SES 598: Space Robotics and AI

Lecture 3, Parameter estimation, January 20, 2026

Jnaneshwar Das
Alberto Behar Associate Research Professor

Center for Global Discovery and Conservation Science

Distributed Robotic Exploration And Mapping Systems Laboratory

WEEK	TOPICS	LECTURES & ASSIGNMENTS	RELATED RESOURCES
1-3 (Jan 13-Jan 30) State estimation and Controls	• Least squares and maximum likelihood estimation (MLE) • State space models and linear dynamical systems • State-estimation with Kalman and particle filters • PID control, linear quadratic regulator (LQR), and model predictive control (MPC) • Entry descent and landing (EDL), guidance navigation and control (GNC), and attitude determination and control system (ADCS)	Assignment 1: First-Order Boustrophedon Navigator (Lawnmower pattern) using ROS2 (Due: Jan 27, 2026)	**Papers:** • MPC for Quadrotor Flight • Mars 2020 EDL • Psyche Mission GNC **Tutorials:** • Parameter Estimation Tutorial
4-5 (Feb 3-Feb 20) Computer Vision and 3D Reconstruction	• Image formation and camera models • Feature detection and matching • Epipolar geometry and stereo vision • Structure from Motion (SfM) • Multi-View Stereo (MVS)	Assignment 2: Optimal Control of Cart-Pole System with LQR (Due: Feb 17, 2026) Assignment 3: ORBSLAM3 with ROS2 on PX4 SITL drone at Bishop Fault Scarp scene (Due: Feb 24, 2026)	**Papers:** • DUSt3R • SLAM Survey • COLMAP: SfM Revisited • ORB-SLAM **Tutorials:** • Random Sample Consensus (RANSAC) Tutorial • Multi-View Geometry Tutorial
6 (Feb 24-Mar 3) Scene Representation, View Synthesis, and Scene Analysis	• Scene representation: Orthomaps, pointcloud, mesh models, voxel grids, implicit surface models, and surfels	Assignment 4: View synthesis and scene analysis on Apollo 17 and	**Papers:** • Gaussian Splatting SLAM

Administrative

Assignments (20%)

Coding assignments throughout the semester to reinforce learning concepts and practical skills.

Midterm Project (20%)

A comprehensive project due mid-semester that integrates core concepts covered in the first half.

Final Project (50%)

A major project that demonstrates mastery of course concepts, including implementation and documentation. This can be a continuation of the midterm project

Class Participation (10%)

Active participation in class discussions, group activities, and engagement with course material.

Office hours: Fridays 11a-12p (Zoom or in person).

DREAMS Laboratory - WCPH 500L2

Recap of lecture 2

1) Traveling salesperson problem (TSP), computational complexity, …many interesting problems in sampling and exploration are hard problems.

2) What is sampling? Can we quickly estimate population sizes through sampling? Estimating number of students in class through uniform random sampling.

3) Foundation models in robotics, taxonomy. LLMs, diffusion models, transformers, CNNs, CLIPs, robotics transformer

4) Robotic manipulators, forward and inverse kinematics.

5) Kalman filter for Apollo missions, state-estimation for a space-craft, how does Kalman filtering work? Time domain vs frequency domain (Wiener filter), Kalman filter is time domain, state-space implementation, on a digital computer.

6) Kalman filter's predict-correct loop, as a recursive filter.

7) Kalman filter uses leasts-squares? What is least-squares? What is linear regression?

8) Hawaii islands formation, can we estimate plate velocity from measured times since creation, and distance between islands?

9) Least-squares minimizes sum of squares errors. Solution for model weights is the Moore-Penrose pseudo-inverse.

Apollo mission GNC

Rudolf Kalman received the National Medal of Science on Oct. 7, 2009, from President Barack Obama.

A New Approach to Linear Filtering and Prediction Problems[1]

R. E. KALMAN
Research Institute for Advanced Study,[2]
Baltimore, Md.

The classical filtering and prediction problem is re-examined using the Bode-Shannon representation of random processes and the "state transition" method of analysis of dynamic systems. New results are:

(1) The formulation and methods of solution of the problem apply without modification to stationary and nonstationary statistics and to growing-memory and infinite-memory filters.

(2) A nonlinear difference (or differential) equation is derived for the covariance matrix of the optimal estimation error. From the solution of this equation the coefficients of the difference (or differential) equation of the optimal linear filter are obtained without further calculations.

(3) The filtering problem is shown to be the dual of the noise-free regulator problem.

The new method developed here is applied to two well-known problems, confirming and extending earlier results.

The discussion is largely self-contained and proceeds from first principles; basic concepts of the theory of random processes are reviewed in the Appendix.

Introduction

An important class of theoretical and practical problems in communication and control is of a statistical nature. Such problems are: (i) Prediction of random signals; (ii) separation of random signals from random noise; (iii) detection of signals of known form (pulses, sinusoids) in the presence of random noise.

In his pioneering work, Wiener [1][3] showed that problems (i) and (ii) lead to the so-called Wiener-Hopf integral equation; he also gave a method (spectral factorization) for the solution of this integral equation in the practically important special case of stationary statistics and rational spectra.

Many extensions and generalizations followed Wiener's basic work. Zadeh and Ragazzini solved the finite-memory case [2]. Concurrently and independently of Bode and Shannon [3], they also gave a simplified method [2] of solution. Booton discussed the nonstationary Wiener-Hopf equation [4]. These results are now in standard texts [5-6]. A somewhat different approach along these main lines has been given recently by Darlington [7]. For extensions to sampled signals, see, e.g., Franklin [8], Lees [9]. Another approach based on the eigenfunctions of the Wiener-Hopf equation (which applies also to nonstationary problems whereas the preceding methods in general don't), has been pioneered by Davis [10] and applied by many others, e.g., Shinbrot [11], Blum [12], Pugachev [13], Solodovnikov [14].

In all these works, the objective is to obtain the specification of a linear dynamic system (Wiener filter) which accomplishes the prediction, separation, or detection of a random signal.[4]

[1] This research was supported in part by the U. S. Air Force Office of Scientific Research under Contract AF 49 (638)-382.

[2] 7212 Bellona Ave.

[3] Numbers in brackets designate References at end of paper.

[4] Of course, in general these tasks may be done better by nonlinear filters. At present, however, little or nothing is known about how to obtain (both theoretically and practically) these nonlinear filters.

Contributed by the Instruments and Regulators Division and presented at the Instruments and Regulators Conference, March 29– April 2, 1959, of The American Society of Mechanical Engineers.

 Manuscript received at ASME Headquarters, February 24, 1959. Paper No. 59—IRD-11.

Present methods for solving the Wiener problem are subject to a number of limitations which seriously curtail their practical usefulness:

(1) The optimal filter is specified by its impulse response. It is not a simple task to synthesize the filter from such data.

(2) Numerical determination of the optimal impulse response is often quite involved and poorly suited to machine computation. The situation gets rapidly worse with increasing complexity of the problem.

(3) Important generalizations (e.g., growing-memory filters, nonstationary prediction) require new derivations, frequently of considerable difficulty to the nonspecialist.

(4) The mathematics of the derivations are not transparent. Fundamental assumptions and their consequences tend to be obscured.

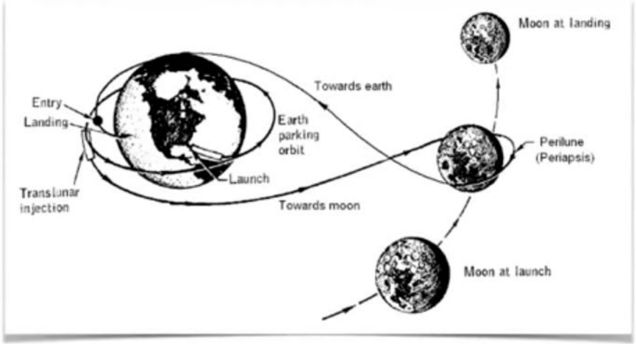
This paper introduces a new look at this whole assemblage of problems, sidestepping the difficulties just mentioned. The following are the highlights of the paper:

(5) *Optimal Estimates and Orthogonal Projections.* The Wiener problem is approached from the point of view of conditional distributions and expectations. In this way, basic facts of the Wiener theory are quickly obtained; the scope of the results and the fundamental assumptions appear clearly. It is seen that all statistical calculations and results are based on first and second order averages; no other statistical data are needed. Thus difficulty (4) is eliminated. This method is well known in probability theory (see pp. 75–78 and 148–155 of Doob [15] and pp. 455–464 of Loève [16]) but has not yet been used extensively in engineering.

(6) *Models for Random Processes.* Following, in particular, Bode and Shannon [3], arbitrary random signals are represented (up to second order average statistical properties) as the output of a linear dynamic system excited by independent or uncorrelated random signals ("white noise"). This is a standard trick in the engineering applications of the Wiener theory [2–7]. The approach taken here differs from the conventional one only in the way in which linear dynamic systems are described. We shall emphasize the concepts of *state* and *state transition;* in other words, linear systems will be specified by systems of first-order difference (or differential) equations. This point of view is

Transactions of the ASME–Journal of Basic Engineering, 82 (Series D): 35-45.

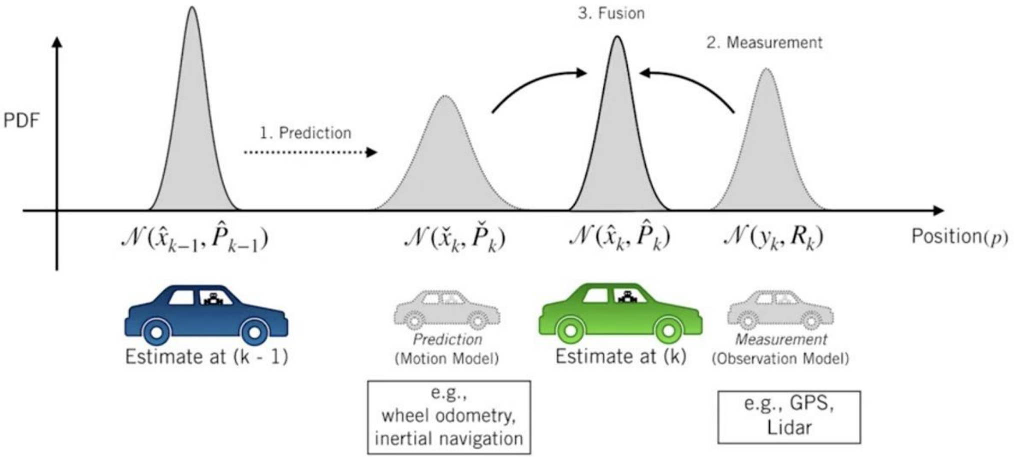
Apollo Guidance Computer



The (extended) Kalman Filter became widely known after its use in the Apollo Guidance Computer for circumlunar navigation.

Kalman Filter - Part 1, Prof. Jonathan Kelly, Univ. of Toronto https://www.youtube.com/watch?v=LioOvUZ1MiM

The Kalman Filter | Prediction and Correction



1: **Algorithm Kalman_filter**($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

2: $\bar{\mu}_t = A_t\, \mu_{t-1} + B_t\, u_t$

3: $\bar{\Sigma}_t = A_t\, \Sigma_{t-1}\, A_t^T + R_t$

4: $K_t = \bar{\Sigma}_t\, C_t^T (C_t\, \bar{\Sigma}_t\, C_t^T + Q_t)^{-1}$

5: $\mu_t = \bar{\mu}_t + K_t (z_t - C_t\, \bar{\mu}_t)$

6: $\Sigma_t = (I - K_t\, C_t)\, \bar{\Sigma}_t$

7: return μ_t, Σ_t

Kalman gain applied on discrepancy between predicted and observed states

Kalman Filter - Part 1, Prof. Jonathan Kelly, Univ. of Toronto

https://www.youtube.com/watch?v=LioOvUZ1MiM

1	Kilauea	0	0.4
3	Mauna Kea	54	0.375
5	Kohala	100	0.43
6	Haleakala	182	0.75
7	Kahoolawe	185	1.03
8	West Maui	221	1.32
9	Lanai	226	1.28
10	East Molokai	256	1.76
11	West Molokai	280	1.9
12	Koolau	339	2.6
13	Waianae	374	3.7
14	Kauai	519	5.1
15	Niihau	565	4.89
15A	Kaula	600	4
17	Nihoa	780	7.2
20	Unnamed	913	9.2
23	Necker	1058	10.3
26	La Perouse Pinnacle	1209	12
27	Brooks Bank	1256	13
30	Gardner Pinnacles	1435	12.3
36	Laysan	1818	19.9
37	Northampton Bank	1841	26.6
50	Pearl and Hermes Re	2281	20.6
52	Midway	2432	27.7
57	Unnamed	2600	28
63	Unnamed	2825	27.4
65	Colohan	3128	38.6
65A	Abbott	3280	38.7
67	Daikakuji	3493	42.4
69	Yuryaku	3520	43.4
72	Kimmei	3668	39.9
74	Koko (southern)	3758	48.1
81	Ojin	4102	55.2
83	Jingu	4175	55.4
86	Nintoku	4452	56.2
90	Suiko (southern)	4794	59.6
91	Suiko (central)	4860	64.7

Models and parameter estimation

1. Age and spatial spread of the islands and seamounts of Hawaii are considered to be drawn from a linear model, corrupted by additive Gaussian noise. The chain of these Hawaii islands/seamounts was formed by the movement of the Pacific Plate, over a volcanic hotspot. A dataset is provided with each row having island/seamount ID, name, distance in kilometers to the Kilauea volcano, and island age in million years. Write Python code to estimate the Pacific Plate drift velocity. The units will be

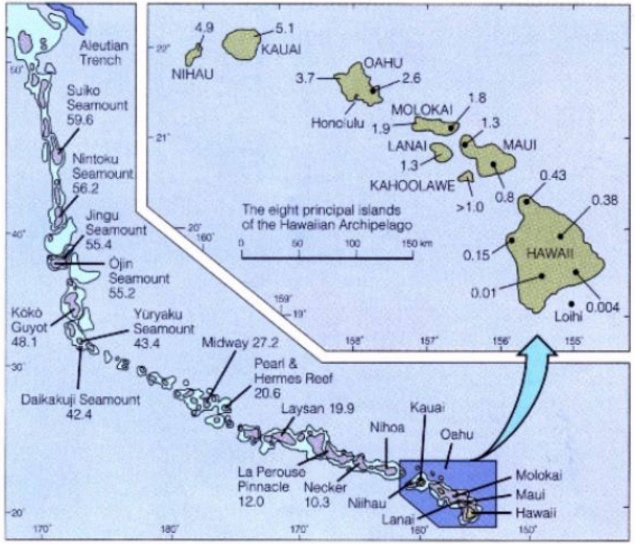


Figure 1: Hawai islands and seamounts were formed by volcanic eruptions, as the Pacific Plate moved very slowly. Courtesy: Kenneth Hon, University of Hawaii, Hilo

,

kilometers per million years. What principles did you use for this parameter estimation problem? Can you comment on how good this estimate is?

https://docs.google.com/spreadsheets/d/17osMD2ChK1pTYFdMv2MBvmtPkJmCDlTsrOK3Ue37TLs/edit?usp=sharing

Least-squares estimation

Fitting a linear model to noisy data

$$y = mx + c$$

Linear model, how many parameters?

Least-squares estimation

Fitting a linear model to noisy data

$$y = mx + c$$

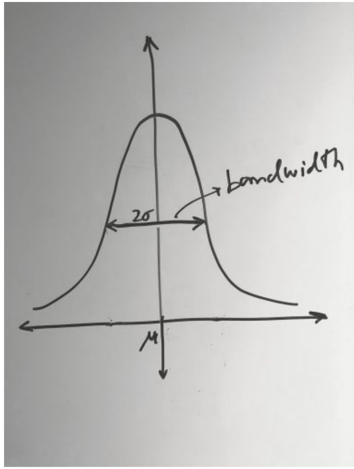
Linear model, how many parameters?

$$y = mx + c + \epsilon$$

Observations from a linear model, with noise

$$\epsilon \sim \mathcal{N}(0,\, \sigma^2)$$

$$\mathcal{N}(\mu, \sigma^2) \sim \frac{1}{\sqrt{2\pi\sigma^2}} exp^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



Least-squares estimation

Fitting a linear model to noisy data

$$y = mx + c$$

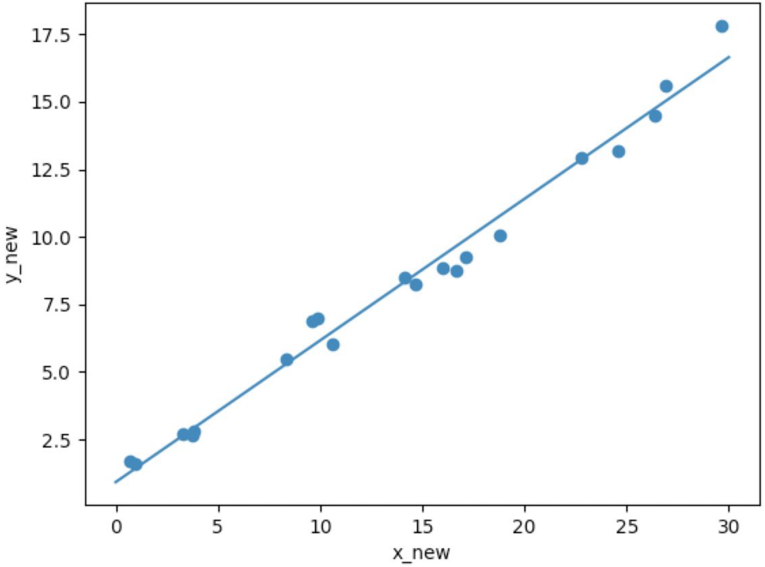
Linear model, how many parameters?

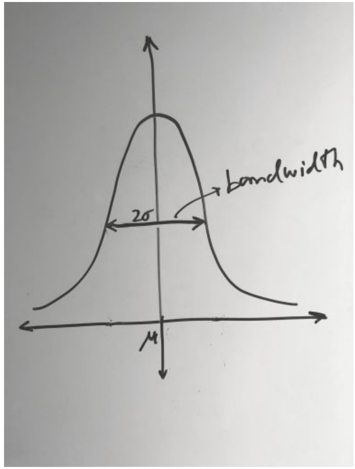
$$y = mx + c + \epsilon$$

Observations from a linear model, with noise

$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$

$$\mathcal{N}(\mu, \sigma^2) \sim \frac{1}{\sqrt{2\pi\sigma^2}} exp^{-\frac{(x-\mu)^2}{2\sigma^2}}$$





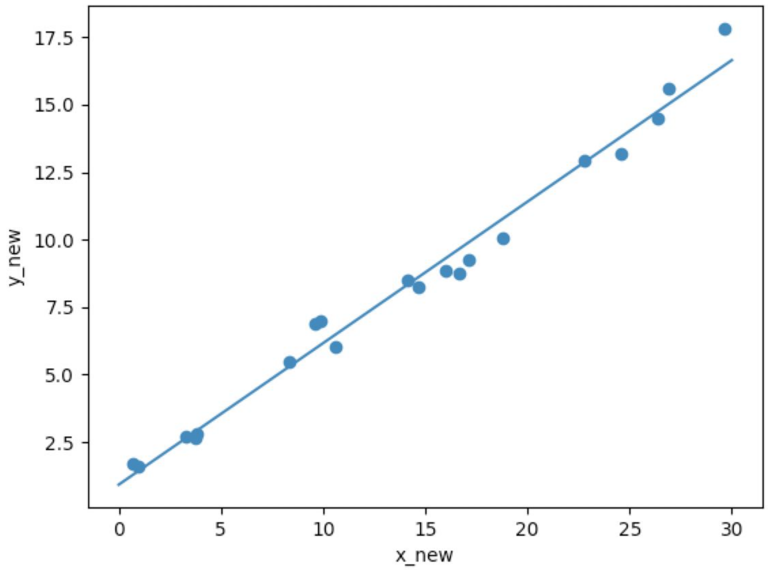
Parameter Estimation

Estimate parameters for a linear model, given observations with noise

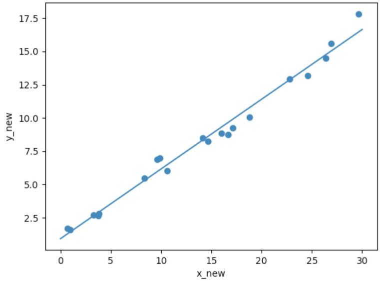
$$y = mx + c$$

$$y = w_0 1 + w_1 x$$

$$\mathbf{x} = \begin{bmatrix} 1 \\ x \end{bmatrix} \quad \mathbf{w} = \begin{bmatrix} c \\ m \end{bmatrix}$$



Parameter Estimation



Estimate parameters for a linear model, given observations with noise

$$y = mx + c$$

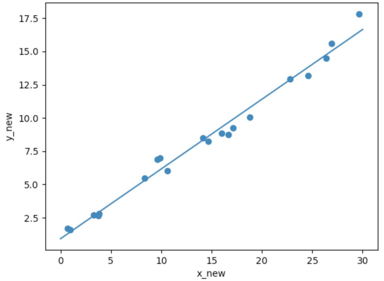
$$y = w_0 1 + w_1 x$$

$$\mathbf{x} = \begin{bmatrix} 1 \\ x \end{bmatrix} \quad \mathbf{w} = \begin{bmatrix} c \\ m \end{bmatrix}$$

$$y = \sum_{i=0}^{D} w_i x_i$$

$$y = \mathbf{w}^T \mathbf{x}$$

Parameter Estimation

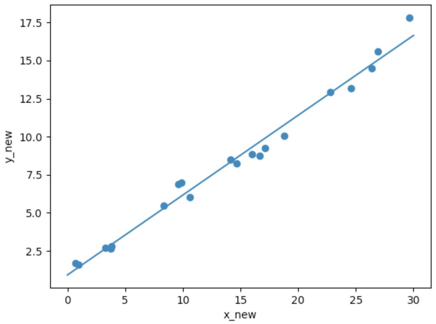


$$y = \mathbf{w}^T \mathbf{x} + \epsilon$$

$$X = \begin{bmatrix} x_0^1 & x_1^1 \\ x_0^2 & x_1^2 \\ \vdots & \vdots \\ x_0^N & x_1^N \end{bmatrix} = \begin{bmatrix} 1 & x^1 \\ 1 & x^2 \\ \vdots & \vdots \\ 1 & x^N \end{bmatrix}$$

$$Y = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{bmatrix}$$

Least squares parameter estimation



$$y = \mathbf{w}^T \mathbf{x} + \epsilon$$

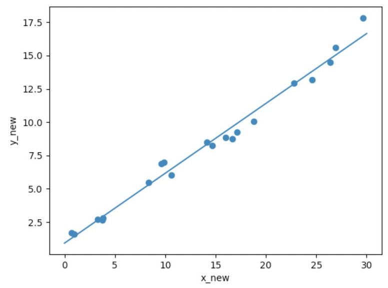
sum of square error

$$X = \begin{bmatrix} x_0^1 & x_1^1 \\ x_0^2 & x_1^2 \\ \vdots & \vdots \\ x_0^N & x_1^N \end{bmatrix} = \begin{bmatrix} 1 & x^1 \\ 1 & x^2 \\ \vdots & \vdots \\ 1 & x^N \end{bmatrix}$$

$$L = \Sigma_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2$$

$$Y = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{bmatrix}$$

Least squares parameter estimation



$$y = \mathbf{w}^T \mathbf{x} + \epsilon$$

sum of square error

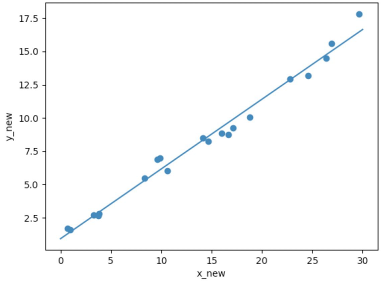
$$X = \begin{bmatrix} x_0^1 & x_1^1 \\ x_0^2 & x_1^2 \\ \vdots & \vdots \\ x_0^N & x_1^N \end{bmatrix} = \begin{bmatrix} 1 & x^1 \\ 1 & x^2 \\ \vdots & \vdots \\ 1 & x^N \end{bmatrix}$$

$$L = \Sigma_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2$$

$$Y = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{bmatrix}$$

$$\boxed{\arg\min_{\mathbf{w}} \sum_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2}$$

Least squares parameter estimation



$$y = \mathbf{w}^T \mathbf{x} + \epsilon$$

$$X = \begin{bmatrix} x_0^1 & x_1^1 \\ x_0^2 & x_1^2 \\ \vdots & \vdots \\ x_0^N & x_1^N \end{bmatrix} = \begin{bmatrix} 1 & x^1 \\ 1 & x^2 \\ \vdots & \vdots \\ 1 & x^N \end{bmatrix}$$

$$Y = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{bmatrix}$$

sum of square error

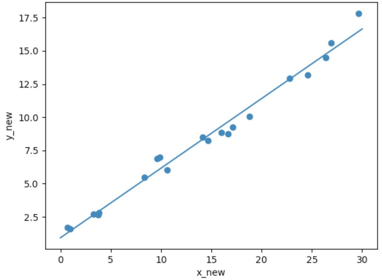
$$L = \Sigma_{i=1}^N (y^i - \mathbf{w}^T \mathbf{x}^i)^2$$

$$\boxed{\arg\min_{\mathbf{w}} \sum_{i=1}^N (y^i - \mathbf{w}^T \mathbf{x}^i)^2}$$

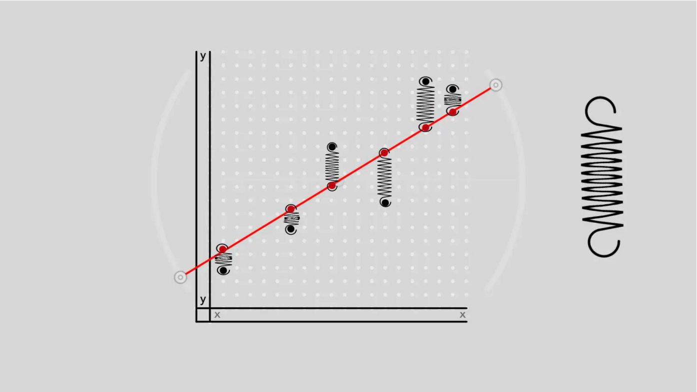
$$\boxed{\mathbf{w} = (X^T X)^{-1} X^T Y}$$

Moore-Penrose
pseudo inverse

Least squares parameter estimation



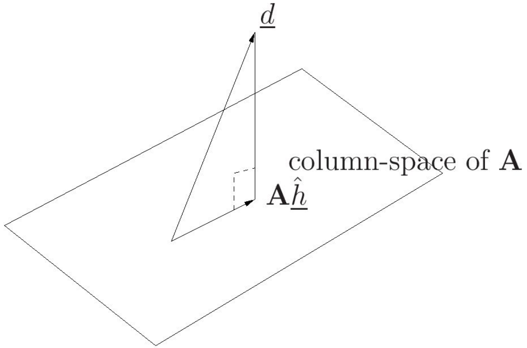
$$y = \mathbf{w}^T \mathbf{x} + \epsilon$$



sum of square error

$$L = \Sigma_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2$$

$$\boxed{\arg\min_{\mathbf{w}} \sum_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2}$$

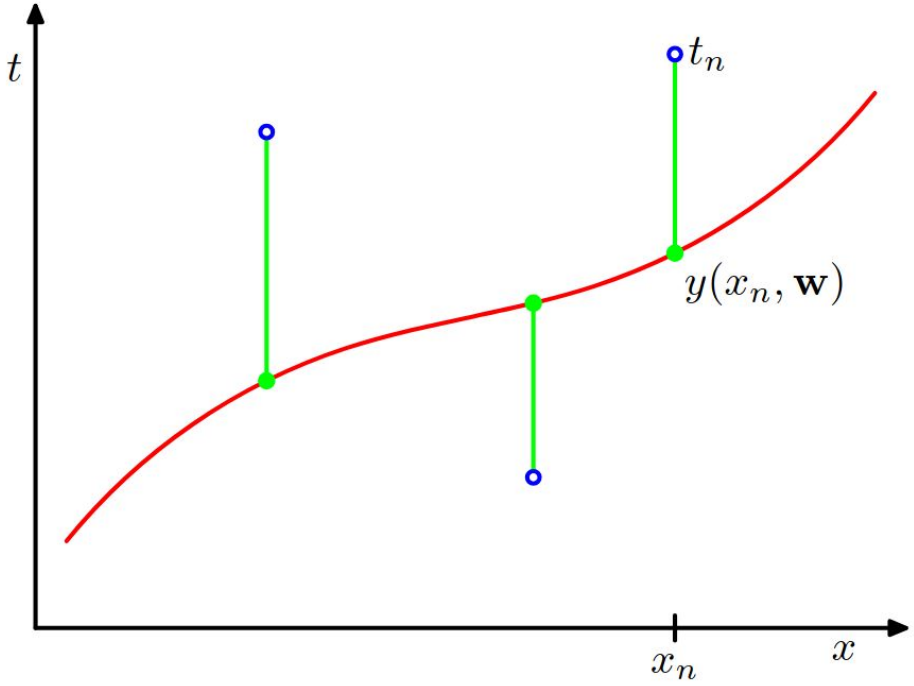


$$\boxed{\mathbf{w} = (X^T X)^{-1} X^T Y}$$

Moore-Penrose pseudo inverse

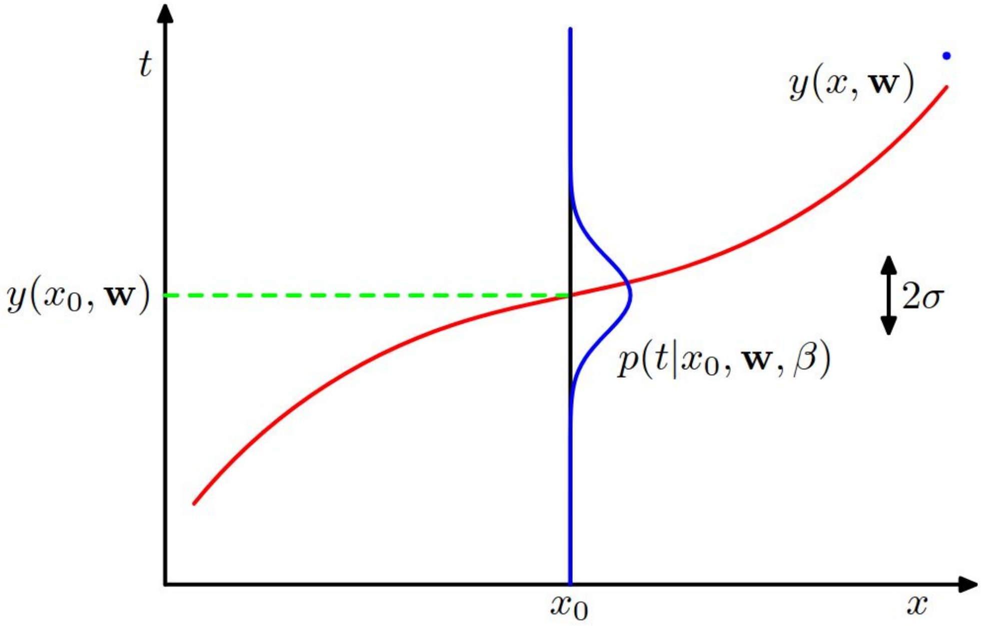
Figure 10.1: Geometric interpretation of orthogonal projection of the vector $\underline{d}$ onto the column-space of $\mathbf{A}$.

Maximum Likelihood Estimation



$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j \qquad (1.1)$$

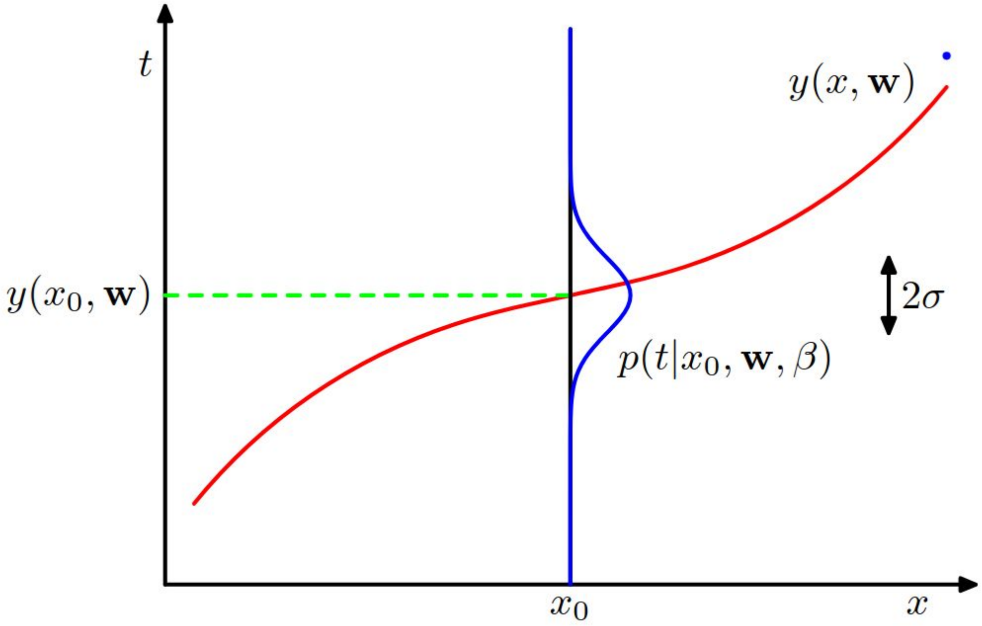
Maximum Likelihood Estimation (MLE)



Training data acquired from the process model, corrupted by Gaussian noise

$$p(t|x, \mathbf{w}, \beta) = \mathcal{N}\left(t|y(x, \mathbf{w}), \beta^{-1}\right) \tag{1.60}$$

Likelihood function



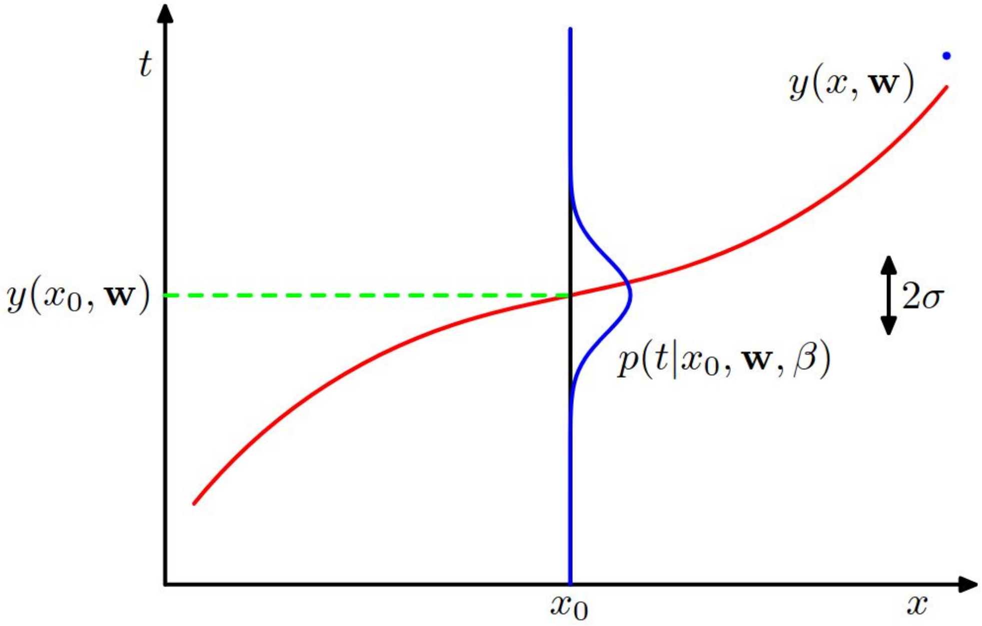
What is the likelihood of observed data?

$$p(t|x, \mathbf{w}, \beta) = \mathcal{N}\left(t|y(x, \mathbf{w}), \beta^{-1}\right) \tag{1.60}$$

$$p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}\left(t_n|y(x_n, \mathbf{w}), \beta^{-1}\right). \tag{1.61}$$

Log-likelihood

What is the log-likelihood of observed data?



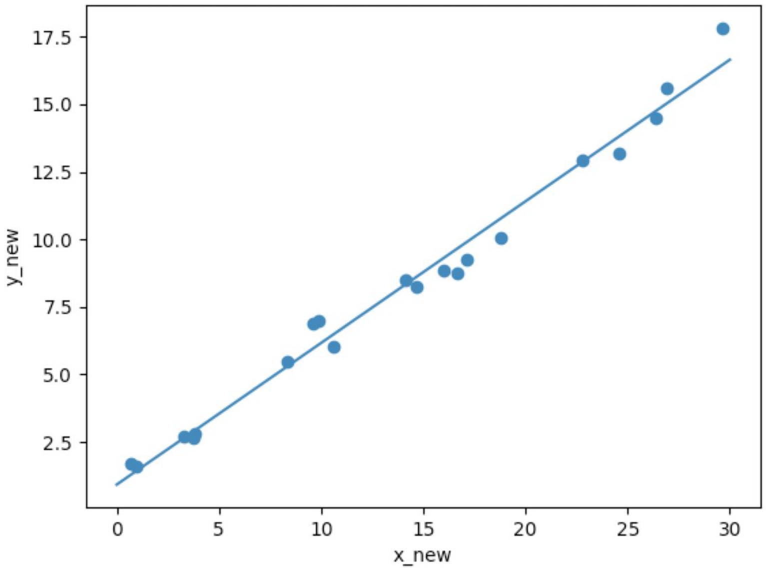
$$p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}\left(t_n | y(x_n, \mathbf{w}), \beta^{-1}\right). \tag{1.61}$$

$$\ln p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) = -\frac{\beta}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi). \tag{1.62}$$

Maximum Likelihood Estimation

The best estimate of weight vector, $\mathbf{w}^*$ is given by,

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \quad \mathcal{L}_{log}(\mathbf{w}; X, Y)$$
$$= \boxed{\arg\max_{\mathbf{w}} \quad -\sum_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2}$$



Maximum Likelihood Estimation ⇔ Least Squares Estimation, under Gaussian noise assumptions

$$\boxed{\arg\min_{\mathbf{w}} \quad \sum_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2} \quad \Longleftrightarrow \quad \boxed{\arg\max_{\mathbf{w}} \quad -\sum_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2}$$