Machine Learning

INFQ612L, 440113450A

Spring Semester

Friday 17:00-18:40

IPS

WASEDA University

Prof. Shoji Makino

Machine Learning

Friday 17:00-18:40

1. 4/18
2. 4/25
3. 5/2
4. 5/9
5. 5/16
6. 5/23
7. 5/30
8. 6/6
9. 6/13
10. 6/20
11. 6/27

-. 7/4 No Lecture

-. 7/11 No Lecture

-. 7/18 No Lecture

-

At Zoom, set your name as:

Student ID, LAST_NAME, First_name

44251234, MAKINO, Shoji

At my class,

please turn on your camera

Machine Learning (4)(5)

Multiple regression

Flow of Machine Learning

Data

Toward the first budget compilation under the new era "Reiwa", the national fiscal system council will consider measures to curb spending such as social security expenses as the budget scale continues to expand. We have set up a subcommittee to have intensive discussions.

The Financial System Council, which proposes to the Minister of Finance what the budget should be, began discussions on the 4th for the first budget formulation for the next fiscal year under "Reiwa".

Features (vector representation of data)

$$X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\},$$
$$\boldsymbol{x}_i \in \mathbb{R}^D$$

Machine learning (supervised, unsupervised):
feature–to–concept mapping

Acquisition of concepts

Spam, Cat, Political News, etc.

Document to Feature Vector: Bag-of-words

Document

In recent years, there has been a growing trend towards outsourcing of computational tasks with the development of cloud services. We propose two building blocks that work with FHE: a novel batch greater-than primitive, and matrix primitive for encrypted matrices

Dictionary (12,000 words)

ID	word
1168	batch
1169	bath
...	
1201	cloud
...	
1239	computation
1240	computational
...	
1172	primitive

Feature vector is characterized by whether or not it appears without considering frequency

$$\boldsymbol{x}_i = (0, 0, \ldots, 0, 1, 0, 1, 0, \ldots, 0, 1, 0, \ldots, 0, 1, 0, \ldots)$$

Document to Feature Vector

Morphological analysis is required for Japanese

TF (term freq.) -IDF (inv. doc. freq.)

- Instead of "0" and "1" in the Bag-of-words vector, enter a value that reflects the frequency of occurrence of words and the rareness of words
- N documents $D = \{d_1, d_2, \ldots, d_N\}$
  - n_{ij} : Number of occurrences of the word *i* in the *j*th document
- TF (term freq.) $\mathrm{tf}_{ij} = \frac{n_{ij}}{\sum_{d_k \in D} n_{kj}}$ Appearance rate of word i in document j
- IDF(inv. doc. freq.) $\mathrm{idf}_i = \frac{N}{|\{d_i \in d, d \in D\}|}$ Rareness of word i
- TF/IDF $\mathrm{tfidf}_{ij} = \mathrm{tf}_{ij} \times \mathrm{idf}_i$ Importance of the word i in the document j

Image to Feature Vector

- For the recognition target, leave the necessary information and delete the unnecessary information

- Facial recognition ⇒ luminance value feature
  - Effective: appearance of the face itself
  - Unnecessary: image brightness
    (brightness changes depending on lighting)

- Object recognition ⇒ higher-order local autocorrelation features
  - Effective: appearance, color, shape, number
  - Unnecessary: position of the object

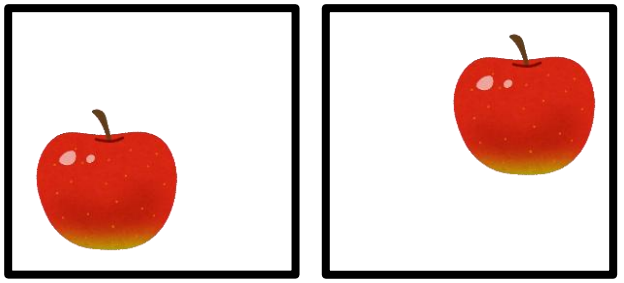
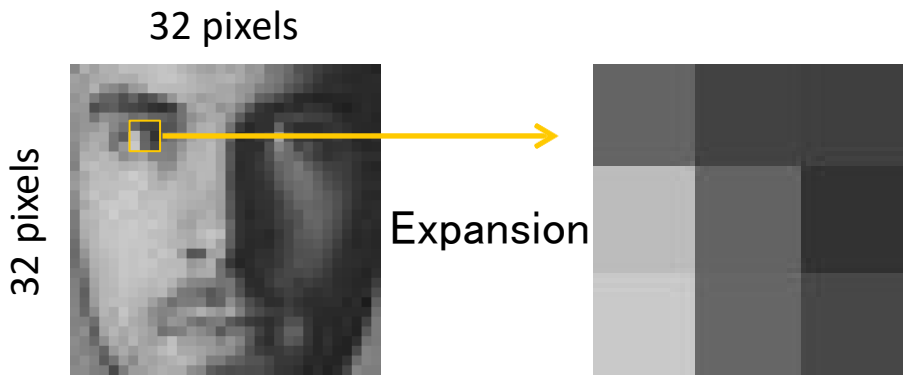


Image to Feature Vector

- Luminance value features
  - Called appearance-based image recognition
  - Use the appearance of the image as it is
  - Pixel x Pixel dimension feature vector

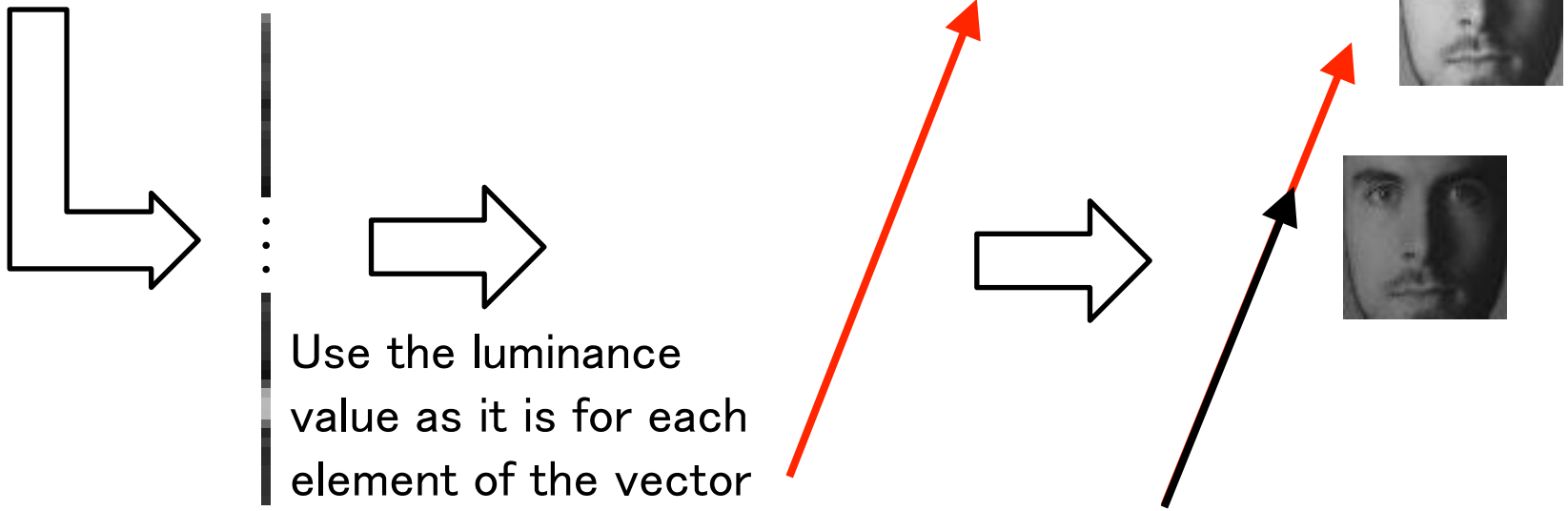
- Higher-order local autocorrelation features
  - Has characteristics suitable for image recognition such as position invariance, additivity, low dimension, high speed, and robustness
  - Various applications (image impression evaluation, gesture recognition, etc.) and extensions (color images, moving images, etc.) have been proposed

Luminance Value Features



100	58	67
180	123	38
202	91	80

Each pixel contains a value from 0 (black) to 255 (white)



Use the luminance value as it is for each element of the vector

1024 pixels × 1 pixel

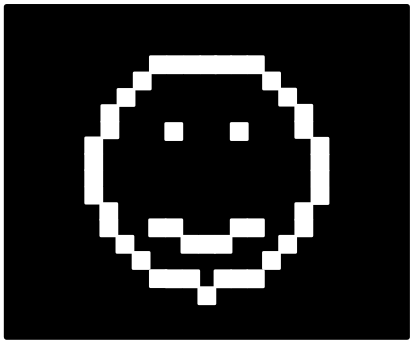
1024 dimensional feature vector

L2 norm normalization

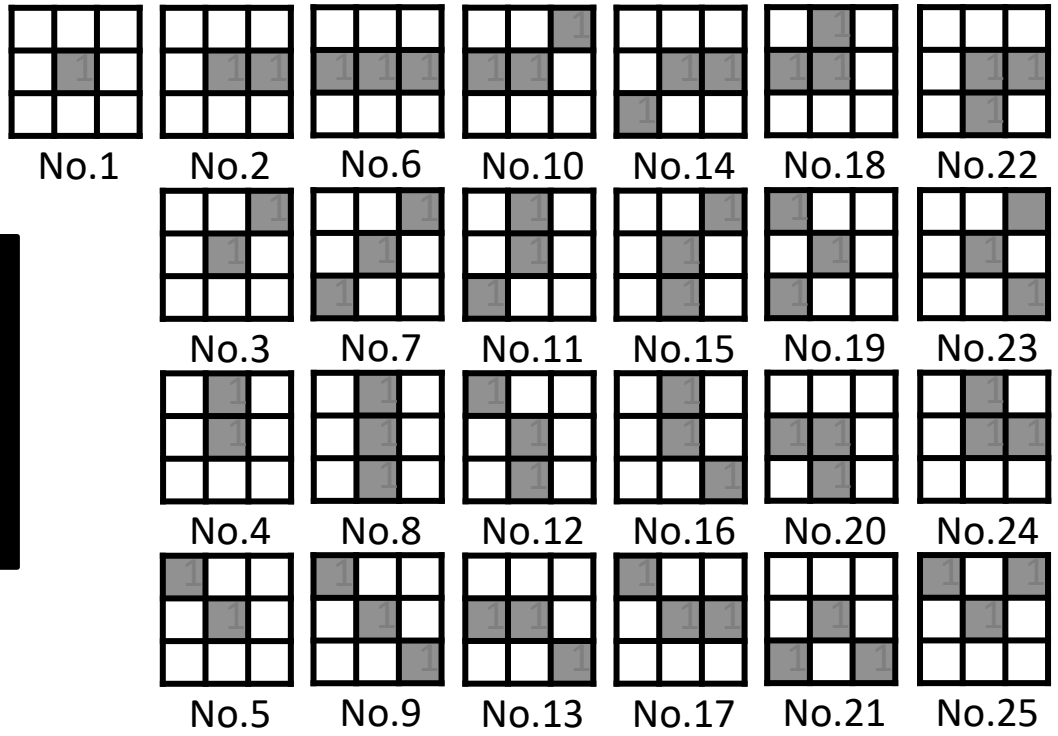
(Make the luminance of the image constant)

Higher-order Local Autocorrelation Features (HLAC)

- For each pixel, calculate the product of the element values that overlap the gray part of the mask, scan the entire screen, and calculate the sum
- It becomes a 25-dimensional feature vector regardless of the size of the image

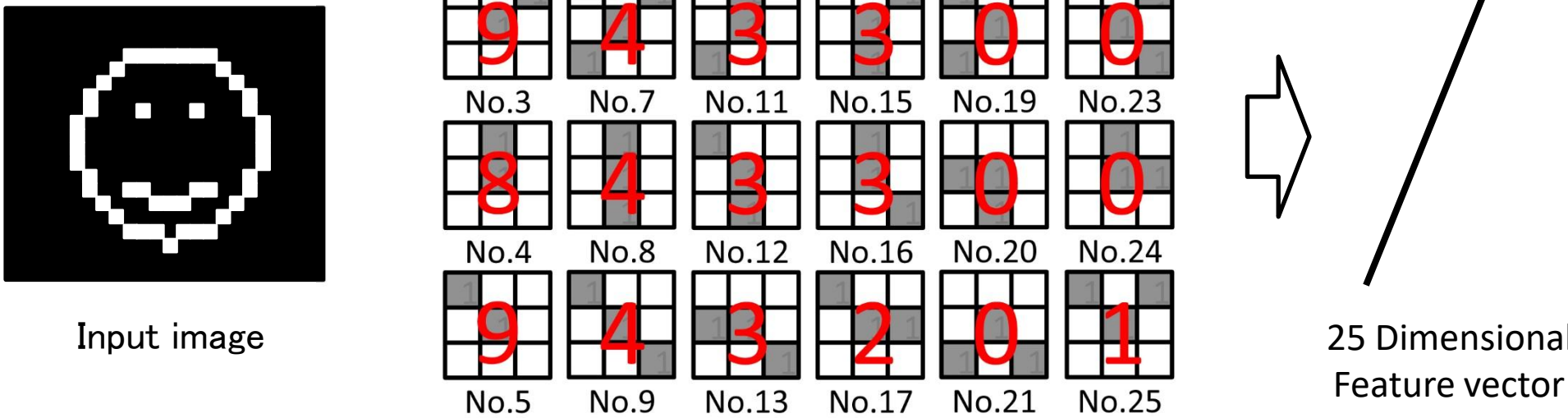


Input image



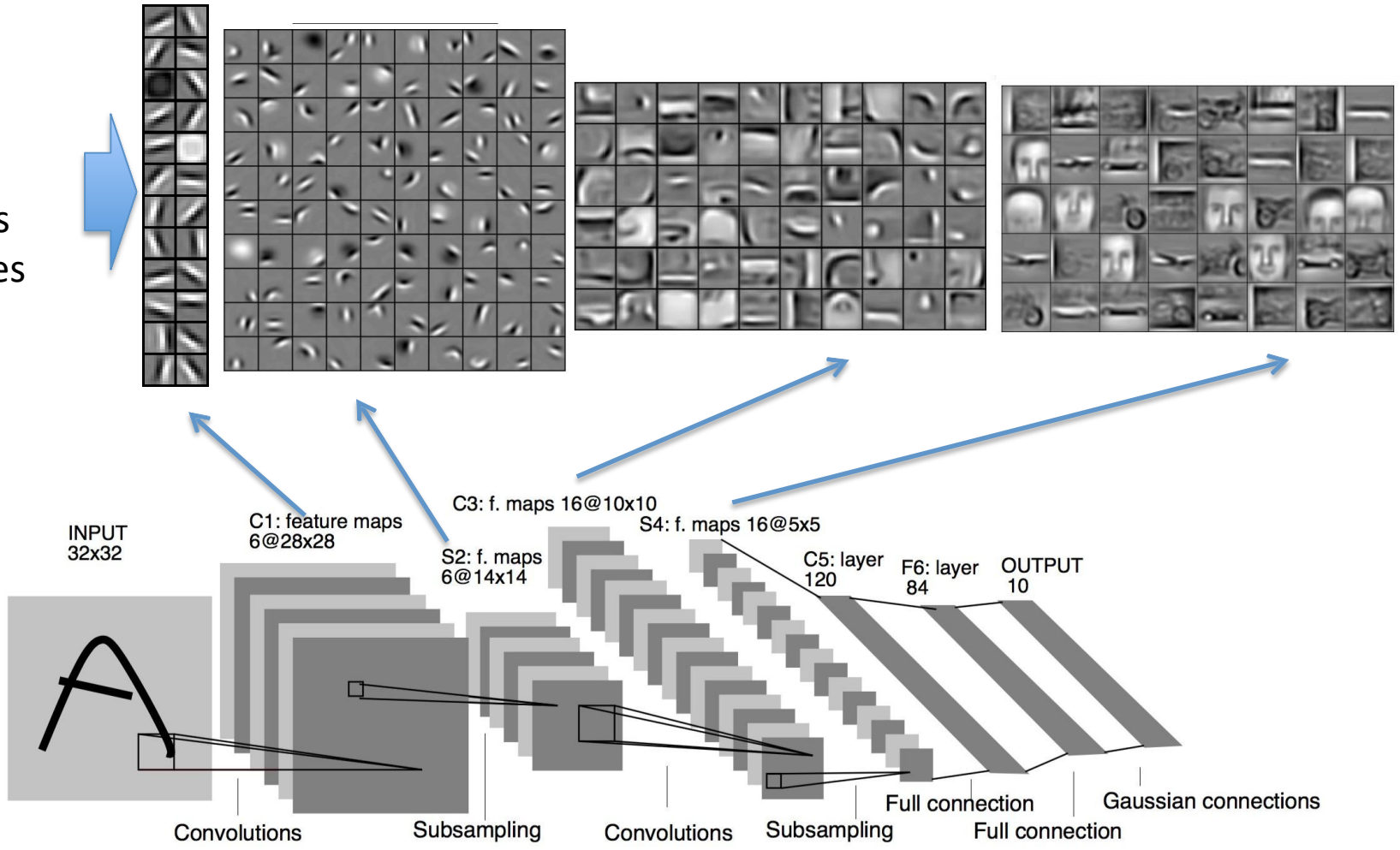
HLAC feature calculation method (for binary images)

- In the case of a binary image, it is equivalent to counting how many masks (straight lines, curves, etc.) are in the image



Feature Discovery by Deep Learning

Faces
Cars
Airplanes
Motorbikes



Recognition results

Lee et al. "Convolutional DBNs for Scalable Unsupervised Learning of Hierarchical Representations" ICML09

Flow of Machine Learning

Data

established a new subcommittee to consider measures to curb spending, such as social security expenses, as the budget scale continues to expand, and discussions will be held intensively.

The Council on Fiscal Systems, which recommends the state of the budget to the Minister of Finance, began discussions on the budget for the next fiscal year, which will be the first under the Reiwa from April 4.。

Deep learning

Features (vector representation of data)

$$X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\},$$
$$\boldsymbol{x}_i \in \mathbb{R}^D$$

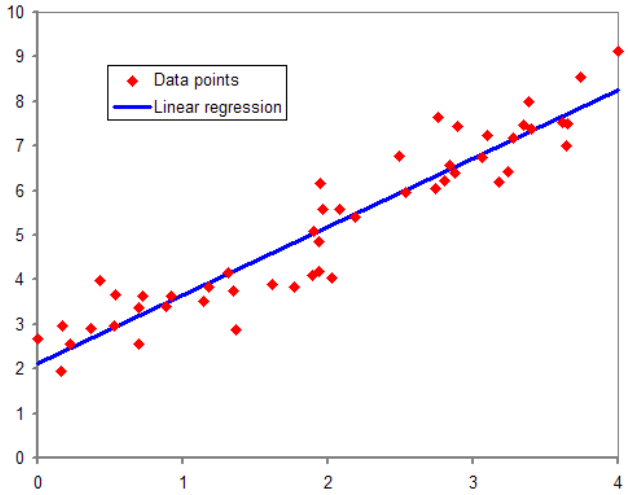
(shallow) machine learning

Machine learning
(supervised, unsupervised)
Feature-to-concept mapping

Acquisition of concept

Spam, cats, political news, etc.

Linear Regression (Multiple regression)



- Features (independent variables)

$$\boldsymbol{x}_i \in \mathbb{R}^D$$

- Target value (dependent variable)

$$t_i \in \mathbb{R}^1$$

- Training: A set of features and target values

$$(\boldsymbol{x}_i, t_i), i = 1, \ldots, N$$

- Objective: Predict target values form test features, given a large number of training data with target values (teachers)

Representation of Data

- Matrix representation of cases

By convention, a sample is represented by a row of matrices

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1D} \\ x_{21} & x_{22} & \cdots & x_{2D} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{ND} \end{pmatrix}$$

- Vector representation of target value

$$\boldsymbol{t} = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix}$$

Linear Regression Formulation

- Linear regression model
  - Single regression $t = w_0 + w_1 x$
  - Multiple regression $t = w_0 + w_1 x_1 + \ldots + w_D x_D = w_0 + \sum_{d=1}^{D} w_d x_d$
    - Predict target values from multiple features and a bias

- Define a case as

Add 1 to the top

$$\begin{pmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1D} \\ 1 & x_{21} & x_{22} & \cdots & x_{2D} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & x_{N2} & \cdots & x_{ND} \end{pmatrix}$$

$$t = \sum_{i=0}^{D} w_i x_i = \boldsymbol{w}^T \boldsymbol{x}$$

Convenient to write multiple regression model with one dot product

Linear Regression of Wine Data

Features x

Target value t

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5

- Single regression (use only one feature)
  - quality = w_0 + w_1 * fixed_acidity
- Multiple regression (use multiple features)
  - quality = w_0 + w_1 * fixed_acidity + w_2 * volatile_acidity +⋯ + w_{10} * alcohol

Minimize Squared Error

- Sum of squared errors $E(\boldsymbol{w}) = \sum_{i=1}^{N} (t_i - \boldsymbol{w}^T \boldsymbol{x}_i)^2$
- Model parameters that minimize the sum of squared errors

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \sum_{i=1}^{N} (t_i - \boldsymbol{w}^T \boldsymbol{x}_i)^2$$

  - Note that the result does not change without *1/N*
  - $\arg\min_x f(x)$: Argument x that minimizes the function $f(x)$
  - How to find w^*?

Exercise 3:
Convex function optimization

$f(\boldsymbol{x}) = 2x_1^2 + x_1x_2 + x_2^2 - 5x_1 - 3x_2 + 4$ $f(\boldsymbol{x})$ is convex

1. f の勾配 ∇f を求めよ
2. $(0,0), (1,2), (1,0.5), (1,1)$ における f の勾配を求めよ
3. f を最小にする $\boldsymbol{x}$ とその時の $f(\boldsymbol{x})$ を求めよ

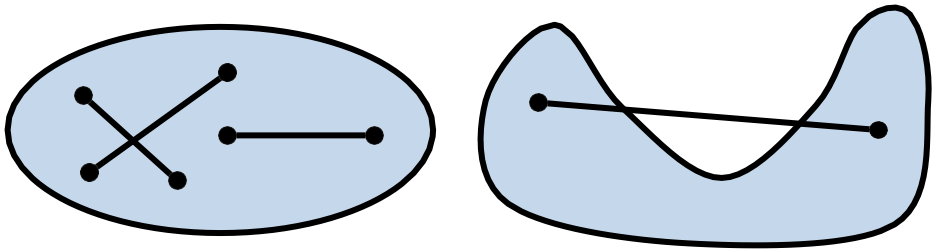
1. Find the gradient ∇f of f
2. Find the gradient of f at (0, 0), (1, 2), (1, 0.5), (1, 1)
3. Find $\boldsymbol{x}$ that minimizes f and $f(\boldsymbol{x})$ at that time

Definition of Optimization

$$\begin{aligned} \text{maximize } & f(x) && \text{Objective function} \\ \text{subject to } & g(x) = 0 && \text{Equality constraint} \\ & h(x) \leq 0 && \text{Inequality constraint} \end{aligned}$$

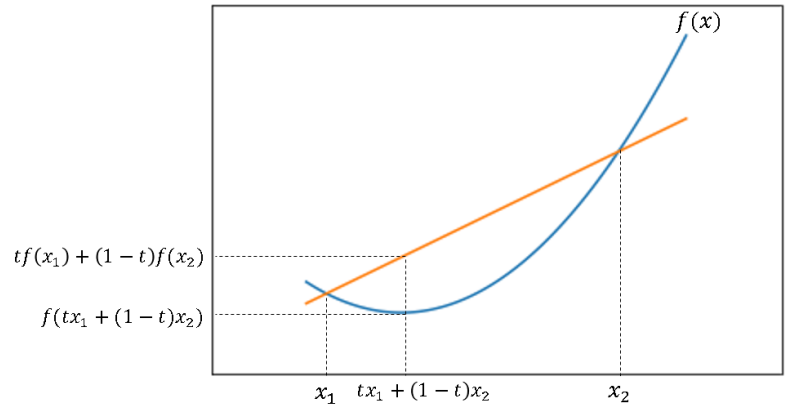
- Constraint-satisfying solution: feasible solution
- Area that meets constraints: feasible area

Convex Set and Convex Function



(a) convex set (b) nonconvex set

- Set A is a convex set ⇔ $\forall x, y \in A \text{ and } t \in [0,1], \quad tx + (1-t)y \in A$
- Downward convex function: Also called concave function



- Function f is convex downward

⇔ $\forall t \in [0,1], f(tx + (1-t)y) \leq tf(x) + (1-t)f(y)$

What Functions are Convex Functions?

- Linear function, affine function
- For definite matrix $\boldsymbol{Q}$ $\quad \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{c}^T\boldsymbol{x}$
- Norm
  - $exp(x), -\log(x), x\log(x)$
  - $(x_1 x_2 \ldots x_n)^{1/n} \quad x_i \geq 0$
  - $\log(e^{x_1} + e^{x_2} + \ldots + e^{x_n})$

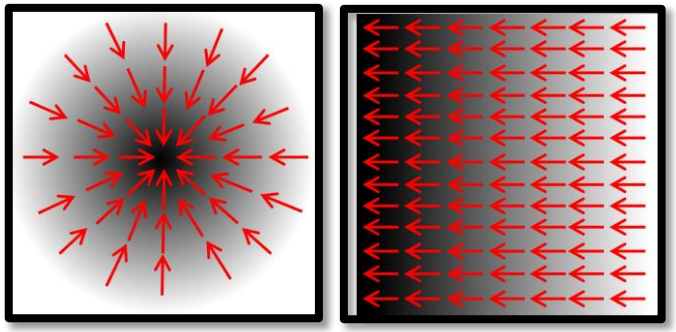
Differentiation of Multivariable Functions

- Function $f : \mathbb{R}^D \rightarrow \mathbb{R}$
- Derivative at a variable (x_d)
- Derivatives (gradients) for all variables

$$f(\boldsymbol{x}) = a$$

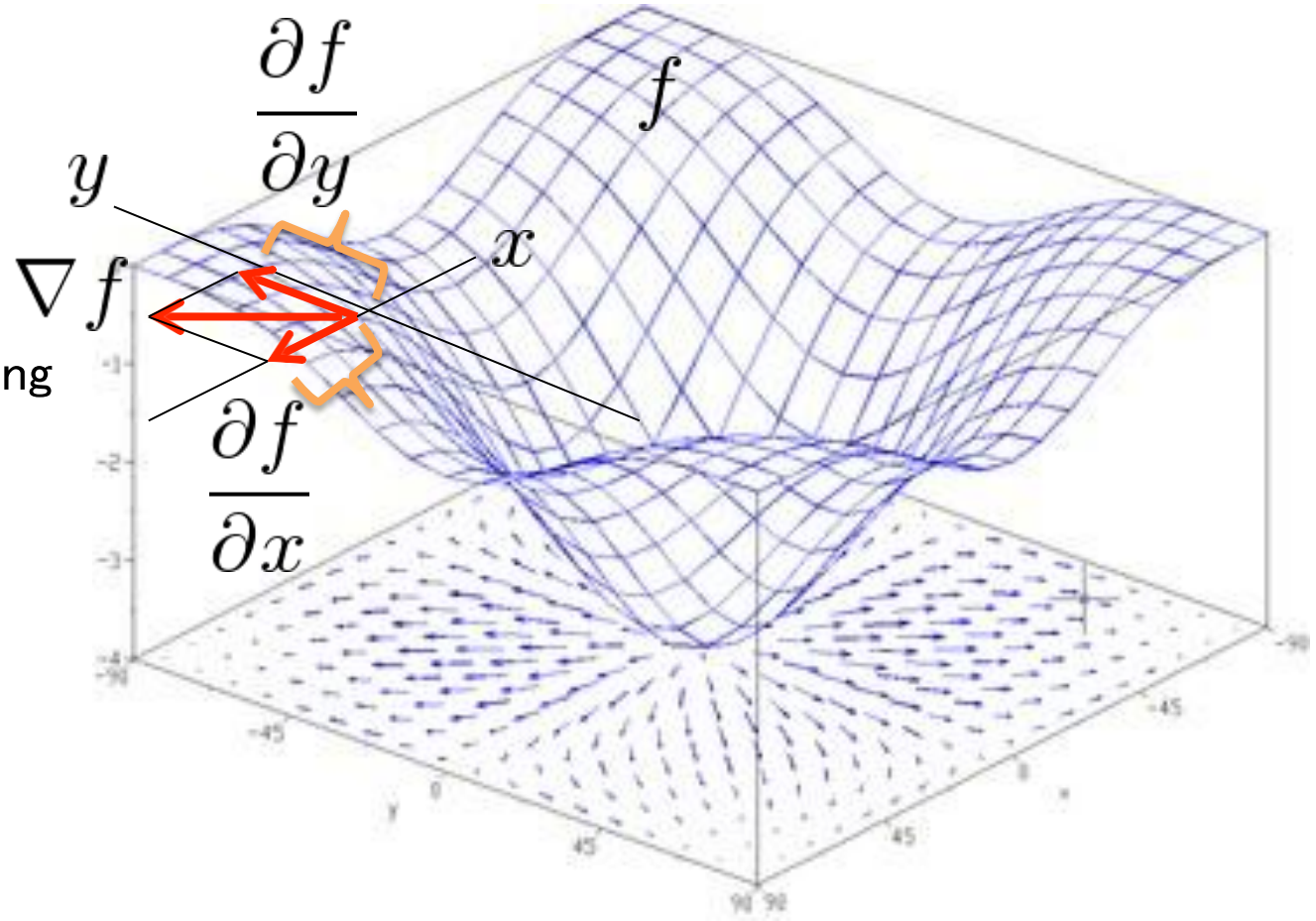
$$\frac{\partial f}{\partial x_d}$$

$$\frac{\partial f}{\partial \boldsymbol{x}} = \nabla_{\boldsymbol{x}} f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_d} \end{pmatrix}$$



Gradient $\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_d} \right)^T$

$f(x,y) = -(\cos^2 x + \cos^2 y)^2$ Projection of the gradient onto the bottom



Steepest climbing direction

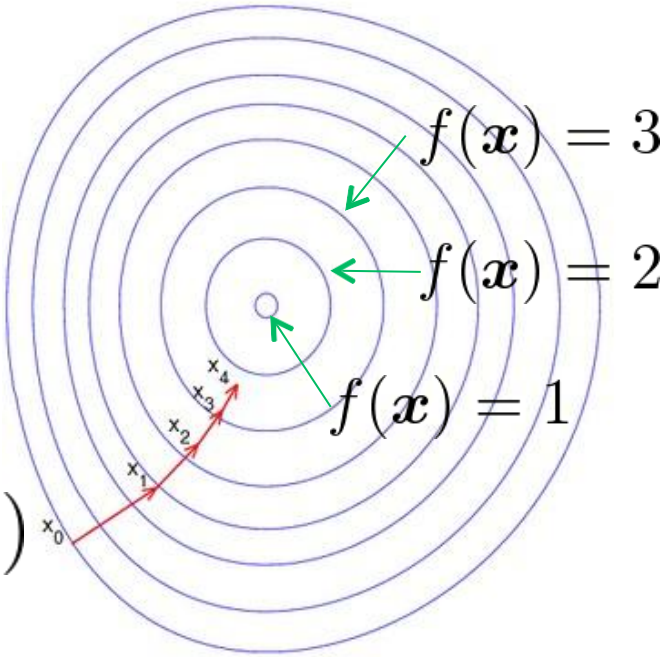
Reference

Contour Lines and Gradients

- Contour lines: a set of $\boldsymbol{x}$ giving the same value of $f(\boldsymbol{x})$
  - Become a closed curve
- Gradient: direction in which $f(\boldsymbol{x})$ increases most at point $\boldsymbol{x}$
  - Gradient of the D-dimensional function is a D-dimensional vector

Gradient $\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_d} \right)^T$

Gradient size = steepness of slope $\left\| \nabla_{\boldsymbol{x}} f(\boldsymbol{x}) \right\|$



$-\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_0)$

Because the value is decreasing, the slope is the opposite direction of the gradient

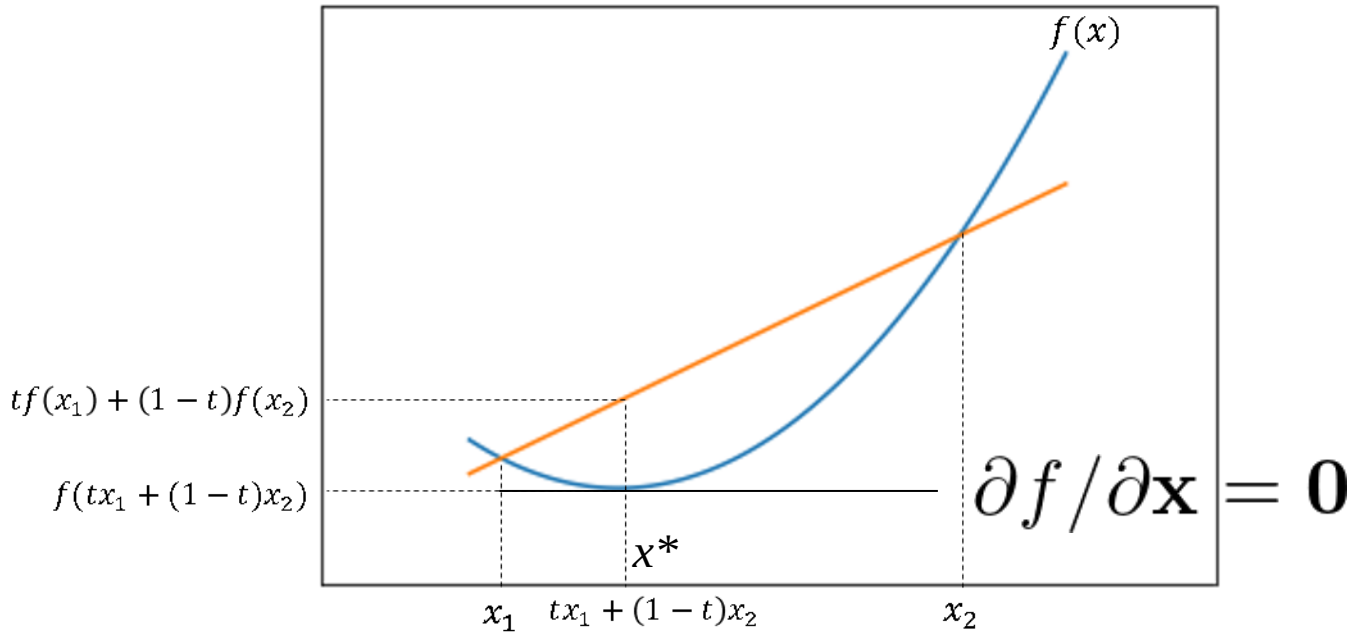
Differentiable Convex Function Optimization

- Analytical solution
  - Analytically find x for which the gradient vector is a 0 vector
  - If an analytical solution is available, an exact solution can be found, so use this if possible

- Approximate solution
  - Gradient descent (and its families)
  - Randomly initializes and repeatedly updates the solution in the gradient direction
  - Approximate solution can be found (accuracy improves according to the time used)
  - Approximate solutions may be used even if analytical solutions are available (for computational reasons)

Analytical Solution
Minimization of differentiable convex functions

- Function $f : \mathbb{R}^D \rightarrow \mathbb{R}$ should be differentiable and convex
- f has a minimum value, where x meets $\partial f / \partial \mathbf{x} = \mathbf{0}$
- If we denote such x by x^*, then the minimum value of f is given by $f(x^*)$



Gradient Descent Method (Approximate solution)

Problem: find w which satisfies $\min_{\boldsymbol{w}} f(\boldsymbol{w})$

– Function f can be differentiable

Algorithm

1. $t = 0, \boldsymbol{w}^0$ randomly initialize
2. $\boldsymbol{w}^{t+1} \leftarrow \boldsymbol{w}^t - \eta \nabla_{\boldsymbol{w}} f$
3. $\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\| < \epsilon$ then stop, otherwise go to step 4
4. $t \leftarrow t + 1$ go to step 2

Step size parameter η : How much to update in one step

Convergence determination parameter ϵ

Convex Planning Problem

- Convex planning problem
  - Objective function is a convex function
  - Feasible region is in a convex set

Convex planning problems	Unconstrained	Equality constraint	Inequality constraint
f is differentiable	The point where the derivative of f is 0 can be obtained analytically ⇒ That point is the optimal solution Not obtained ⇒ Gradient descent method, etc.	Lagrange's undetermined multiplier method	Lagrange's undetermined multiplier method Not dealt with this time (appears in SVM derivation etc.)
f is not differentiable	I will touch it in a few weeks	Not dealt with this time	Not dealt with this time

Minimization of Sum of Squared Errors

- Sum of squared errors $E(\boldsymbol{w}) = \sum_{i=1}^{N} (t_i - \boldsymbol{w}^T \boldsymbol{x}_i)^2$
- Minimize squared error
  - $E(\boldsymbol{w})$ is a downward convex function with respect to $\boldsymbol{w}$
    - Assumed that it is convex downward *
  - Find $\boldsymbol{w}$ which satisfies $\frac{\partial E}{\partial \boldsymbol{w}} = 0$
  - $\boldsymbol{w} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{t}$ (Derivation is an exercise)

*All the eigenvalues of the Hessian matrix of E should be positive

Summary of Regression so far

- Predict x to t in linear form $t = \sum_{i=0}^{D} w_i x_i = \boldsymbol{w}^T \boldsymbol{x}$
- Performance of prediction by the regression coefficient w is evaluated by the squared error

$$E(\boldsymbol{w}) = \sum_{i=1}^{N} (t_i - \boldsymbol{w}^T \boldsymbol{x}_i)^2$$

- Since the squared error is convex with respect to w, w that minimizes the squared error is where $E(w)$ is differentiated by w and becomes zero

$$\frac{\partial E}{\partial \boldsymbol{w}} = 0 \Rightarrow \boldsymbol{w} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{t}$$

Exercise 4:
Derivation of linear regression

$$\boldsymbol{x}_i = \begin{pmatrix} 1 \\ x_{i1} \\ \vdots \\ x_{iD} \end{pmatrix}, \boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{pmatrix}, \boldsymbol{t} = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix}, \boldsymbol{w} = \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_D \end{pmatrix}$$

Find a linear regression model $\boldsymbol{t} = \boldsymbol{w}^T\boldsymbol{x}$ using the training samples

1. Express the sum of squared errors E as a function of $\boldsymbol{w}$
2. Derive the following (a)(b) to find the gradient $\nabla_w E$ for $\boldsymbol{w}$ of the sum of squared errors E

   (a) $\sum_{i=1}^{N} t_i \boldsymbol{x}_i = \boldsymbol{X}^T \boldsymbol{t}$

   (b) $\sum_{i=1}^{N} \boldsymbol{x}_i \boldsymbol{w}^T \boldsymbol{x}_i = \boldsymbol{X}^T \boldsymbol{X} \boldsymbol{w}$
3. Show the gradient $\frac{\partial E}{\partial \boldsymbol{w}}$ in terms of $\boldsymbol{x_i}$ (or $\boldsymbol{X}$), $\boldsymbol{t}$
4. (Approximate solution) Show the parameter update equation for the linear regression model using the gradient descent method in terms of $\boldsymbol{x_i}$ (or $\boldsymbol{X}$), $\boldsymbol{t}$
   Initial solution $\boldsymbol{w^0}$, t−th update $\boldsymbol{w^t}$, step size parameter η
5. (Analytic solution) Show that $\boldsymbol{w}$ where $\frac{\partial E}{\partial \boldsymbol{w}} = 0$ is

$$\boldsymbol{w} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{t}$$

Exercise 4: Derivation of linear regression

$$\boldsymbol{x}_i = \begin{pmatrix} 1 \\ x_{i1} \\ \vdots \\ x_{iD} \end{pmatrix}, \boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{pmatrix}, \boldsymbol{t} = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix}, \boldsymbol{w} = \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_D \end{pmatrix}$$ とする。事例群 $\{(\boldsymbol{x}_i, t_i)\}_{i=1}^N$ を使って線形回帰モデル $t = \boldsymbol{w}^T \boldsymbol{x}$ を求めることを考える。

1. 二乗誤差和 E を $\boldsymbol{w}$ の関数で表せ
2. 二乗誤差和 E の $\boldsymbol{w}$ についての勾配 $\nabla_{\boldsymbol{w}} E$ を求めるために、以下を導出せよ
   (a) $\sum_{i=1}^N t_i \boldsymbol{x}_i = \boldsymbol{X}^T \boldsymbol{t}$
   (b) $\sum_{i=1}^N \boldsymbol{x}_i \boldsymbol{w}^T \boldsymbol{x}_i = \boldsymbol{X}^T \boldsymbol{X} \boldsymbol{w}$
3. 勾配 $\frac{\partial E}{\partial \boldsymbol{w}}$ を $\boldsymbol{x}_i$(あるいは $\boldsymbol{X}$), $\boldsymbol{t}$ の式で示せ。
4. (近似解法) 線形回帰モデルを最急降下法で求めるときの、パラメータの更新式を $\boldsymbol{x}_i$(あるいは $\boldsymbol{X}$), $\boldsymbol{t}$ の式で示せ。初期解を $\boldsymbol{w}^0$, t 回目の更新時の回を $\boldsymbol{w}^t$, ステップサイズパラメータを η とする。
5. (解析解) $\frac{\partial E}{\partial \boldsymbol{w}} = 0$ なる $\boldsymbol{w}$ が $\boldsymbol{w} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{t}$ であることを示せ。

Machine Learning (4)(5)

Multiple regression

Machine Learning

INFQ612L, 440113450A

Spring Semester

Friday 17:00-18:40

IPS

WASEDA University

Prof. Shoji Makino