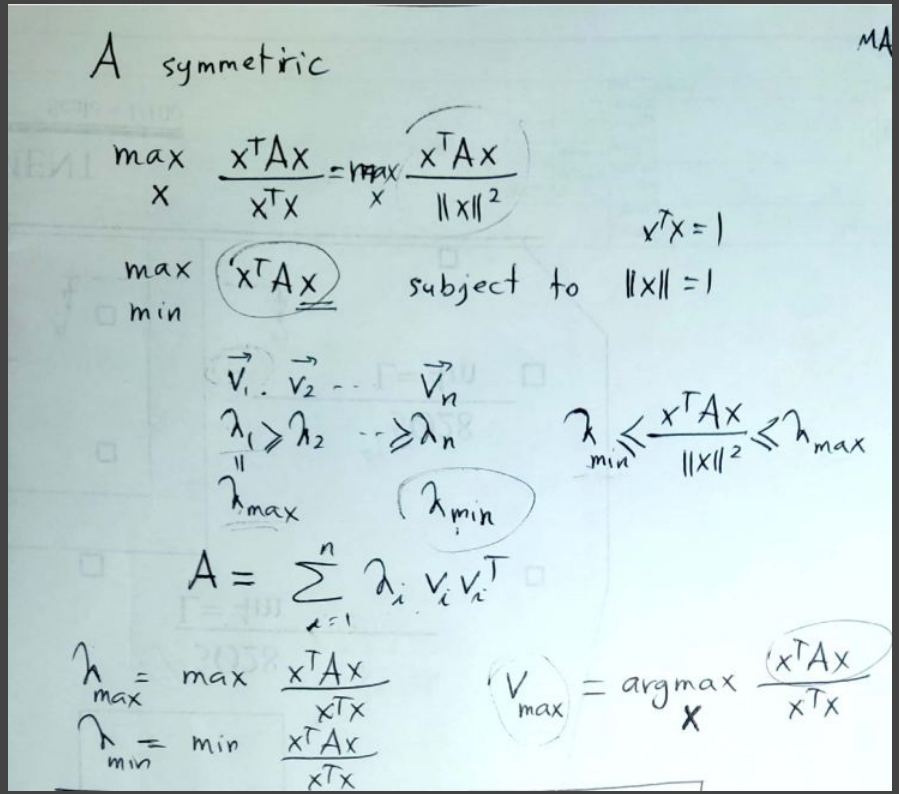
Mathematics for AI

Lecture 14
Noisy Homogeneous Equations, Affine spaces, affine maps, nonlinear functions, linearization and derivatives

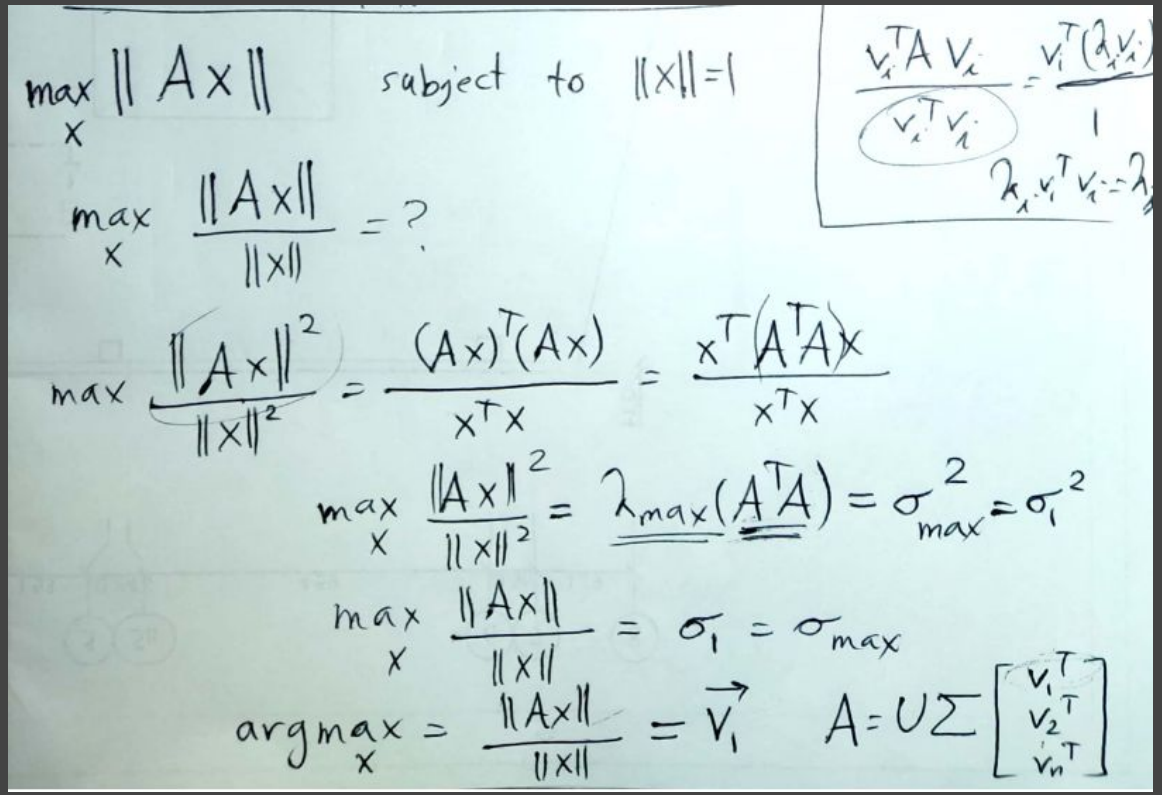
Eigen-decomposition and optimization

K. N. Toosi
University of Technology



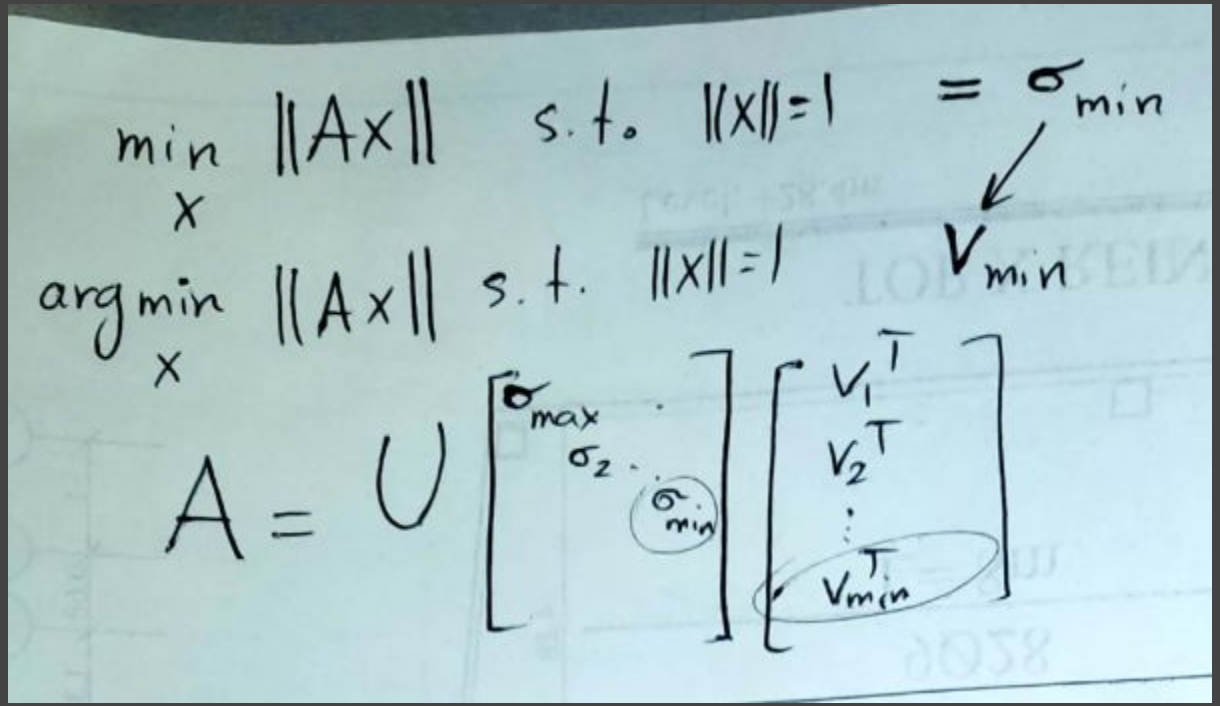
SVD and optimiztion

K. N. Toosi
University of Technology



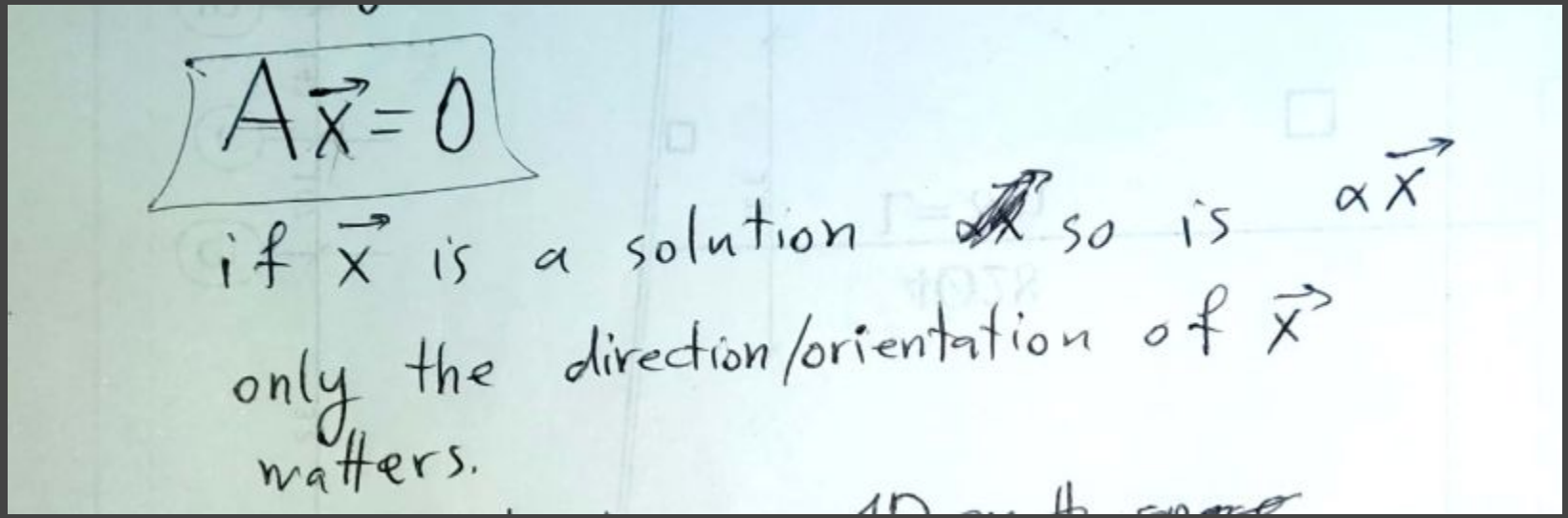
SVD and optimiztion

K. N. Toosi
University of Technology



Remember: homogeneous equations

K. N. Toosi
University of Technology



Remember: homogeneous equations

K. N. Toosi
University of Technology

$$A\vec{x} = 0$$

In many applications A has a 1D null space

$A \in \mathbb{R}^{m \times n}$ $\Rightarrow$ A has $n-1$ independent columns

$\text{rank}(A) = n-1$

$X = \text{null space}(A)$

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = 0 \qquad a_1^T x = 0$$
$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = 0 \qquad a_2^T x = 0$$
$$\vdots$$
$$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = 0 \qquad a_m^T x = 0$$

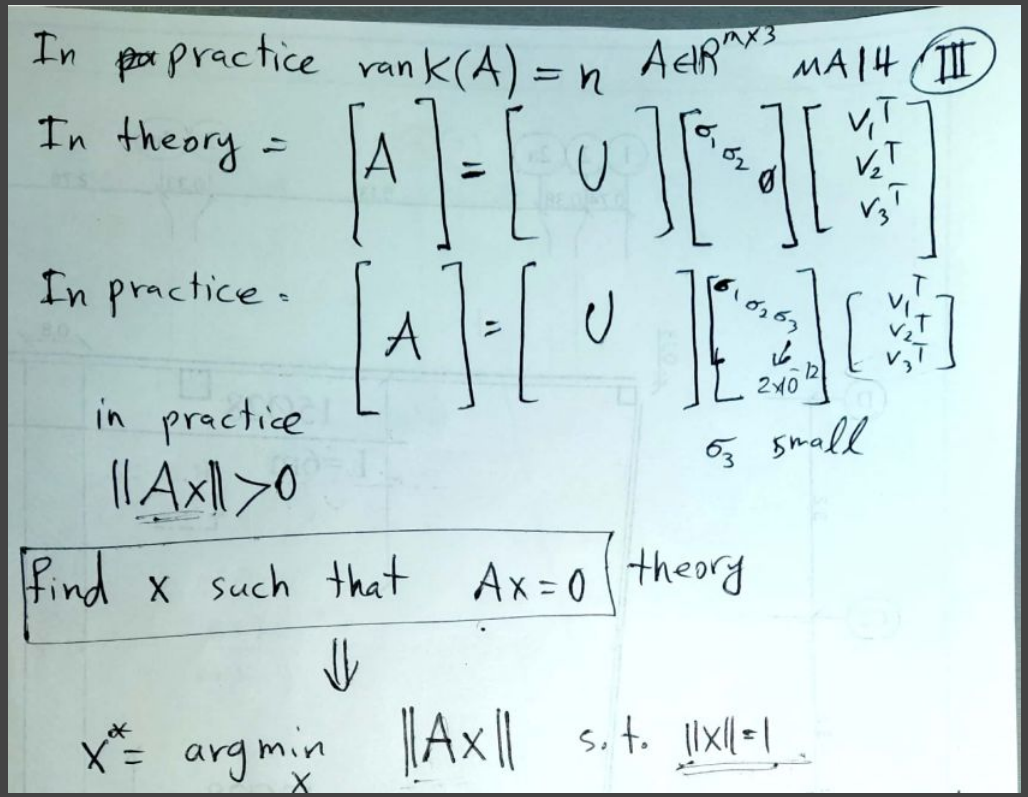
$$\begin{bmatrix} a_1^T \\ a_2^T \\ \vdots \\ a_m^T \end{bmatrix} \vec{x} = \vec{0}$$

$A\vec{x} = 0$

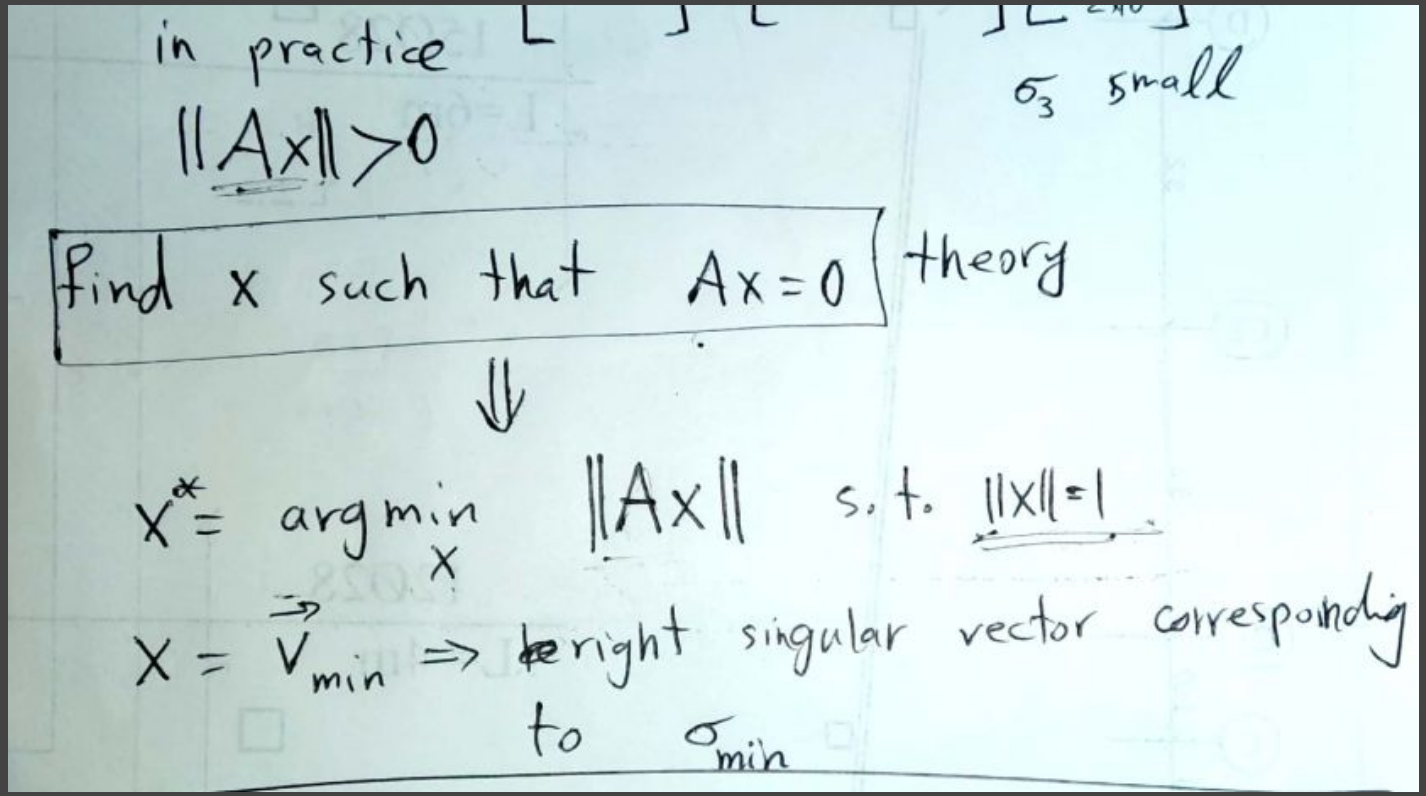
A is of rank $n-1$ $\quad m \gg n$

Due to noise/computation errors

Noisy homogeneous equations

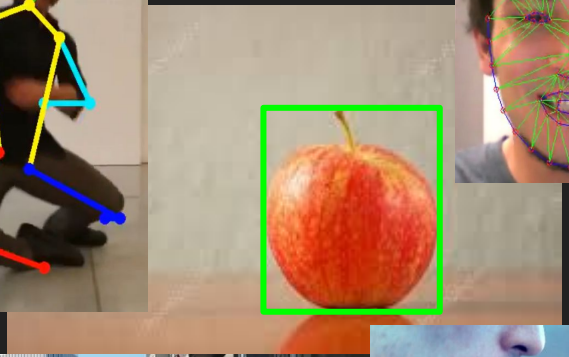


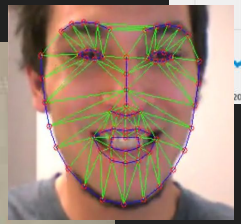
Noisy homogeneous equations

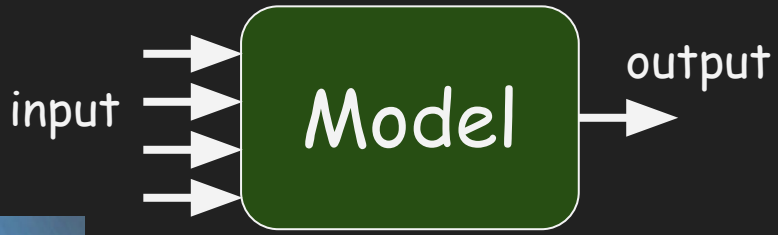


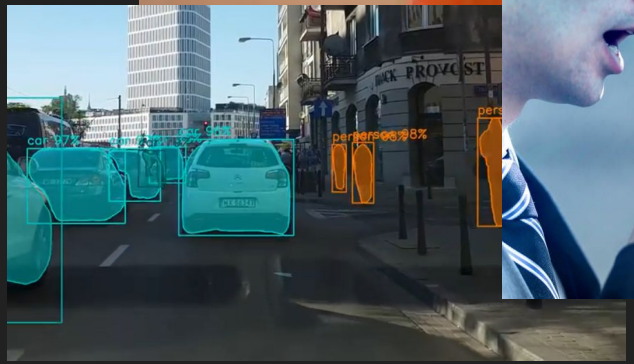
Machine Learning

K. N. Toosi
University of Technology









Classification

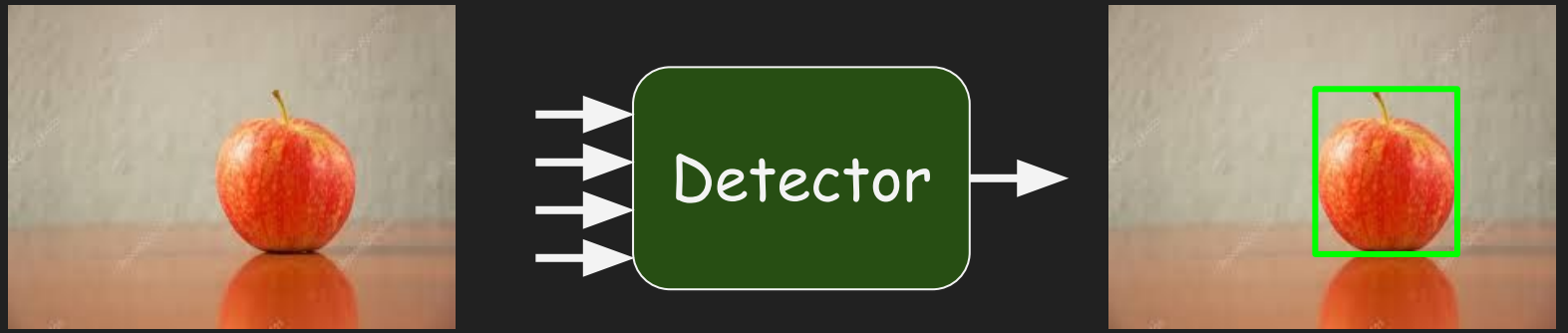
K. N. Toosi
University of Technology

Classification

K. N. Toosi
University of Technology

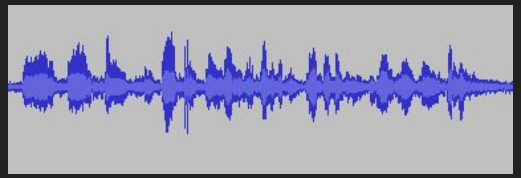
Object detection

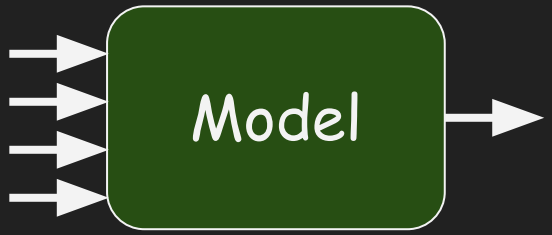
K. N. Toosi
University of Technology



Speech Recognition

K. N. Toosi
University of Technology

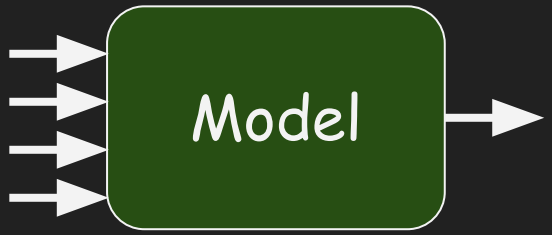


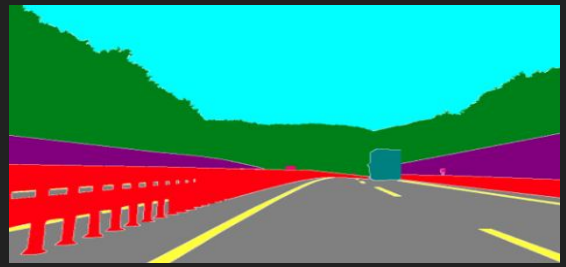


یکی بود یکی نبود.

Segmentation

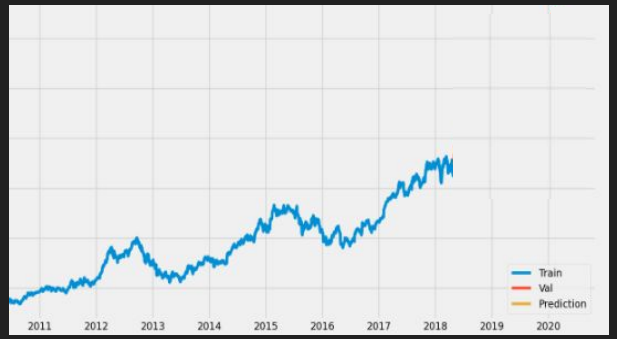
K. N. Toosi
University of Technology





Stock Market Prediction

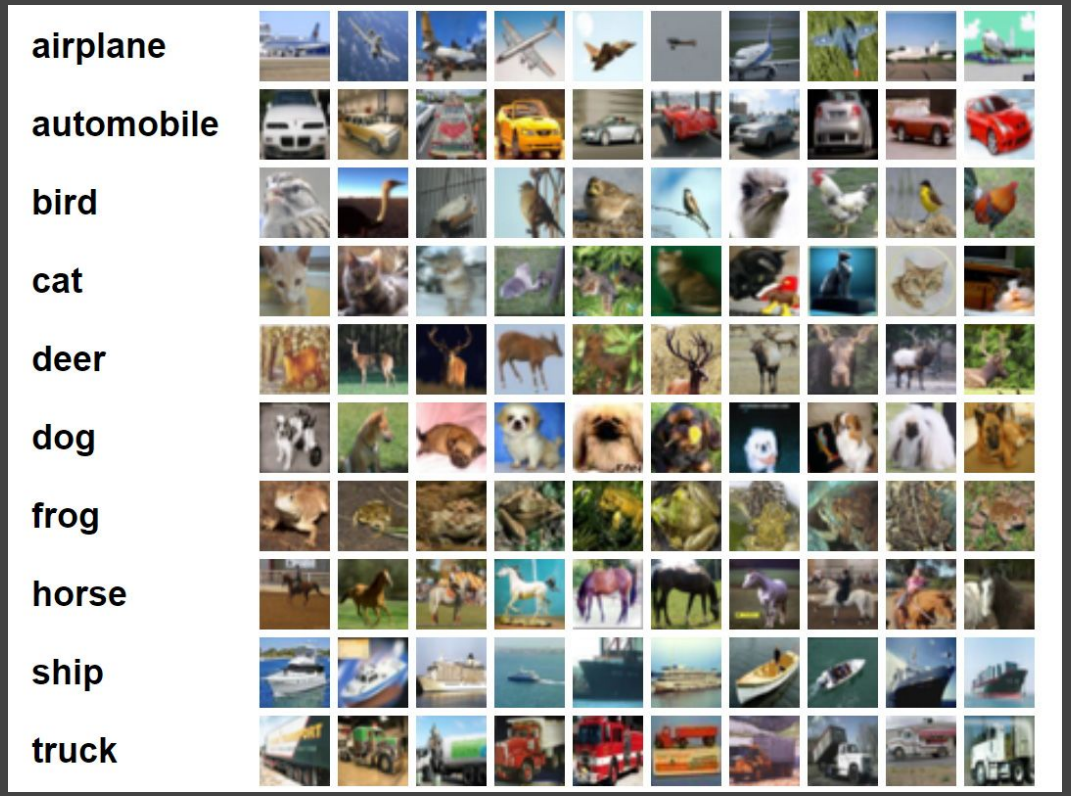
K. N. Toosi
University of Technology



Learning from data

https://www.analyticsvidhya.com/blog/2018/03/comprehensive-collection-deep-learning-datasets/

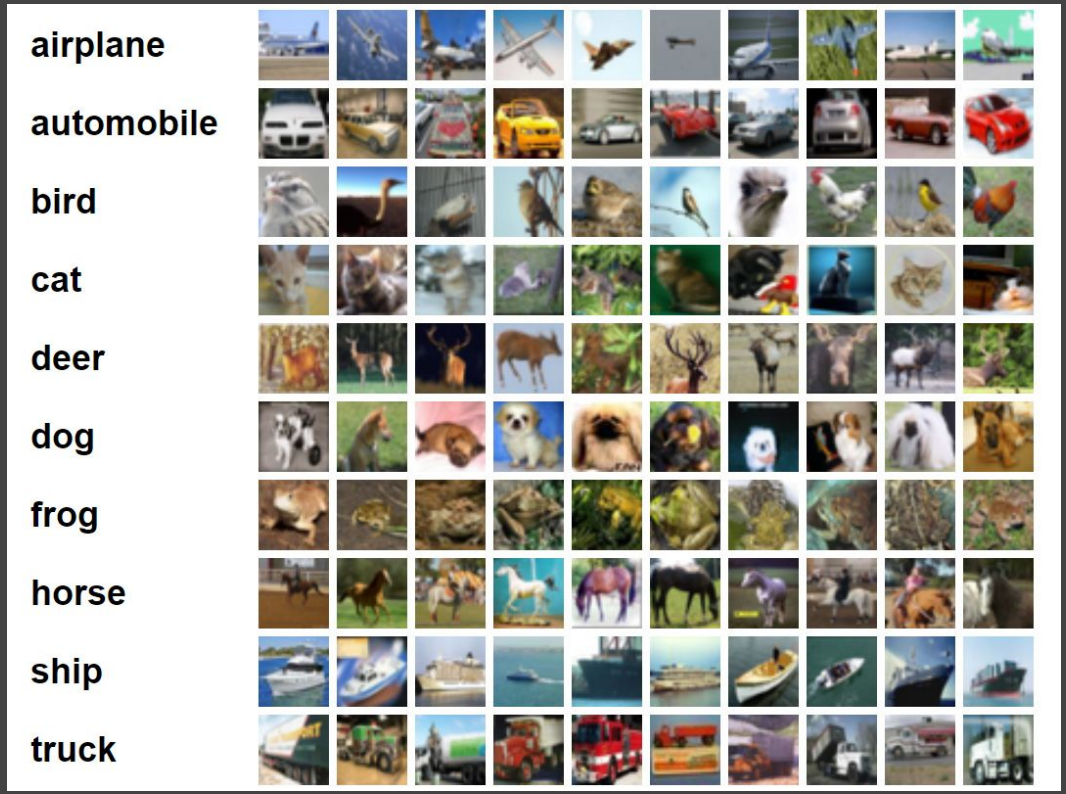
Supervised Learning



http://seansoleyman.com/effect-of-dataset-size-on-image-classification-accuracy/

K. N. Toosi
University of Technology

Supervised Learning



Training data:

X_1, y_1

X_2, y_2

X_3, y_3

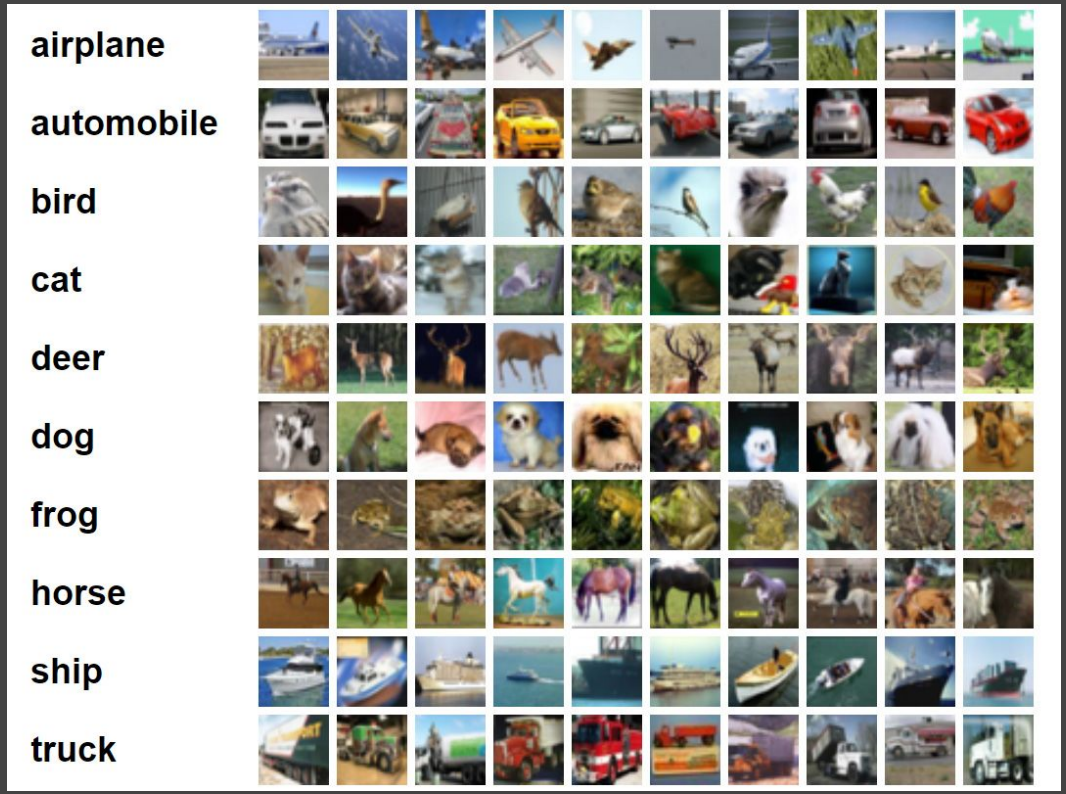
.

X_n, y_n

http://seansoleyman.com/effect-of-dataset-size-on-image-classification-accuracy/

K. N. Toosi
University of Technology

Supervised Learning



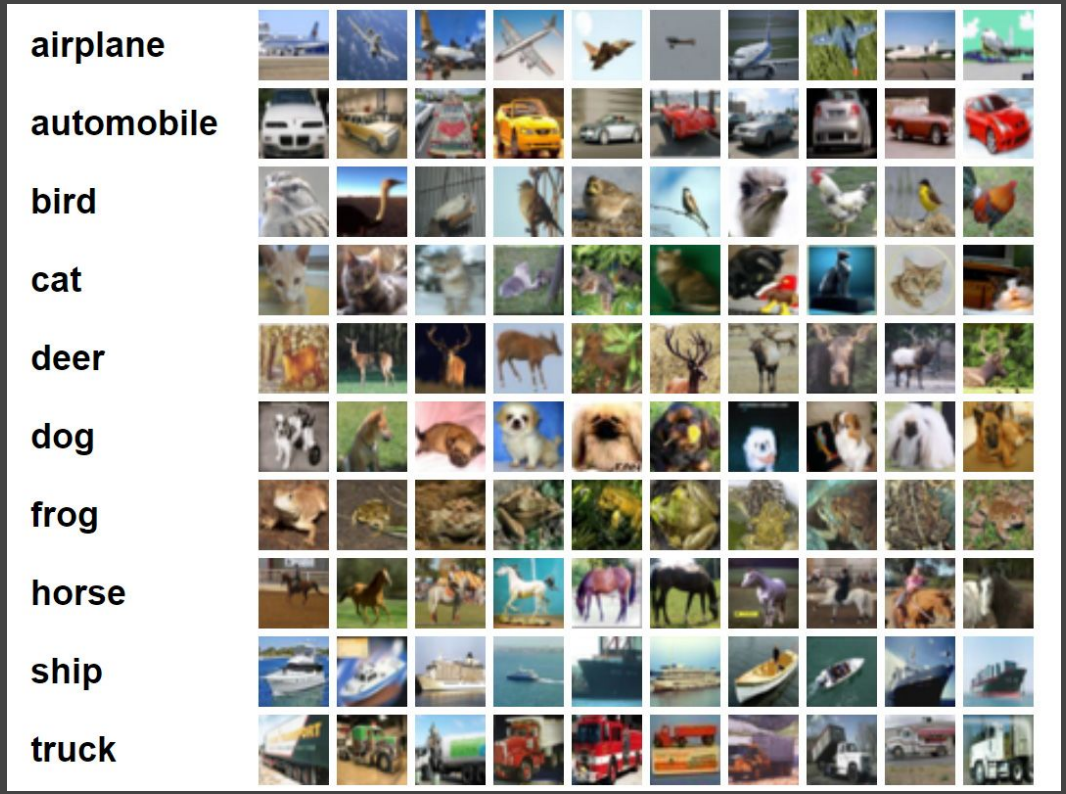
http://seansoleyman.com/effect-of-dataset-size-on-image-classification-accuracy/

K. N. Toosi
University of Technology

Training data:

	Apple
	Apple
	Orange
⋮	
	Orange

Supervised Learning



http://seansoleyman.com/effect-of-dataset-size-on-image-classification-accuracy/

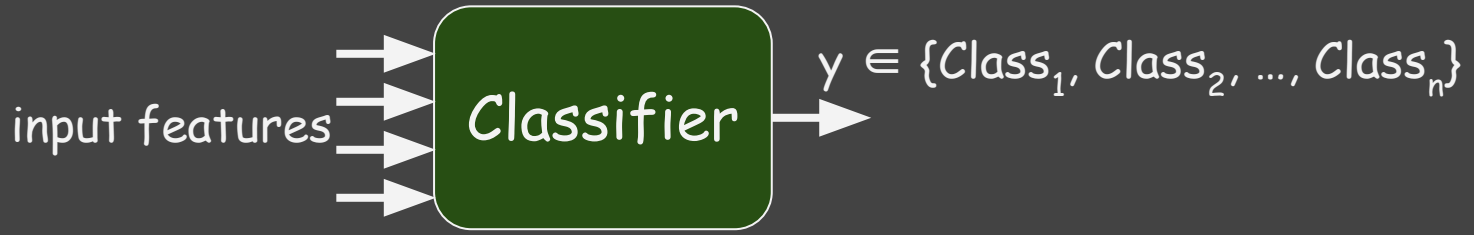
K. N. Toosi
University of Technology

Training data:

	0
	0
	1
⋮	
	1

Supervised Learning

Classification



Classification

K. N. Toosi
University of Technology

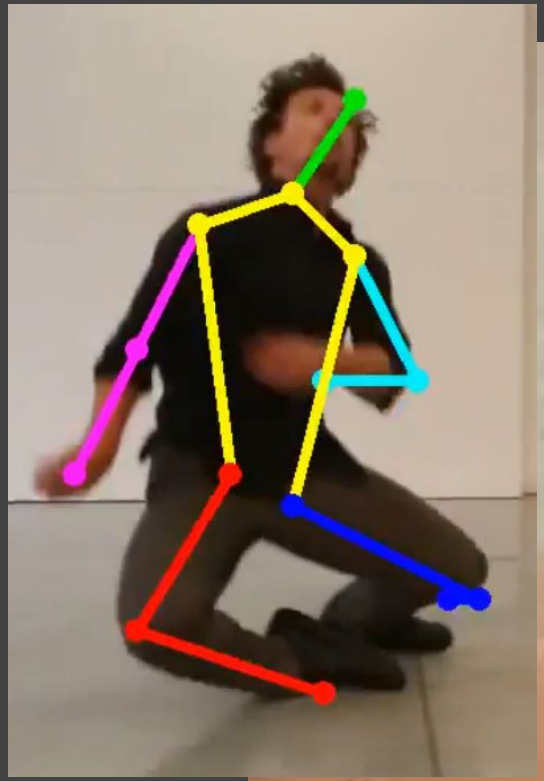
Classification

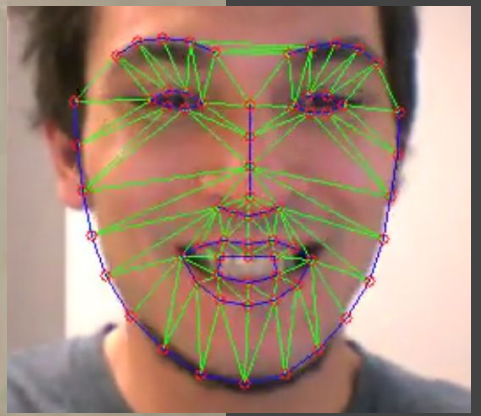
K. N. Toosi
University of Technology

Regression

Regression

Regression





K. N. Toosi
University of Technology

Learnable Models

Learnable Models: Example

K. N. Toosi
University of Technology

Learnable Models: Example

K. N. Toosi
University of Technology

Learnable Models: Input-output map

$$y = f(x)$$

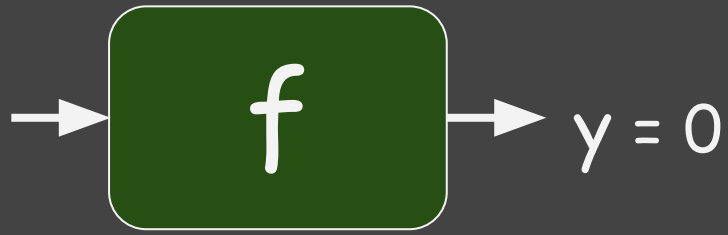
$$f: R^m \rightarrow R^n$$

Learnable Models: Example

K. N. Toosi
University of Technology

I

x = I.flatten()



y = 0

$$y = f(x)$$

$$f: R^m \rightarrow R^n$$

Learnable Models: Example

K. N. Toosi
University of Technology

I

x = features(I)

y = 0

$$y = f(x)$$

$$f: R^m \rightarrow R^n$$

Learnable Models: Example

$$y = f(x)$$

$$f: R^m \rightarrow R^n$$

Learnable Models: Example

K. N. Toosi
University of Technology

What about a linear function $y = f(x) = A\,x$?

$f: R^m \rightarrow R^n$

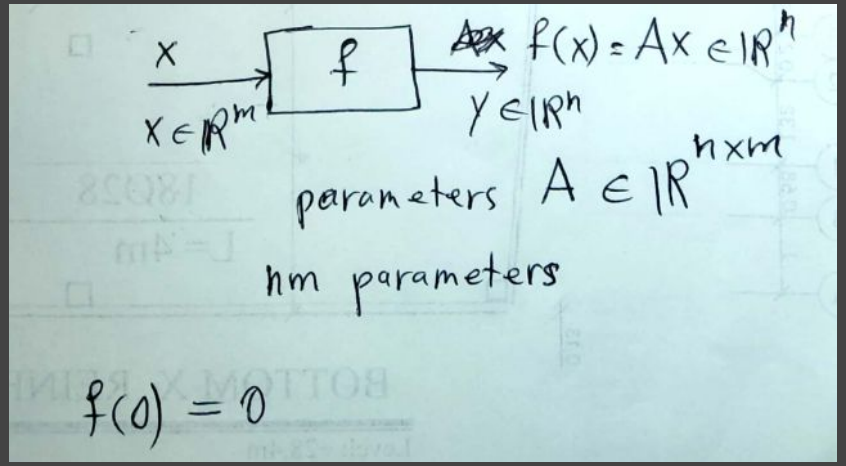
Learnable Models: Example

What about a linar function $y = f(x) = A\,x$?

$f: R^m \rightarrow R^n$

Learnable Models: Example

K. N. Toosi
University of Technology



What about a linar function $y = f(x) = A\,x$?

$f: R^m \rightarrow R^n$

Learnable Models: Example

K. N. Toosi
University of Technology

$y = f(x),\ f: R^m \rightarrow R^n$

How to make f learnable?

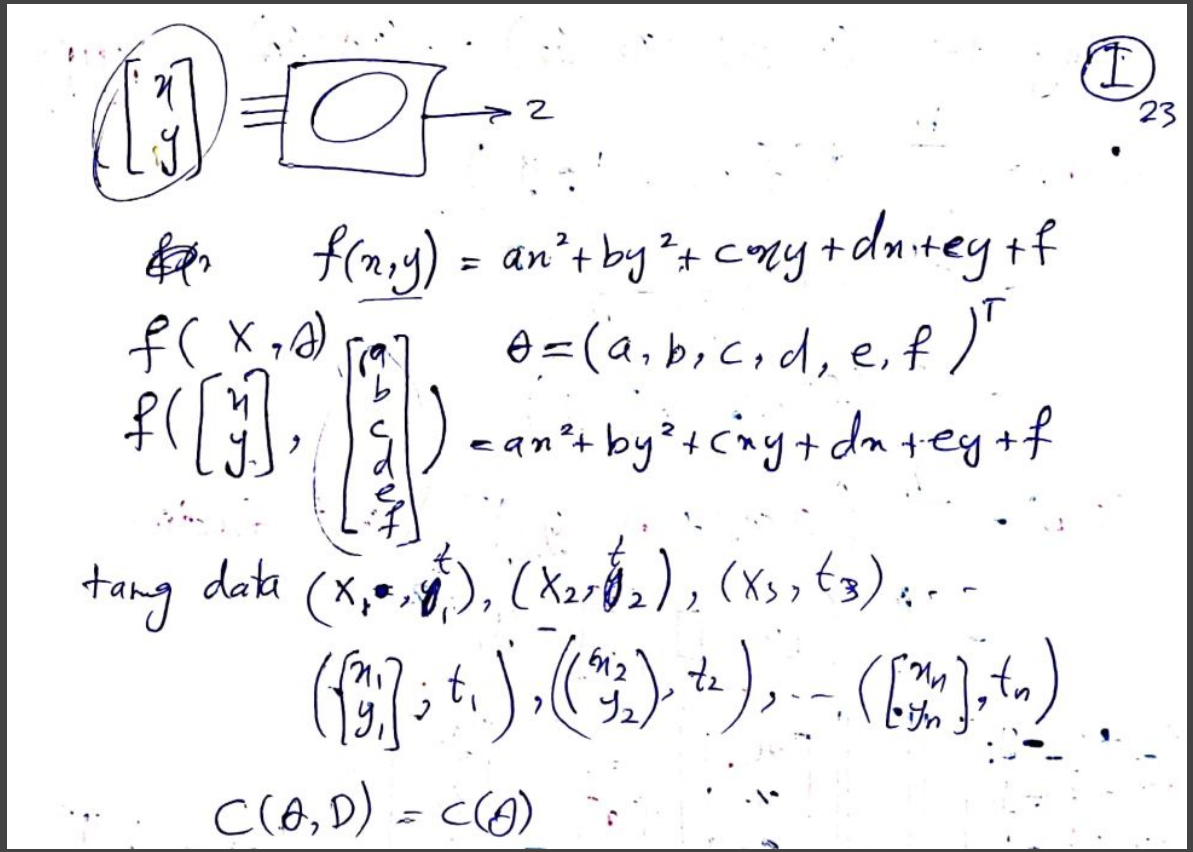
Learnable Models: parameters

K. N. Toosi
University of Technology

$$y = f(\theta, x)$$

θ: model parameters

Learnable Models: parameters



Learnable Models: parameters

K. N. Toosi
University of Technology

- Parameter Learning:
  - A collection of input-output paris $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$,
  - choose θ such that $y = f(\theta, x)$ is a reasonable output for any input x.

Learning from data

K. N. Toosi
University of Technology

- Parameter Learning:
  - A collection of input-output paris $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$,
  - choose θ such that $y = f(\theta, x)$ is a reasonable output
    - for training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
    - for unseen data (generalization)

Learning from data

K. N. Toosi
University of Technology

- Parameter Learning:
  - A collection of input-output paris $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$,
  - choose θ such that $y = f(\theta, x)$ is a reasonable output
    - for training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
    - for unseen data (generalization)

Learning from data

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i

Learning from data: Cost function

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i

Cost function

Learning from data: Cost function

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i
  - cost function:

$$C(\theta) = \Sigma_{i=1..N}\, d(\, f(\theta, x_i), y_i\,)$$

Learning from data: Cost function

K. N. Toosi
University of Technology

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i
  - cost function:

$$C(\theta) = \Sigma_{i=1..n}\, d(\, f(\theta, x_i), y_i \,)$$

data output

Learning from data: Cost function

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i
  - cost function:

$$C(\theta) = \Sigma_{i=1..n}\, d(\, f(\theta, x_i), y_i \,)$$

model output given x_i

Learning from data: Cost function

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i
  - cost function:

$$C(\theta) = \Sigma_{i=1..n}\, d(\, f(\theta, x_i), y_i \,)$$

distance

Learning from data: Cost function

K. N. Toosi
University of Technology

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i
  - cost function:

$$C(\theta) = \Sigma_{i=1..n} \| f(\theta, x_i) - y_i \|^2$$

distance

Learning from data: Cost function

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i
  - cost function:

$$C(\theta) = \Sigma_{i=1..n}\, d(\, f(\theta, x_i), y_i\,)$$

Learning from data: Cost function

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i
  - cost function:

$$C(\theta) = \Sigma_{i=1..n}\, d(\, f(\theta, x_i), y_i\,)$$

choose θ such that $C(\theta)$ is small

Learning from data: Cost function

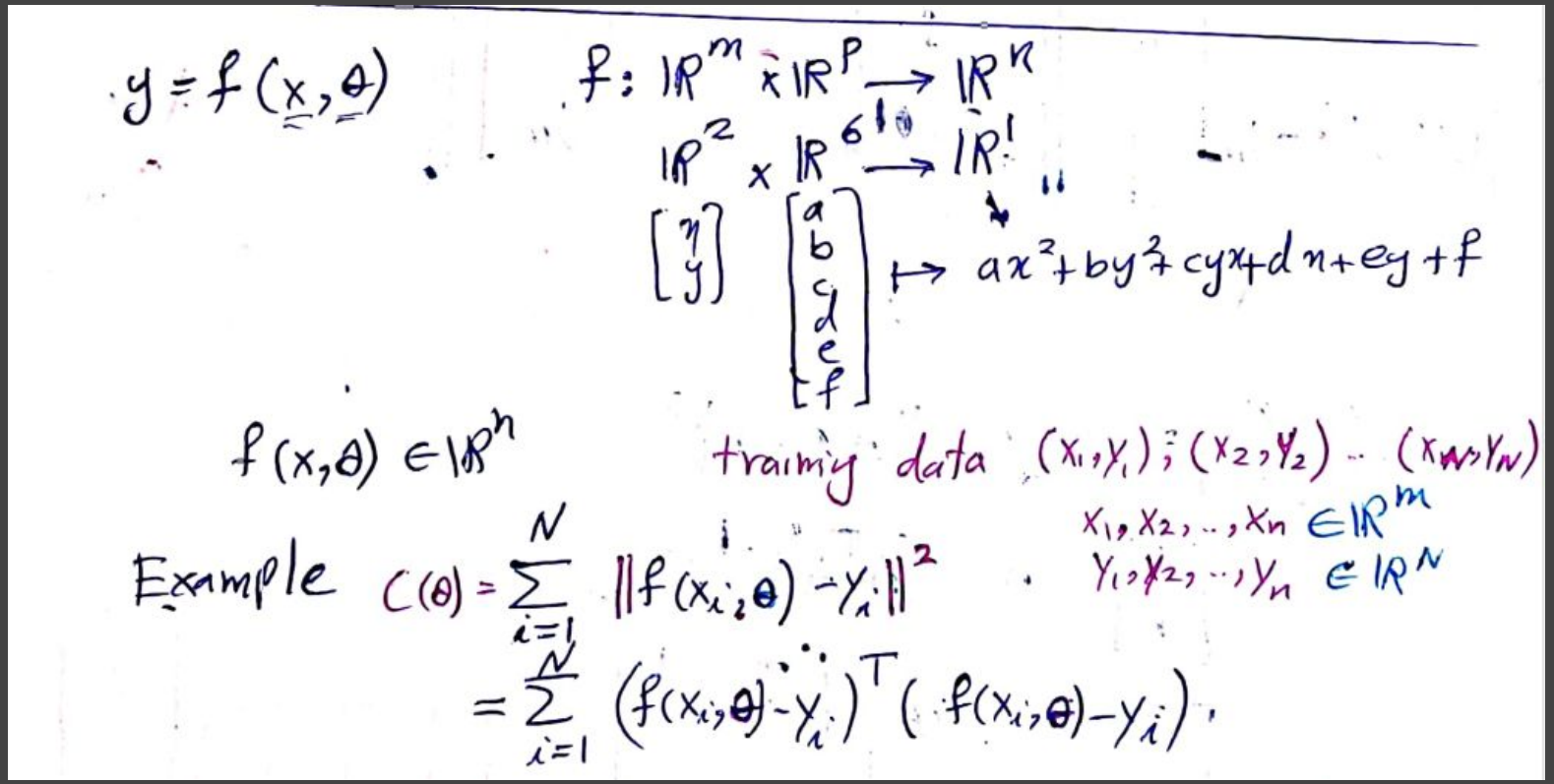
K. N. Toosi
University of Technology

- Training data $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$
  - choose θ such that $f(\theta, x_i)$ is close to y_i
  - cost function:

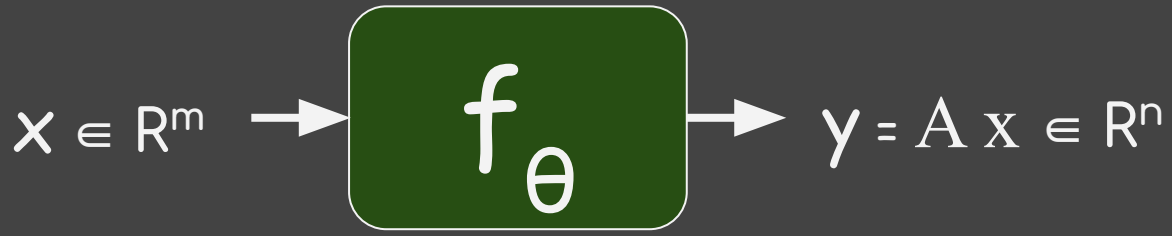
$$C(\theta) = \Sigma_{i=1..n}\, d(\, f(\theta, x_i), y_i\,)$$

$$\theta^* = \text{argmin}_\theta\, C(\theta)$$

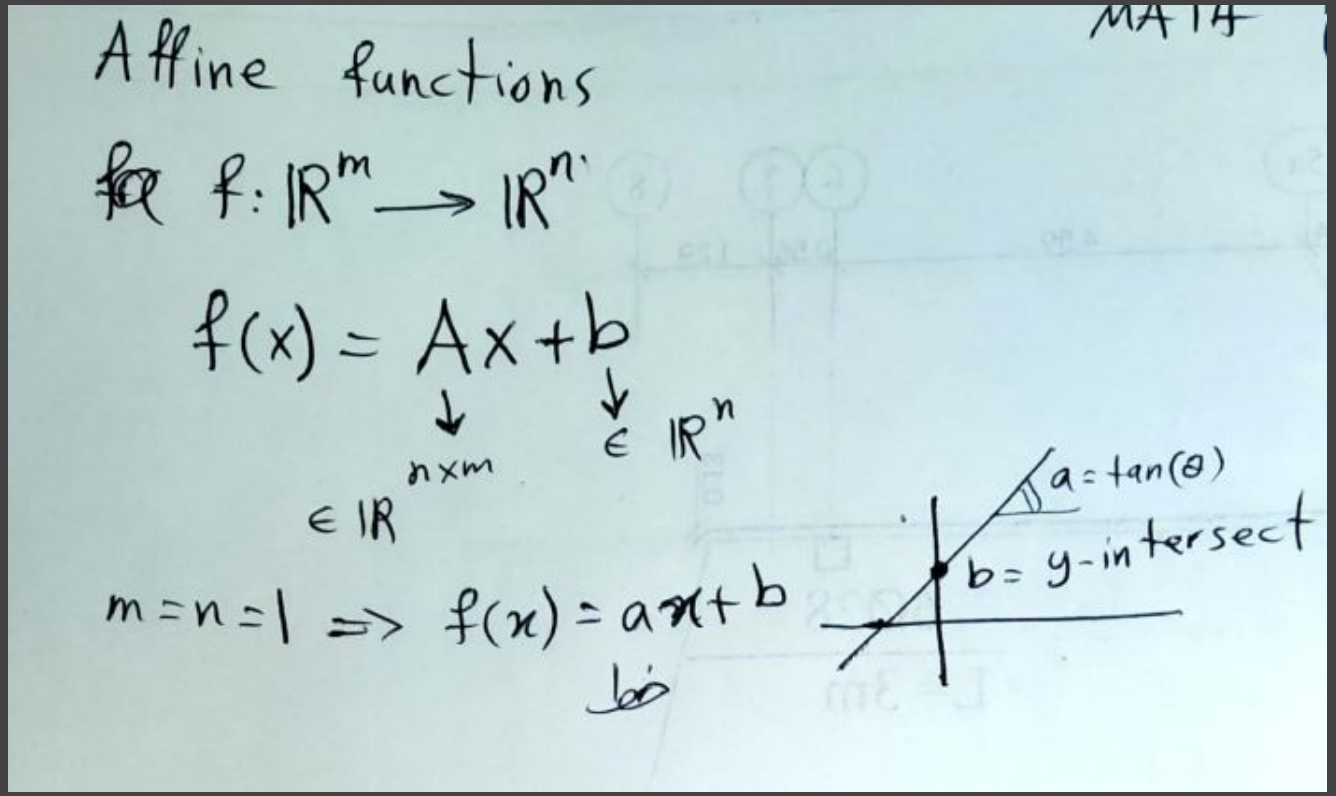
Cost function



Example: Linear fuctions



Affine functions

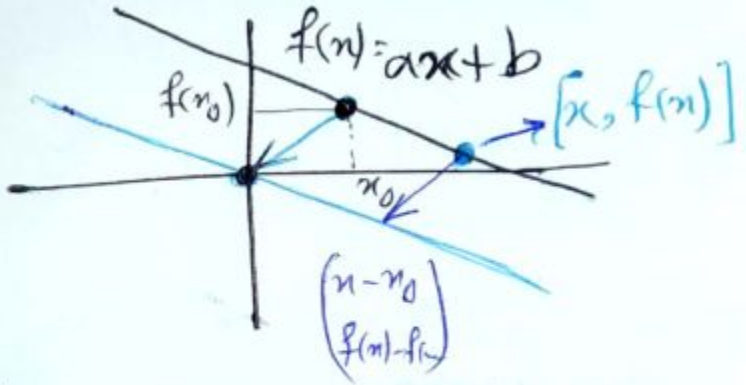


Affine functions

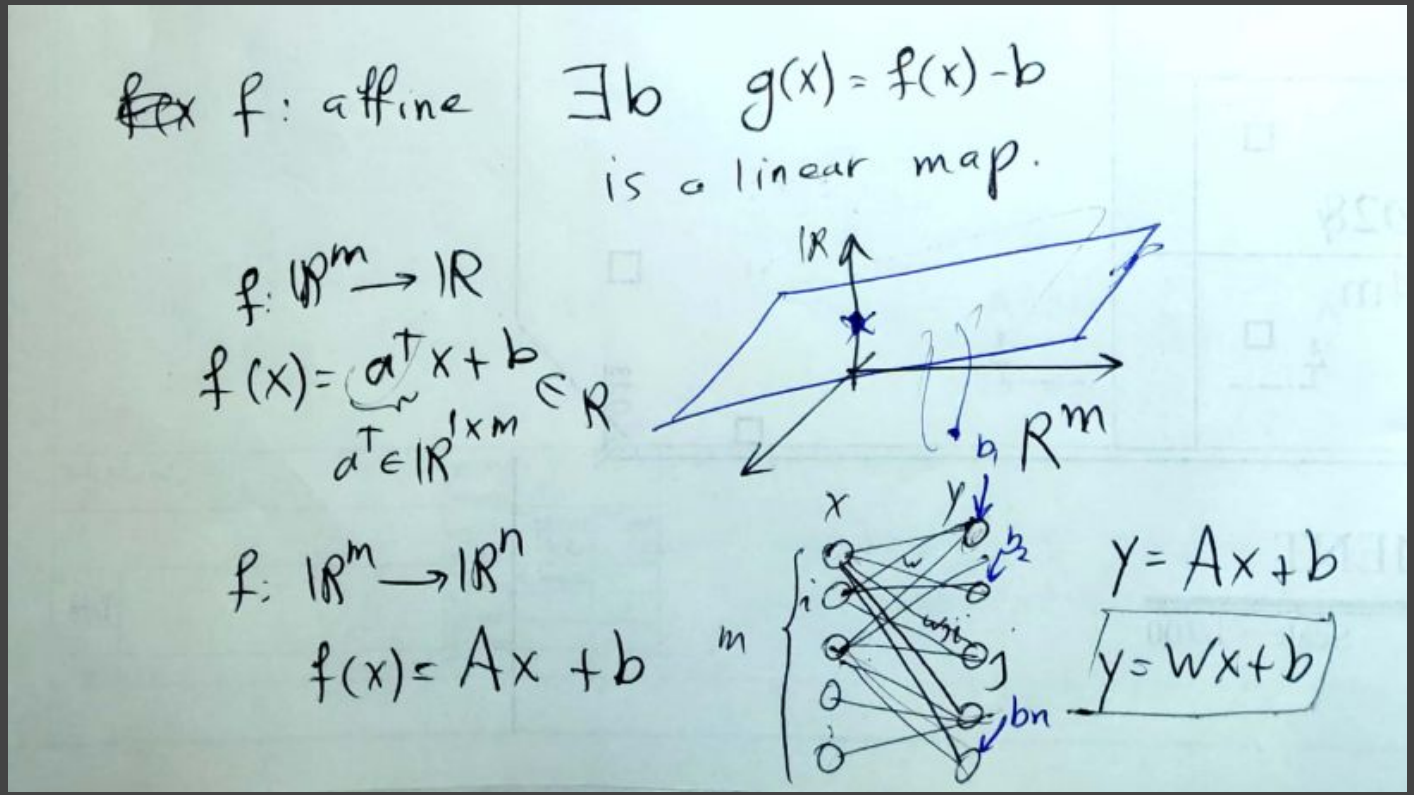
K. N. Toosi
University of Technology

$$\underbrace{f(x)-f(x_0)}_{y'} = a\underbrace{(x-x_0)}_{x'}$$

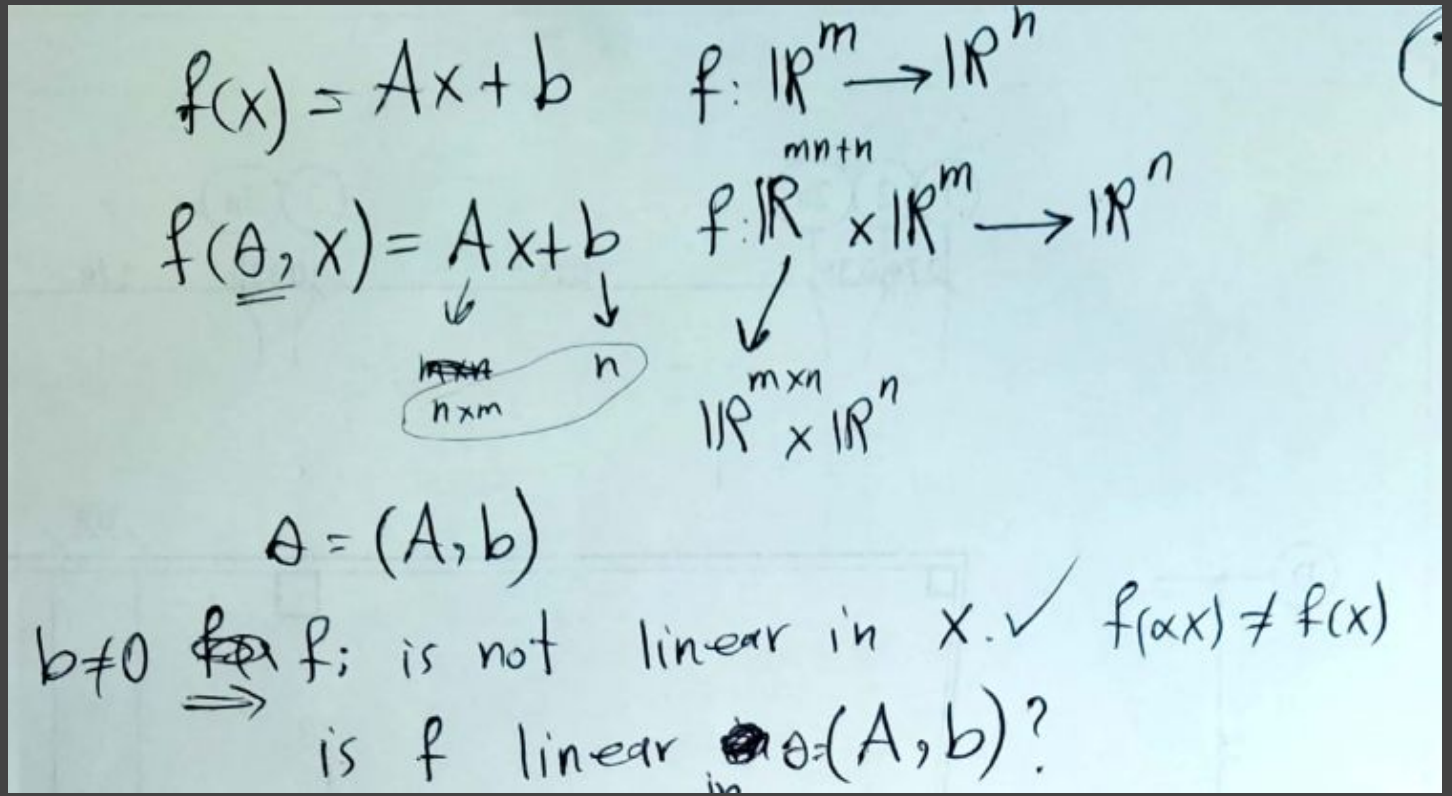
$$f(x) = ax - ax_0 + f(x_0)$$
$$= ax + (f(x_0) - ax_0)$$
$$= ax + b$$



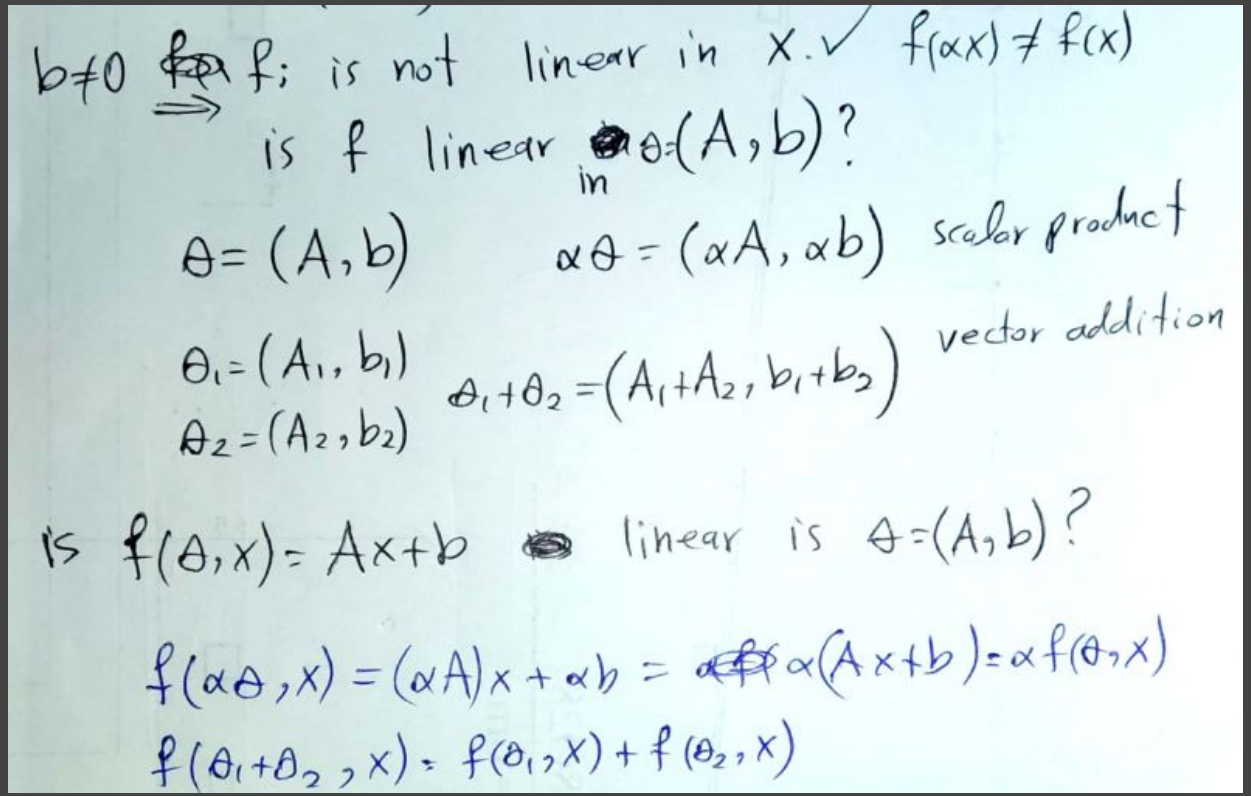
Affine functions



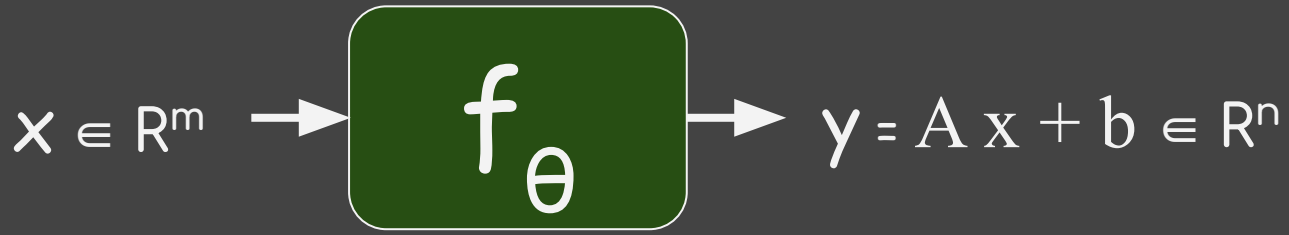
Affine functions



Affine functions

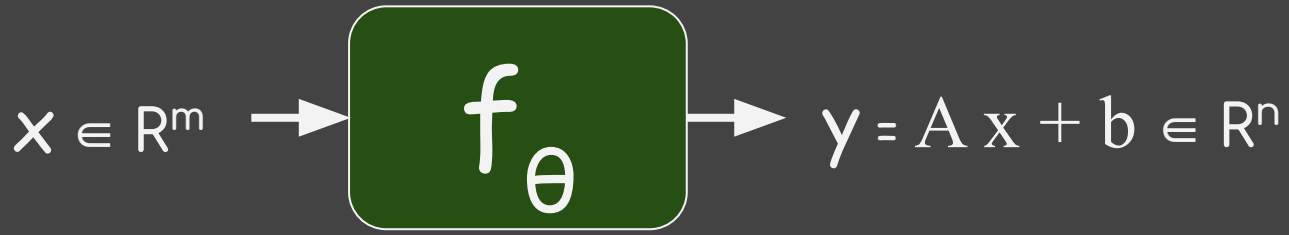


Example: Linear Regression



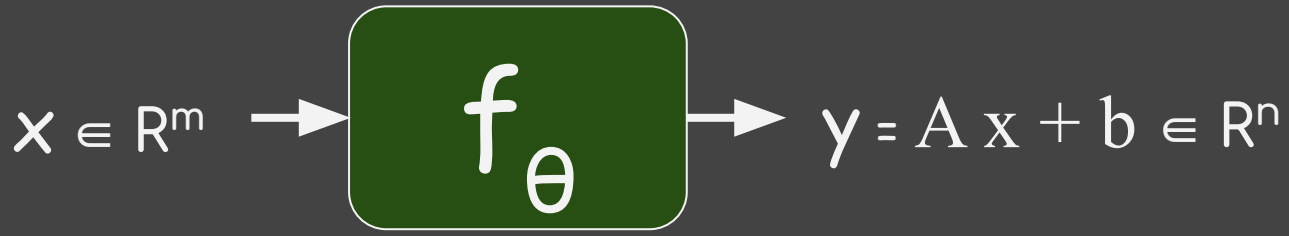
Example: Linear Regression

K. N. Toosi
University of Technology



A: ? by ? matrix
b: ?-D vector

Example: Linear Regression

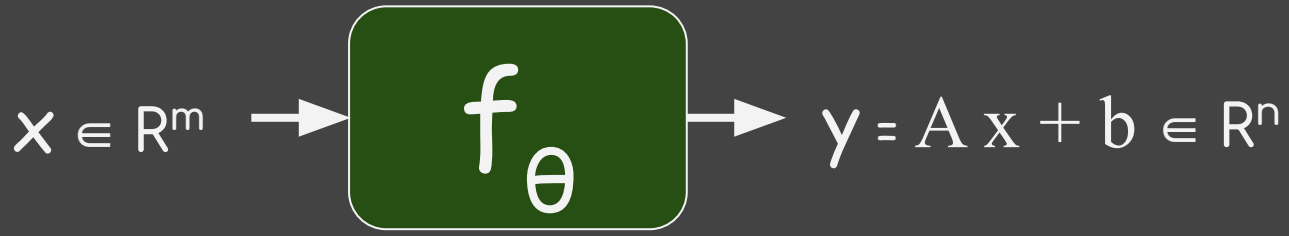


A: n by m matrix

b: n-D vector

Example: Linear Regression

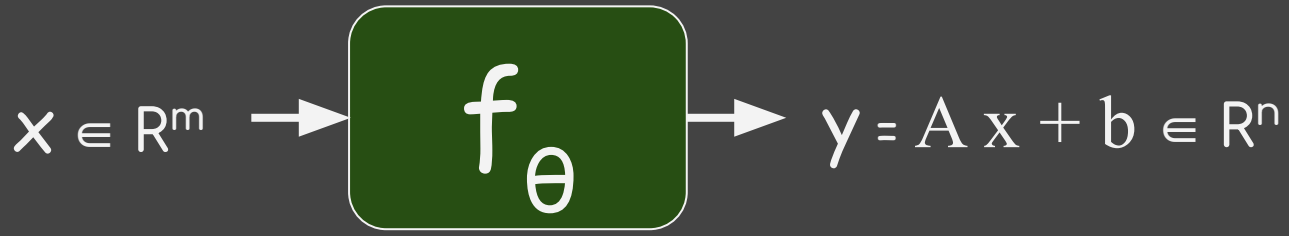
K. N. Toosi
University of Technology



$$y = f(\theta, x)$$
$$\theta = ?$$

Example: Linear Regression

K. N. Toosi
University of Technology



$$y = f(\theta, x)$$
$$\theta = (A, b)$$

Example: Linear Regression

K. N. Toosi
University of Technology

$f(x) = Ax + b$

$f: \mathbb{R}^m \longrightarrow \mathbb{R}^n$

is f linear?

$f(\alpha x) = A(\alpha x) + b = \alpha Ax + b$

$\alpha f(x) = \alpha Ax + \alpha b$

$\Rightarrow f$ is not linear in general $(b \neq 0)$

Affine maps

K. N. Toosi
University of Technology

$g(x) = f(x+x_0) - f(x_0) = A(x+x_0) + b - (Ax_0 + b) = Ax$

$g(x) = f(x) - f(0) = Ax$

$g(x) = f(x) - b = Ax$

$f(x) = Ax + b$

affine function

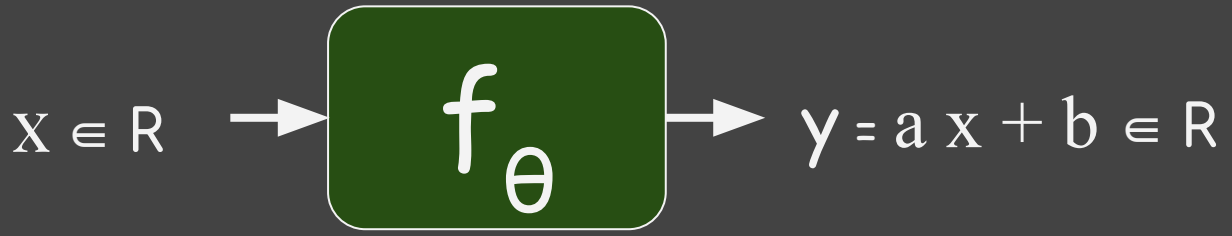
$f(x) : \mathbb{R}^m \to \mathbb{R}^n \quad \exists v \in \mathbb{R}^n \quad g(x) = f(x) - v$

linear

$g(x) = f(x - x_0) - f(x_0)$

Example: Linear Regression

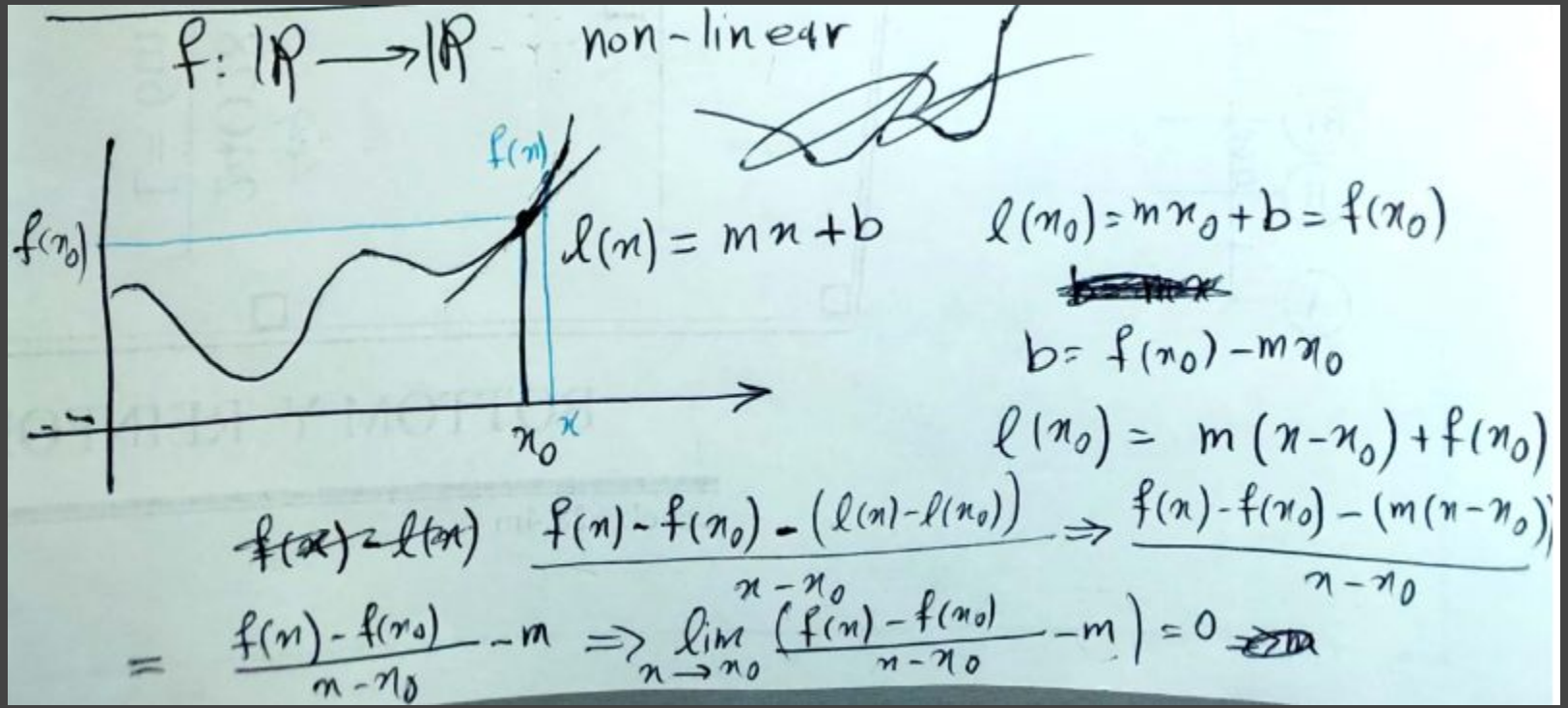
K. N. Toosi
University of Technology



$$y = f(\theta, x)$$
$$\theta = ?$$

nonlinear functions

K. N. Toosi
University of Technology



nonlinear functions: linearization

K. N. Toosi
University of Technology