require the knowledge of *convex functions*. Both the definition of convex functions relies on the concept of *convex sets*. So in the second part, we will study what the convex set is and also investigate some important examples in an effort to be familiar with the concept. Next we will study the definition of convex functions together with a couple of examples and crucial

이번 수업 내용

1

...

Previously we investigated examples of convex sets where only affine functions are introduced. Actually there are many convex sets which concern convex functions. Here we list a couple of such examples.

One such example is:

$$S = \{x : f(x) \leq 0\} \tag{13}$$

where $f(x)$ is a convex function. Here is the proof that S is a convex set. Suppose $x, y \in S$. Then, $f(x) \leq 0$ and $f(y) \leq 0$. This together with the convexity of f, reflected in the condition (9), gives:

$$f(\lambda x + (1-\lambda)y) \leq 0,$$

which in turn implies that $\lambda x + (1-\lambda)y \in S$. This completes the proof.

Another example is the intersection of such convex sets:

$$S = S_1 \cap S_2$$
$$S_1 = \{x : f_1(x) \leq 0\}, \quad S_2 = \{x : f_2(x) \leq 0\}. \tag{14}$$

Try the proof in Problem Set (PS) 1. Actually the intersection of arbitrary convex sets is also convex - check in PS1 as well.

Convex optimization problem in standard form

We are now ready to define the convex optimization problem. It is an optimization problem which satisfies the following three: (i) The objective function is convex; (ii) The set induced by inequality constraints is convex; and (iii) The set induced by equality constraints is convex. So the standard form of the convex optimization problem is (2) in which (i) $f(x)$ is convex; (ii) $f_i(x)$ is convex; and (iii) $h_i(x)$ is affine. Notice that the set induced by affine equality constraints

Look ahead

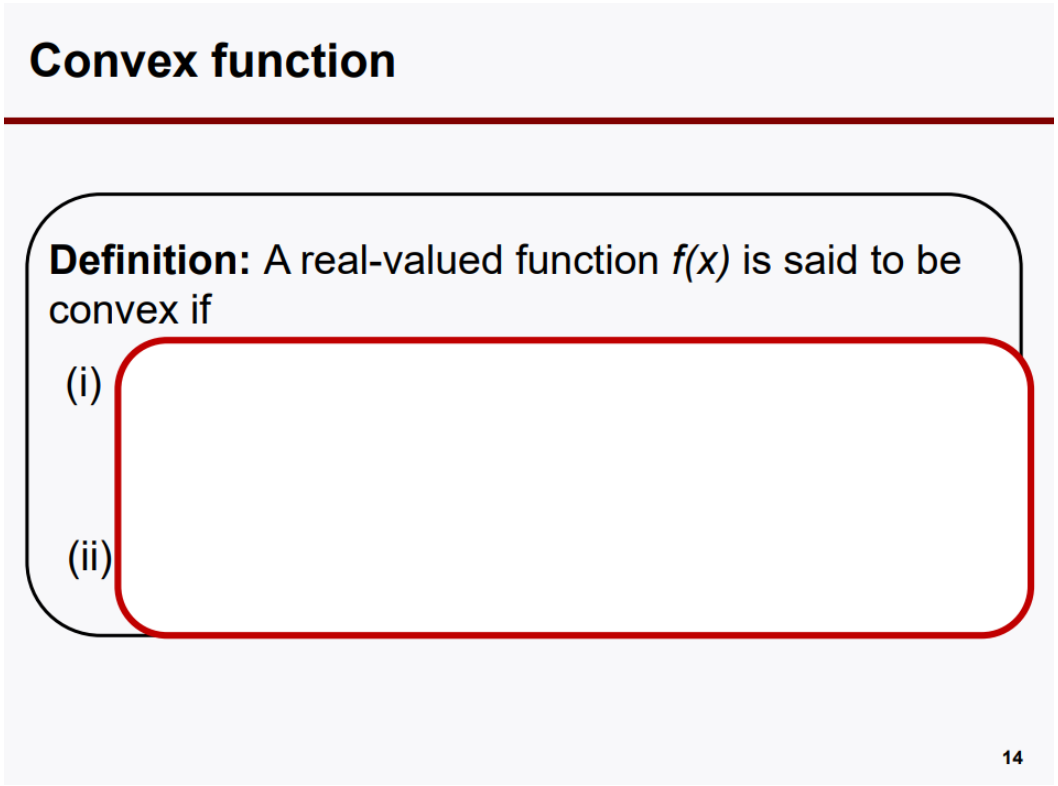
Of course t[illegible] on why [illegible] optimization [illegible] in such a manner. Th[illegible] is be[illegible] way of d[illegible] m[illegible] the problem *tractable*. [illegible] time, we will provide an i[illegible] as [illegible] convex [illegible] is tractable. We will [illegible] investigating one instanc[illegible] ex[illegible] tion: Linear Prog[illegible].

다음 수업 내용

6

예습을 위한 6~7 pages 노트 제공

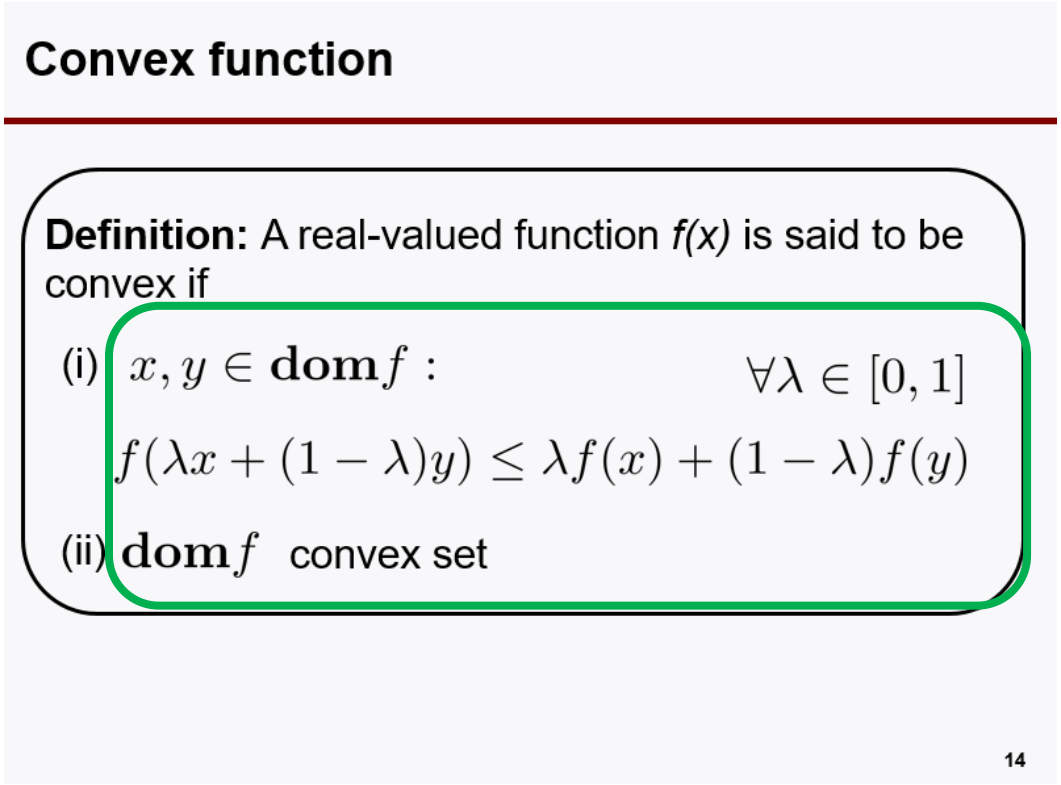
Preview slides



수업 전 **빈칸**으로 구성된 슬라이드 제공

→ 능동적 수업참여 유도

Lecture slides



Convex function

Definition: A real-valued function *f(x)* is said to be convex if

(i) $x, y \in \mathbf{dom} f$: $\forall \lambda \in [0,1]$

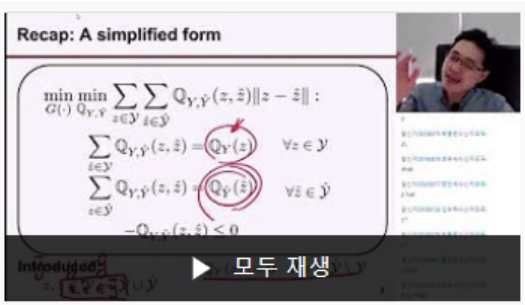
$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)$$

(ii) $\mathbf{dom} f$ convex set

14

수업 후 정답이 채워진 슬라이드 제공

Lecture videos



EE424: Introduction to Optimization Techniques

동영상 107개 • 조회수 4,785회 • 최종 업데이트: 2020. 12. 3.

일부 공개

EE424 Introduction to Optimization techniques 구독중

00 Lecture 2: Announement&Recap&Introduction 15:06
EE210 Probability and Introductory Random Processes

이전 수업 내용

01 Lecture 2: Part I. Mathematical formulation of optimization problems 12:00
EE210 Probability and Introductory Random Processes

02 Lecture 2: Part II. Convex set 25:29
EE210 Probability and Introductory Random Processes

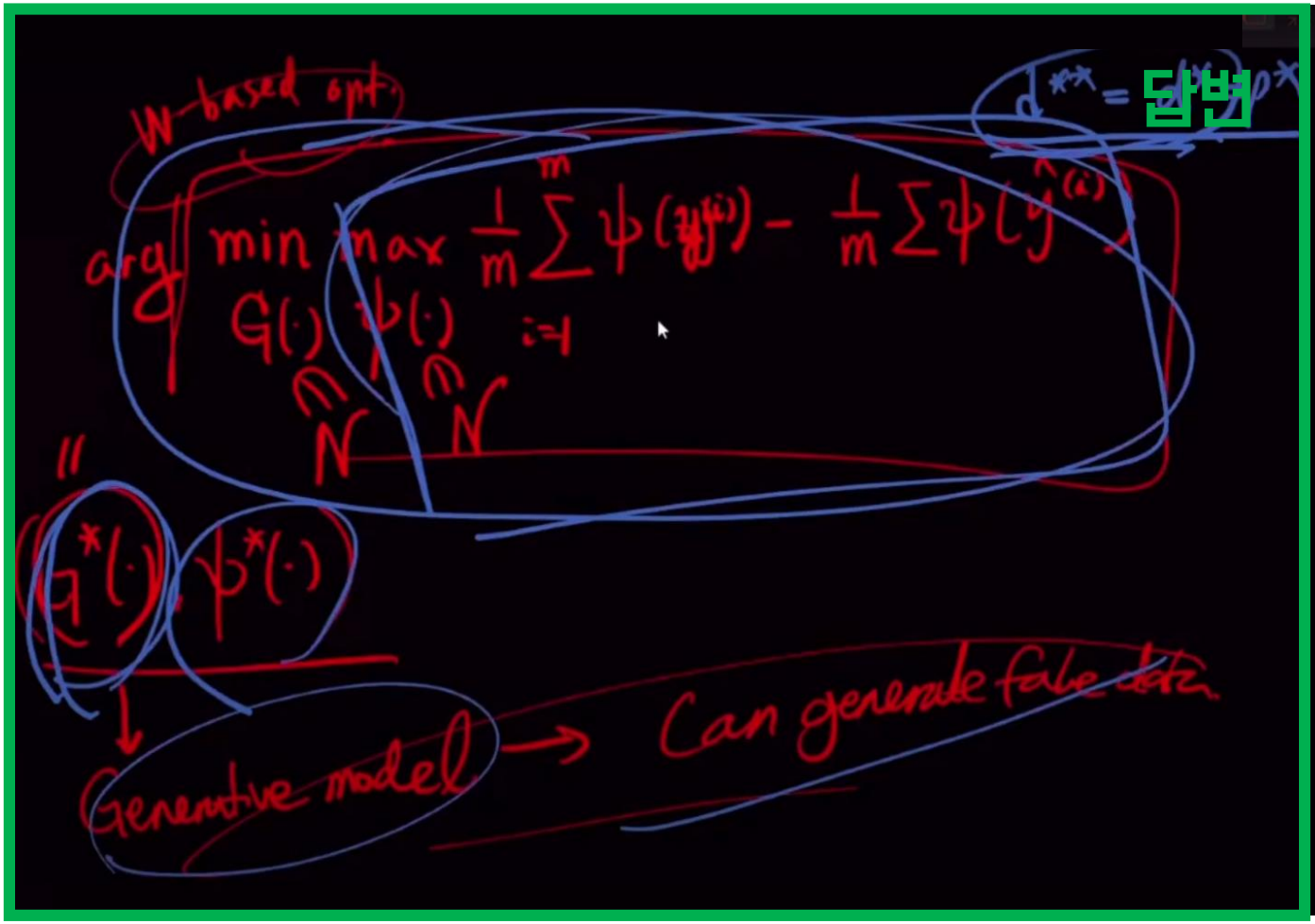
이번 수업 내용

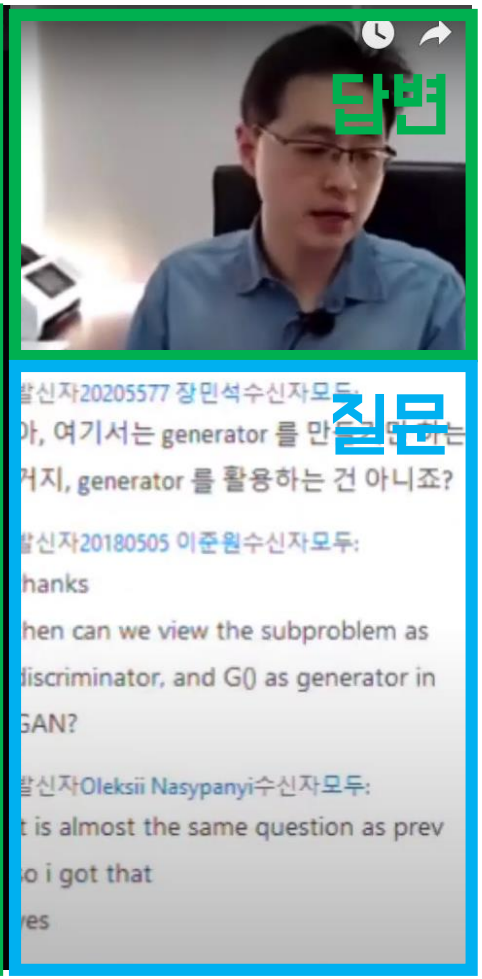
03 Lecture 2: Part III. Convex function 17:24
EE210 Probability and Introductory Random Processes

04 Lecture 2: Part IV. Definition of convex optimization 7:17
EE210 Probability and Introductory Random Processes

다음 수업 내용

효율적인 복습을 위한 파트별 편집 동영상 제공





양방향 인터랙션을 통한 효율적 비대면 강의

Problem sets

EE424 Introduction to Optimization
KAIST, Fall 2020

September 3, 2020
Changho Suh (chsuh@kaist.ac.kr)

Problem Set 1

Due: September 11 (Friday)

Note 1: Submit your solution (a soft copy) to klms system (PS1 tap is created in our website).

Note 2: Each problem will be graded by a different TA (indexed by TA1, TA2, TA3):

TA1: Sihyung; TA2: Jongseong; TA3: Jinyeop

1. (TA1) *(Least Squares)* Let $A := [a_1 \ \cdots \ a_m]^T$ and $b := [b_1 \ \cdots \ b_m]^T$ where $a_i \in \mathbf{R}^d$ and $b_i \in \mathbf{R}$, $i \in \{1, \ldots, m\}$.

(*a*) Consider a function $f : \mathbf{R}^d \to \mathbf{R} : f(x) = \|a_1^T x - b_1\| + \cdots + \|a_m^T x - b_m\|$. A student claims that the function f can be represented as:

$$f(x) = \|Ax - b\|. \tag{1}$$

Prove or disprove the claim.

(*b*) Consider another function $f : \mathbf{R}^d \to \mathbf{R}$:

$$f(x) = \|Ax - b\|^2. \tag{2}$$

(Only) using the definition of a convex function, show that $f(x)$ is convex in x.

1. 수업내용 이해도 향상을 위한 관련 문제 제출

Problem sets

6. (TA1) *(Monty Hall Problem – Python lab)* In this problem, you are asked to perform some simulations to empirically verify what we proved in the Monty Hall Problem:

$$\mathbb{P}(\text{win w/ sticking}) = \frac{1}{3}; \tag{2}$$

$$\mathbb{P}(\text{win w/ switching}) = \frac{2}{3}. \tag{3}$$

For your sake, let us repeat the game procedure. Suppose there are three doors. The prize "car" is behind one door, but it is unknown to the trader, while being revealed to the host (Monty Hall). Behind the other two doors there are two goats (sort of "qquang"). The trader is first asked to choose one out of three doors. The host then opens one door behind which there is a goat. Next, the trader is given an option between *sticking* with the initial choice vs *switching* to another unopened door. Let N be the number of games tried out in simulation.

(a) Suppose the trader takes the *sticking* strategy. Implement a Python function that returns 1 (for winning) or 0 (for losing) for one game.

Hint: You may want to use functions like ***random.randint****.*

(b) Using the python function in part (a), plot the empirical winning rate ($= \frac{\text{number of winnings}}{N}$) as a function of N. How does the winning rate converge in light of (2) as N grows?

(c) Now suppose the trader takes the *switching* strategy. Repeat parts (a) and (b). Of course, in this case, you should compare to (3).

Solution:

(a) In sticking strategy, set *switch* as false and *num_doors* as 3. With *random.randint(0, num_doors-1)*, we randomly choose the location of car and player between 0, 1 and 2. In this problem, we don't care about switching. If the index of player is equal to the index of car, the program code returns 1, otherwise 0.

In [1]:
```
import random
import math
import matplotlib.pyplot as plt
```

In [2]:
```
def open_doors(num_doors: int, door_car: int, door_player: int) -> int:
    '''
    num_doors: number of doors
    door_car: the index of the door with car behind
    door_player: the index of the door which player chose
    '''

    if door_player != door_car:
        return door_car
    else:
        return (door_player + 1) % num_doors
```

In [3]:
```
def game(switch: bool, num_doors: int) -> int:
    '''
    switch: whether the player switch or not
    num_doors: number of doors
    '''

    doors = [x for x in range(num_doors)]    # index of doors
    door_car = random.randint(0, num_doors-1)   # index of the door with car behind
    door_player = random.randint(0, num_doors-1)     # index of the door player chose

    door_left_closed = open_doors(num_doors, door_car, door_player)

    if switch:
        door_player = door_left_closed

    return 1 if door_player == door_car else 0
```

2. Python/TensorFlow 활용 코딩 문제 제출

Exam guidelines

EE321 Communication Engineering
KAIST, Spring 2020

June 8, 2020
Changho Suh (chsuh@kaist.ac.kr)

Final instructions

Logistics: The exam runs live online via zoom, as promised. It starts from 12:50 pm and ends at 2:10 pm on June 17 (Wednesday), 2020. You are allowed to use one cheating sheet, A4-sized and double-sided. The total score is 100 + 10 points. The maximum score that you can get is 100 points though. The overflowing points (if any) would contribute to compensating for a loss (if any) that occurred in your midterm exam. Please show all of your work in details to maximize chances for partial credits. For details, see "final notice" that would be uploaded by the head TA (Minguen Kang) on KLMS.

Contents: The best preparation for the final is to carefully review all of the lecture slides, course notes, problem sets, exercise problems and related reference. I strongly recommend this as a first-order preparation tool. Here is a summary of the key topics covered in this course as well as the corresponding references.

(*a*) A digital communication architecture
CN1, LS1.
Prob 4,5 in PS1.

(*b*) Review of probability laws and random variable
Chapters 1 ∼ 3 in BT.
Prob 1 ∼ 3 in PS1; Prob 1, 2 in PS2; Prob 1 ∼ 3 in PS3; Prob 1 in PS4; Prob 1 ∼ 5 in EP1.

(*c*) Gaussian noise model, PAM, optimum receiver principles, error probability analysis
CN2 ∼ 6, LS2 ∼ 6.
Prob 6 of PS1; Prob 3 ∼ 6 in PS2; Prob 4 ∼ 7 in PS3; Prob 2 in PS4; Prob 6 in EP1.

(*d*) Sequential communication and repetition coding
CN7 ∼ 8, LS7 ∼ 8.
Prob 5 ∼ 7 in PS3, Prob 2 ∼ 6 in PS4; Prob 7 in EP1; Prob 1 in PS5.

(*e*) Capacity of the AWGN channel
CN9, LS9.
Prob 7 in PS4; Prob 8 in EP1.

(*f*) Waveform shaping
CN10 ∼ 11, LS10 ∼ 11.
Prob 2 ∼ 3 in PS5.

(*g*) ISI channel modeling
CN12 ∼ 13, LS12 ∼ 13.
Prob 4 ∼ 5 in PS5.

(*h*) Viterbi algorithm
CN13 ∼ 14, LS13 ∼ 14.
Prob 6 in PS5; Prob 1 in PS6.

시험 전 어느 부분을 공부 해야하는지 알려줌

→ 중요한 부분을 효율적으로 공부함

Textbooks

출판 승인

최적화

정보이론

출판 리뷰 중

확률

통신

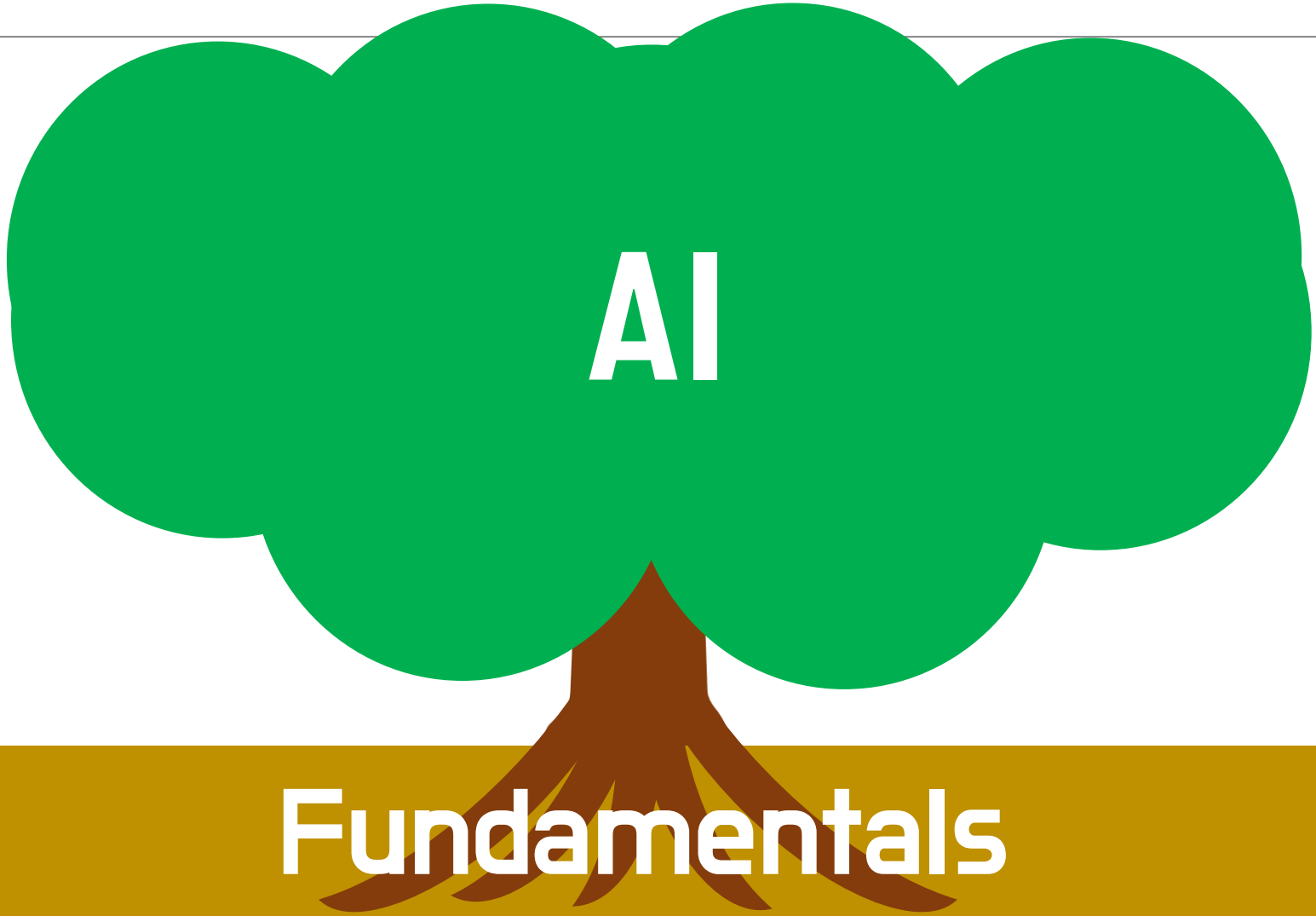
교육에 대한

세 가지 생각

"교육은 양동이를 채우는 것이 아니라
불을 지피는 것이다."

1. 동기 부여 최우선

2. 사례 중심교육




유퀴즈!
제106화 N주년
개교 50주년 카이스트 총장
갓구운클립

3. 존중