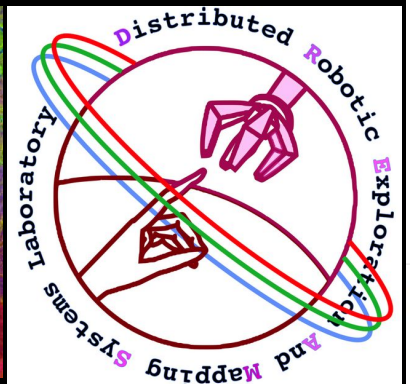
# SES 598: Space Robotics and AI

Lecture 2, Introduction, January 15, 2026

Jnaneshwar Das
Alberto Behar Associate Research Professor

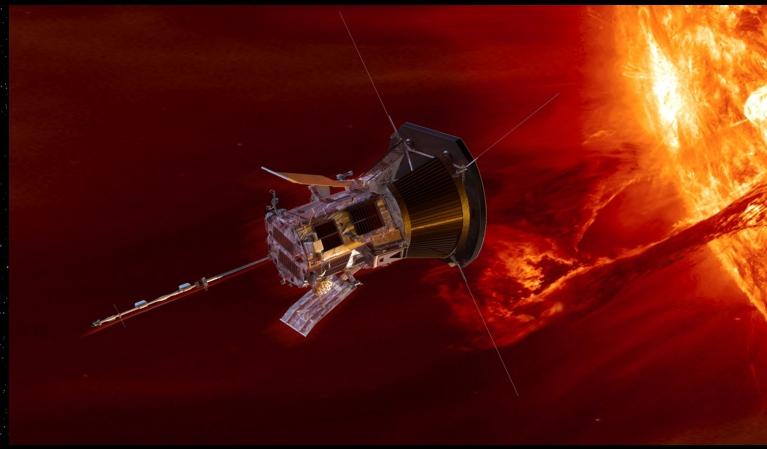




| WEEK | TOPICS | LECTURES & ASSIGNMENTS | RELATED RESOURCES |
|---|---|---|---|
| **1-3 (Jan 13-Jan 30)** State estimation and Controls | • Least squares and maximum likelihood estimation (MLE)<br>• State space models and linear dynamical systems<br>• State-estimation with Kalman and particle filters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

# Recap of Lecture 1

- What AI methods for space, will this course cover?
- What is a state-vector?
- What role do sensor noise and actuator noise play in state estimation?
- What is PID control?
- Can AI be used to tune a PID controller?
- What optimization problems can AI solve?
- Can AI fly a spacecraft?

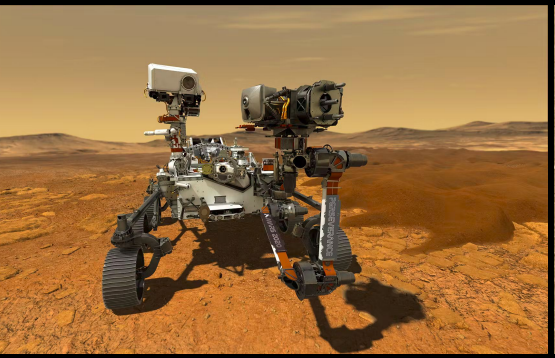
# 3Ds of Robotics and AI

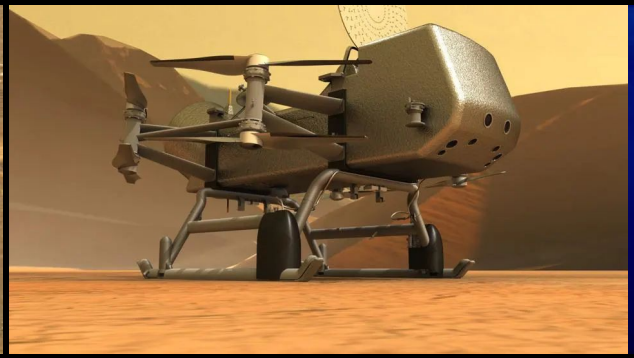


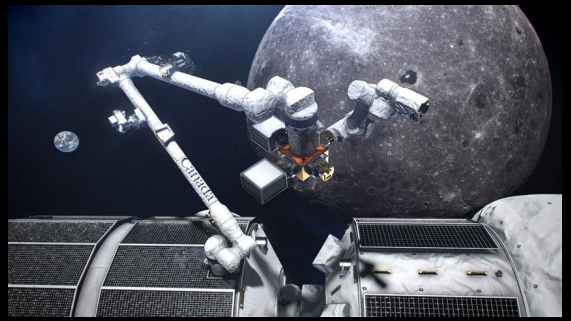
Dirty

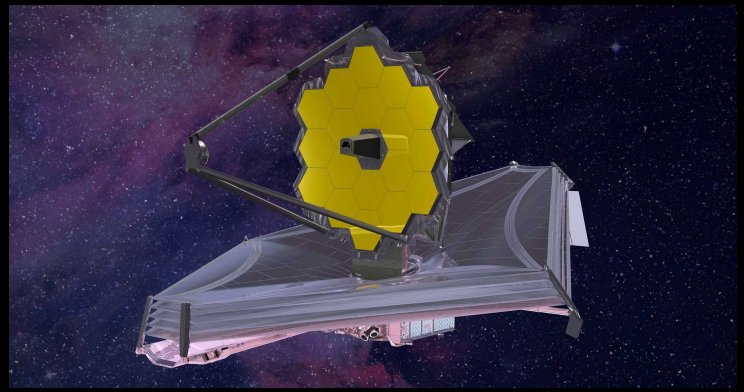
Dull

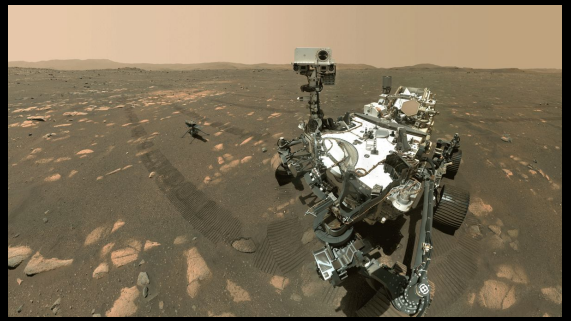
Dangerous

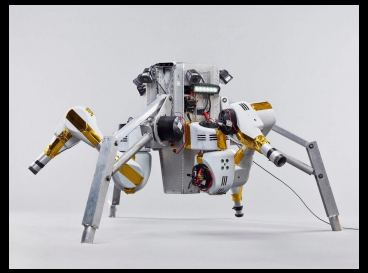


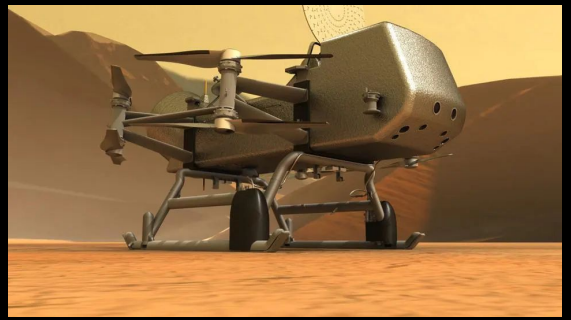






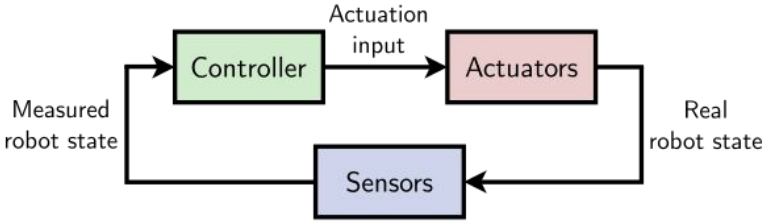


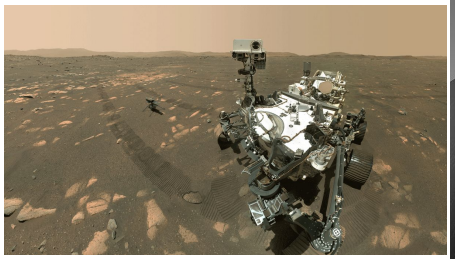




# Robotics and AI – the scope and promise

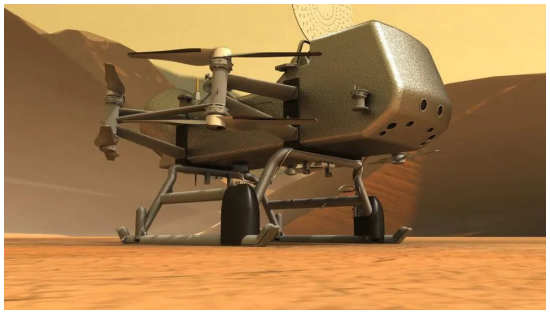
Close the loop – perception, cognition, action, communication

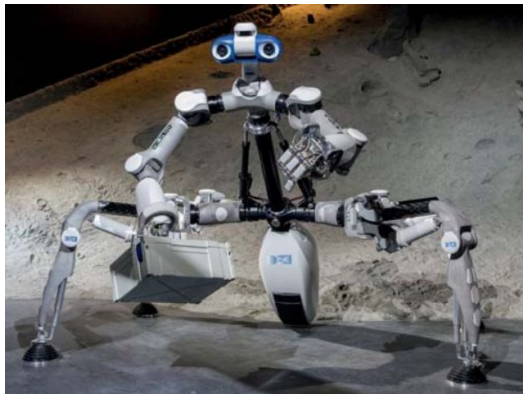


Scout robots, space probes, orbiters, landers

Debris removal,
Habitat management,
Planetary greenhouse,





Portable laboratories

Cobots - quadrupeds, humanoids

# Closing the loop

## AI

Automation and scaling of **data analysis**

domain experts




cyber-physical systems in environment



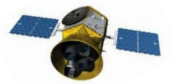
## Robotics

Automation and scaling of **data collection**

Questions, Challenges

(food, water, energy) ⇒ representations ⇒ abstractions ⇒ optimization ⇒ Insights, Solutions,Tools

# Closing the loop



**Cyber-Physical Twins**

domain experts

OpenUAV simulation testbed

Experimental design

models

plan and policies

cyber-physical systems in environment

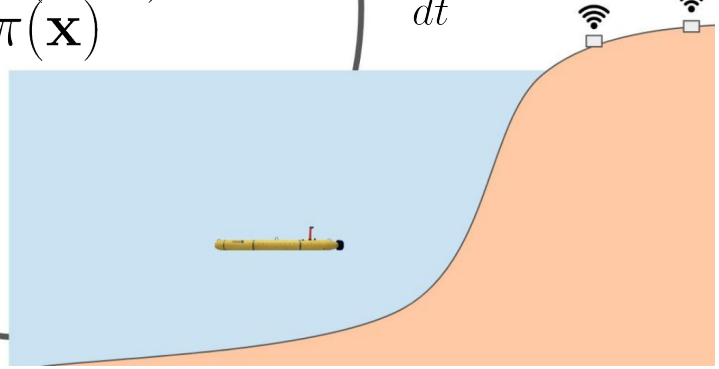
$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$$
$$\mathbf{u} = \pi(\mathbf{x})$$

$$\frac{d\mathbf{x}}{dt} \propto \mathbf{x}$$

DeepGIS decision support system

Modeling

data



## AI

Automation and scaling of **data analysis**

## Robotics

Automation and scaling of **data collection**

Questions, Challenges

(food, water, energy) ⇒ representations ⇒ abstractions ⇒ optimization ⇒ Insights, Solutions,Tools

# Computational thinking bridges disciplines

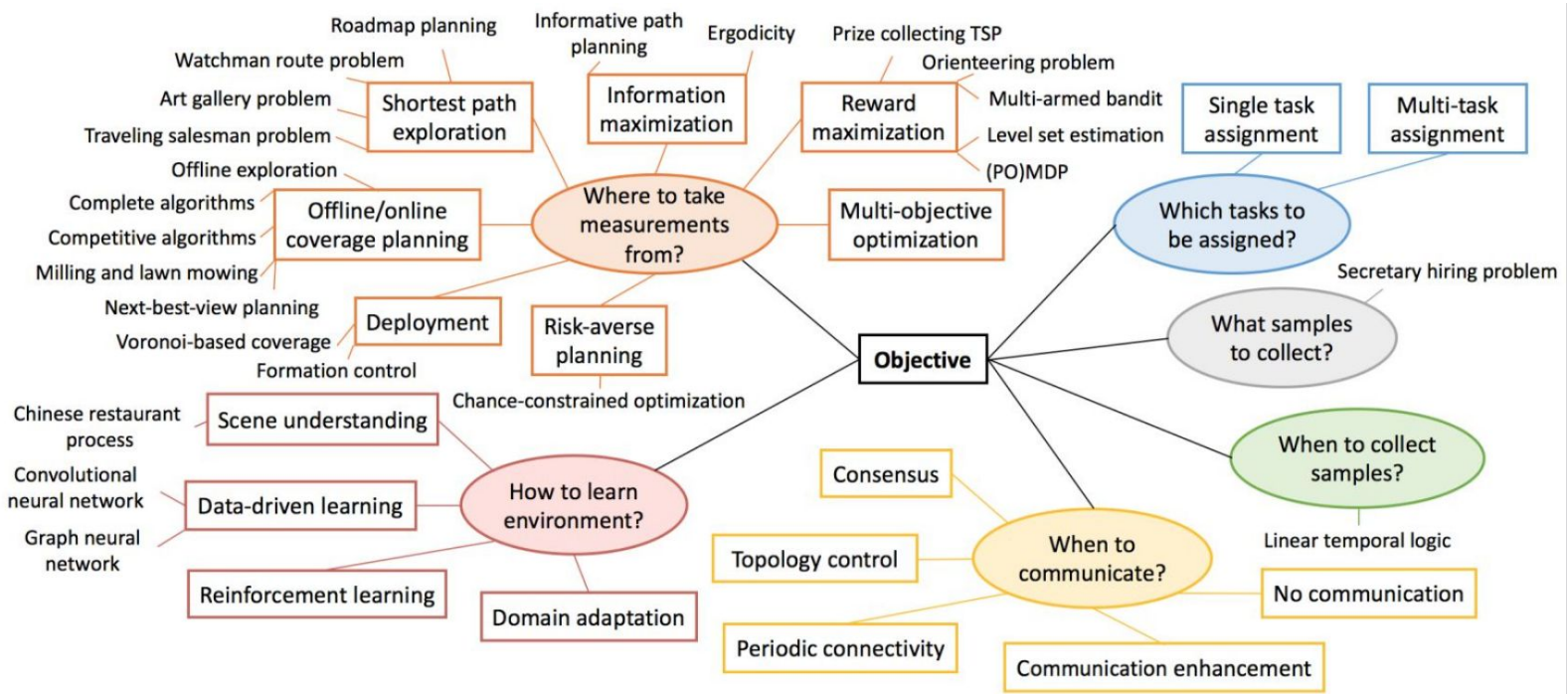



**Figure 4.1:** Taxonomy of decision-theoretic approaches, categorizing a diverse range of methods and algorithms under overarching objectives. This taxonomy emphasizes the relationships between various tasks, decision-making methodologies, and their corresponding scientific objectives.

*Yoonchang Sung, Zhiang Chen, Jnaneshwar Das and Pratap Tokekar (2023), "A Survey of Decision-Theoretic Approaches for Robotic Environmental Monitoring", Foundations and Trends® in Robotics: Vol. 11: No. 4, pp 225-315.*

# Computational thinking bridges disciplines

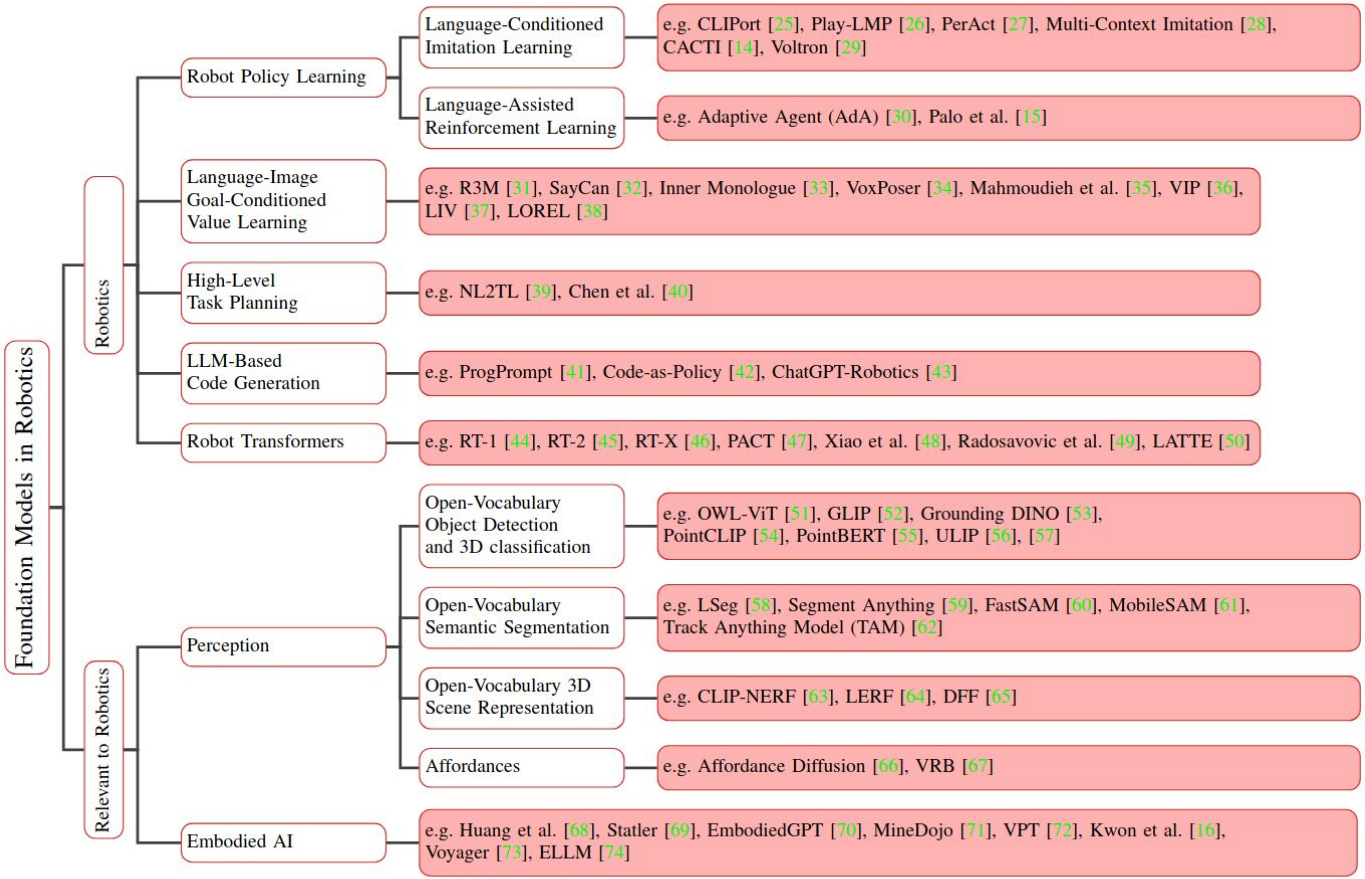



Fig. 1. Overview of Robotics Tasks Leveraging Foundation Models.

Firoozi, Roya, Johnathan Tucker, Stephen Tian, Anirudha Majumdar, Jiankai Sun, Weiyu Liu, Yuke Zhu et al. "Foundation models in robotics: Applications, challenges, and the future." *The International Journal of Robotics Research* (2023): 02783649241281508.

# Computational thinking bridges disciplines

[PDF] **Attention is all you need**
A Vaswani - Advances in Neural Information Processing Systems, 2017 - user.phil.hhu.de
**Attention** is **all you need** **Attention** is **all you need** ...
☆ Save 99 Cite Cited by 150580 Related articles »

## Attention Is All You Need


**Ashish Vaswani*** Google Brain avaswani@google.com
**Noam Shazeer*** Google Brain noam@google.com
**Niki Parmar*** Google Research nikip@google.com
**Jakob Uszkoreit*** Google Research usz@google.com
**Llion Jones*** Google Research llion@google.com
**Aidan N. Gomez*** † University of Toronto aidan@cs.toronto.edu
**Łukasz Kaiser*** Google Brain lukaszkaiser@google.com
**Illia Polosukhin*** ‡ illia.polosukhin@gmail.com



### Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

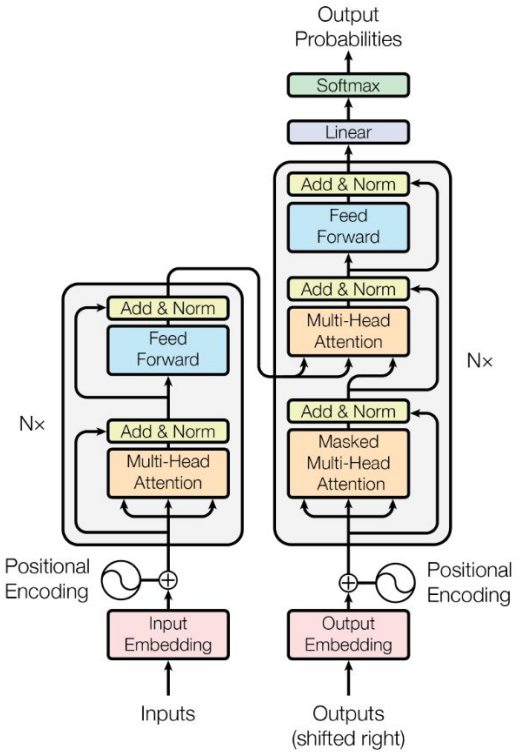




Figure 1: The Transformer - model architecture.

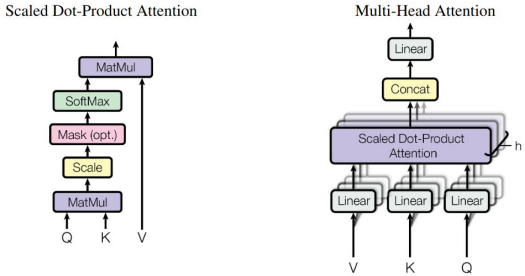



Figure 2: (left) Scaled Dot-Product Attention. (right) Multi-Head Attention consists of several attention layers running in parallel.

# Computational thinking bridges disciplines

deepseek

https://github.com/deepseek-ai/DeepSeek-V3/

## DeepSeek-V3 Technical Report

DeepSeek-AI

research@deepseek.com

### Abstract

We present DeepSeek-V3, a strong Mixture-of-Experts (MoE) language model with 671B total parameters with 37B activated for each token. To achieve efficient inference and cost-effective training, DeepSeek-V3 adopts Multi-head Latent Attention (MLA) and DeepSeekMoE architectures, which were thoroughly validated in DeepSeek-V2. Furthermore, DeepSeek-V3 pioneers an auxiliary-loss-free strategy for load balancing and sets a multi-token prediction training objective for stronger performance. We pre-train DeepSeek-V3 on 14.8 trillion diverse and high-quality tokens, followed by Supervised Fine-Tuning and Reinforcement Learning stages to fully harness its capabilities. Comprehensive evaluations reveal that DeepSeek-V3 outperforms other open-source models and achieves performance comparable to leading closed-source models. Despite its excellent performance, DeepSeek-V3 requires only 2.788M H800 GPU hours for its full training. In addition, its training process is remarkably stable. Throughout the entire training process, we did not experience any irrecoverable loss spikes or perform any rollbacks. The model checkpoints are available at https://github.com/deepseek-ai/DeepSeek-V3.


DeepSeek has improved Transformer architecture.
More efficient at learning.

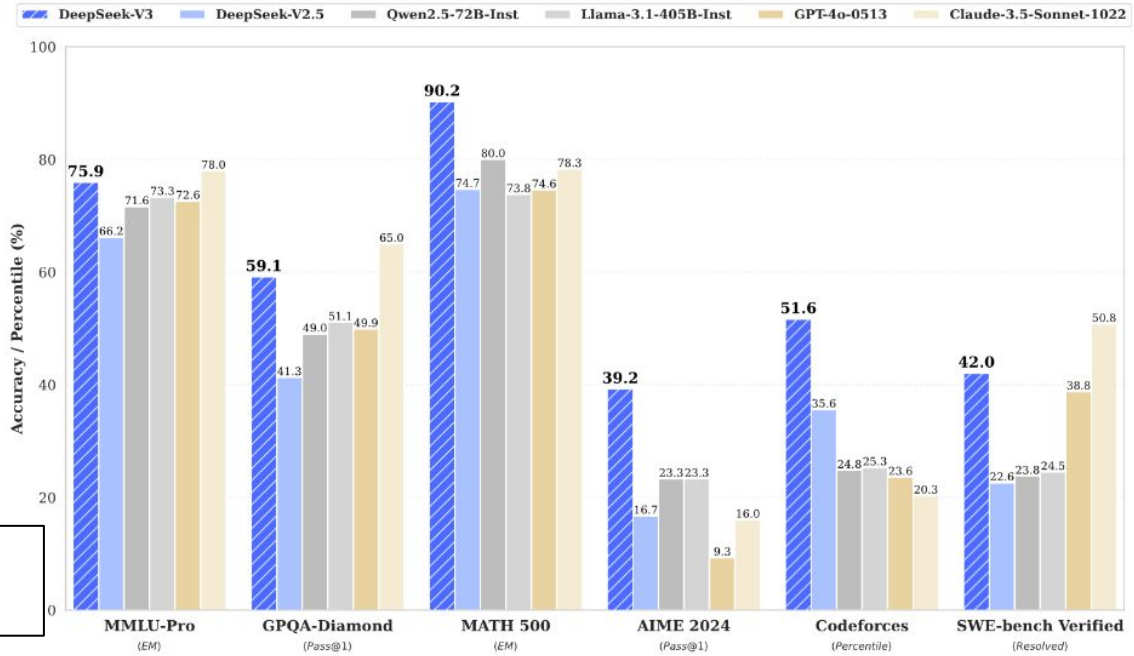
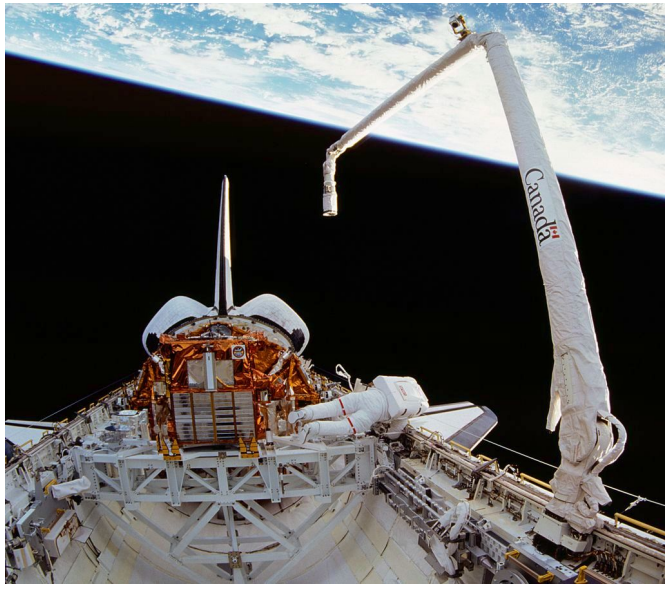
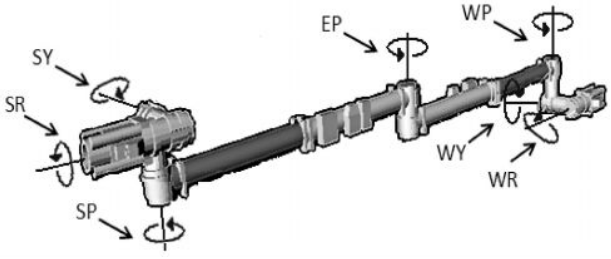


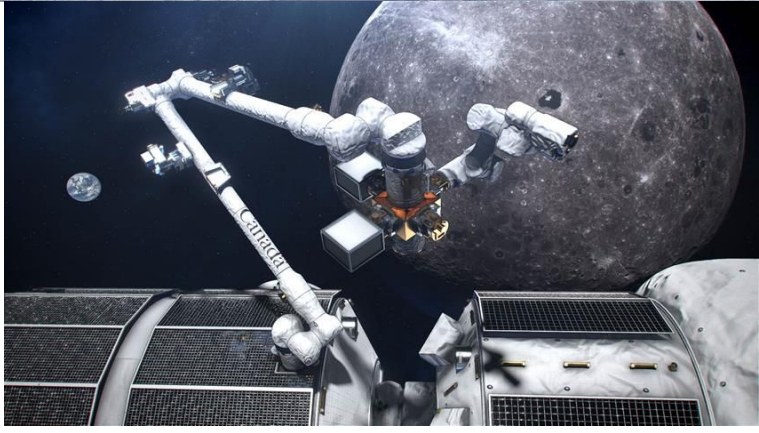

Figure 1 | Benchmark performance of DeepSeek-V3 and its counterparts.

# Robotic Manipulators



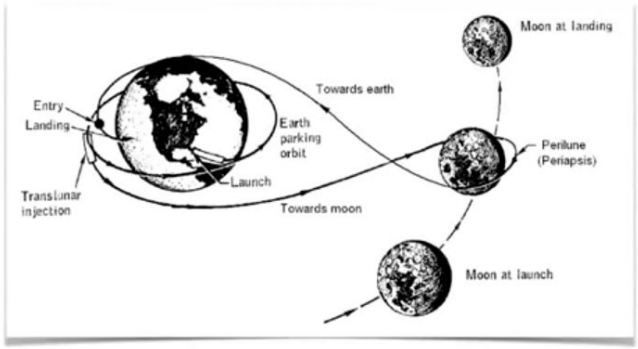

Canadarm 1, Space Shuttle
Canadarm 2, ISS
Canadarm 3 (proposed, moon)



# Apollo mission GNC

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

Apollo Guidance Computer




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

## 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]

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. This 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

[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.

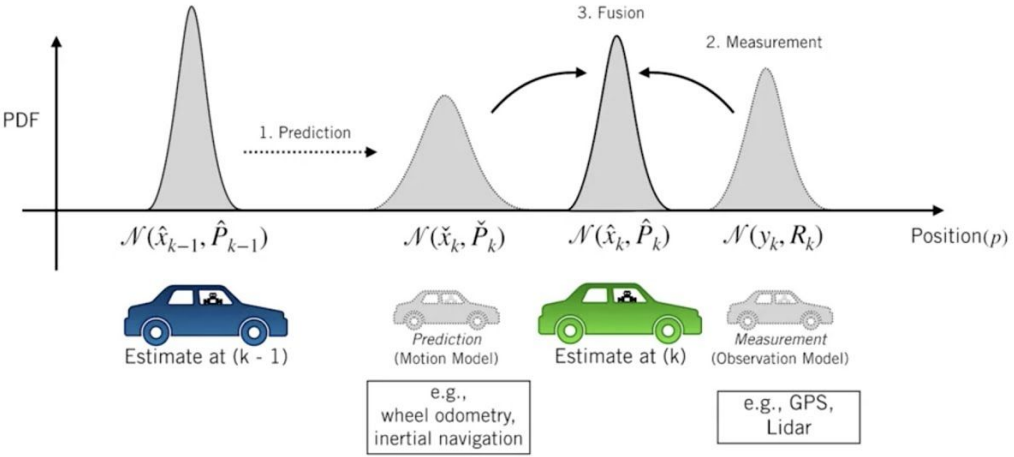
Contributed by the Instruments and Regulators Division and presented at the Instruments and Regulators Conference, March 29–April 2, 1959, of THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS.

NOTE: Statements and opinions advanced in papers are to be understood as individual expressions of their authors and not those of the Society. Manuscript received at ASME Headquarters, February 24, 1959. Paper No. 59—IRD-11.

Transactions of the ASME–Journal of Basic Engineering, 82 (Series D): 35-45. Copyright © 1960 by ASME

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

# The Kalman Filter | Prediction and Correction




Kalman gain applied on discrepancy between predicted and observed states

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

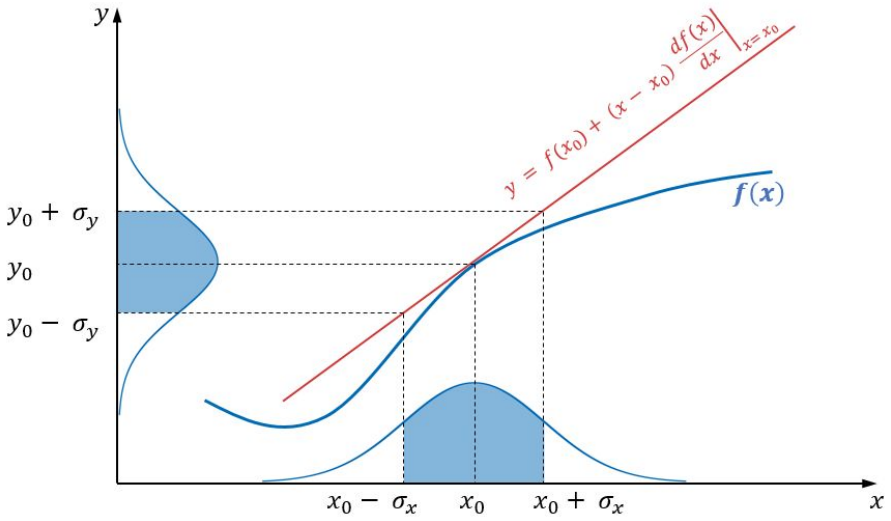
$$\bar{\mu}_t = A_t\ \mu_{t-1} + B_t\ u_t$$
$$\bar{\Sigma}_t = A_t\ \Sigma_{t-1}\ A_t^T + R_t$$
$$K_t = \bar{\Sigma}_t\ C_t^T (C_t\ \bar{\Sigma}_t\ C_t^T + Q_t)^{-1}$$
$$\mu_t = \bar{\mu}_t + K_t(z_t - C_t\ \bar{\mu}_t)$$
$$\Sigma_t = (I - K_t\ C_t)\ \bar{\Sigma}_t$$
return $\mu_t, \Sigma_t$

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

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

# Extended Kalman Filter (EKF)

- Linearize at estimated state (Jacobian matrix)
- State propagation and observation model used directly
- Jacobians can be computed offline
- Covariance propagation and update based on linearized model
- Optimality of state estimation not guaranteed.

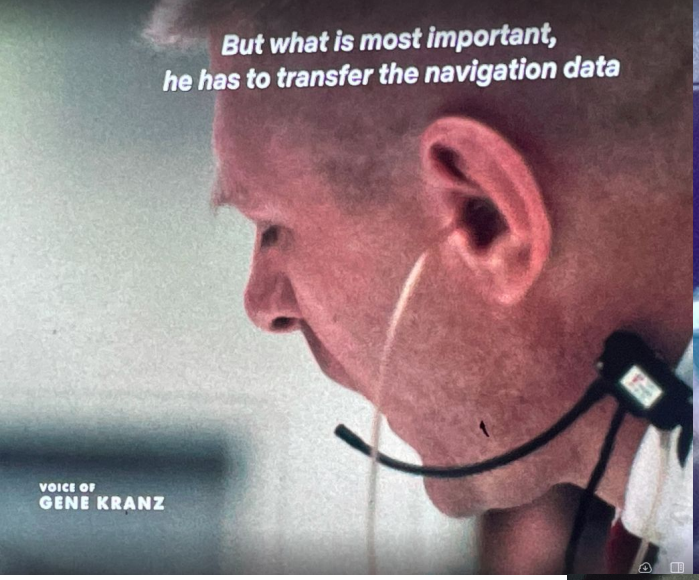



$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \cdots$$
$$\cdots + \frac{h^{(n-1)}}{(n-1)!}f^{(n-1)}(x) + \frac{h^n}{n!}f^n(x+\lambda h)$$

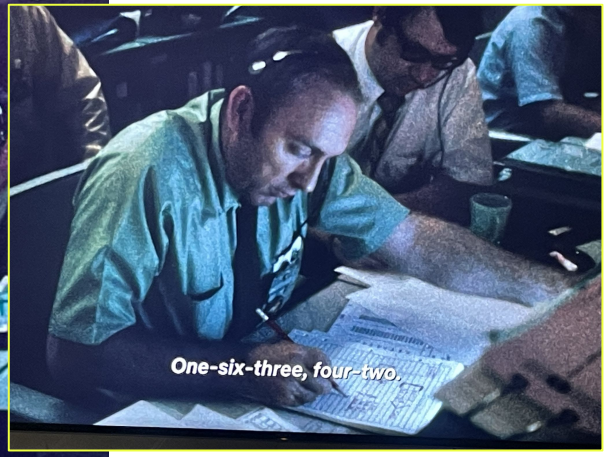
$$g'(u_t, x_{t-1}) \quad := \quad \frac{\partial g(u_t, x_{t-1})}{\partial x_{t-1}}$$

Taylor series expansion

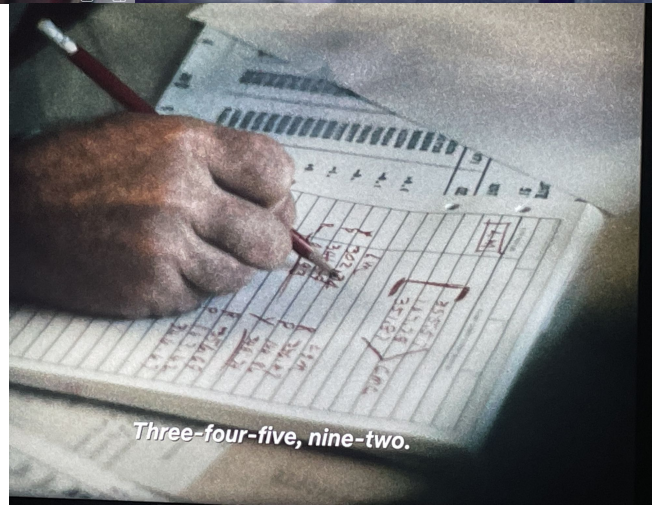
1: **Algorithm Extended_Kalman_filter**$(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t)$**:**
2: $\bar{\mu}_t = g(u_t, \mu_{t-1})$
3: $\bar{\Sigma}_t = G_t\ \Sigma_{t-1}\ G_t^T + R_t$
4: $K_t = \bar{\Sigma}_t\ H_t^T (H_t\ \bar{\Sigma}_t\ H_t^T + Q_t)^{-1}$
5: $\mu_t = \bar{\mu}_t + K_t(z_t - h(\bar{\mu}_t))$
6: $\Sigma_t = (I - K_t\ H_t)\ \bar{\Sigma}_t$
7: return $\mu_t, \Sigma_t$

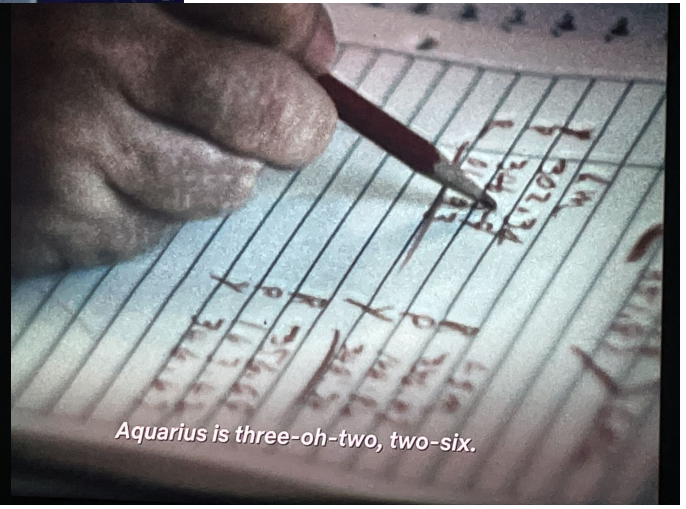




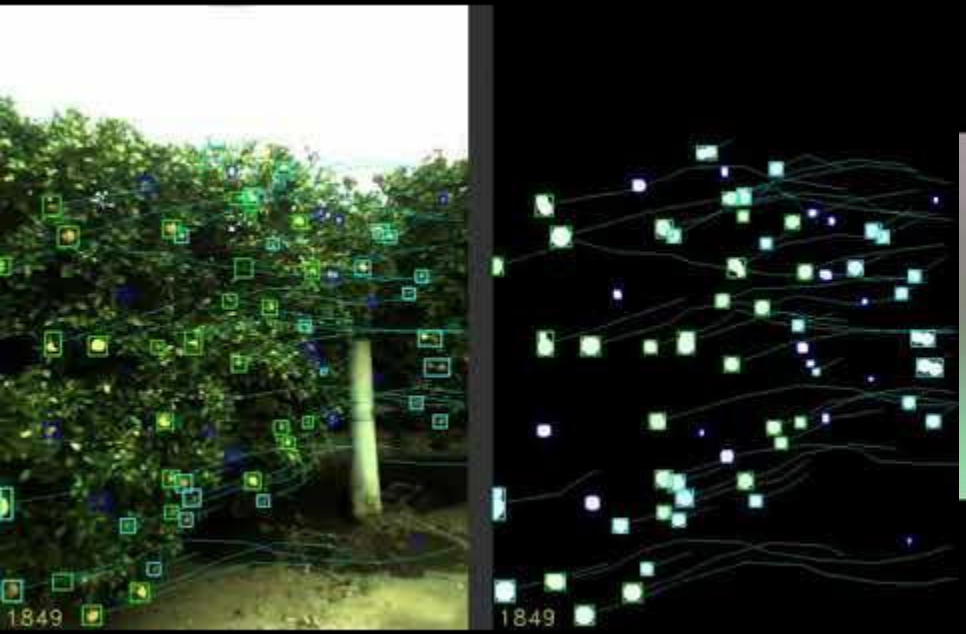


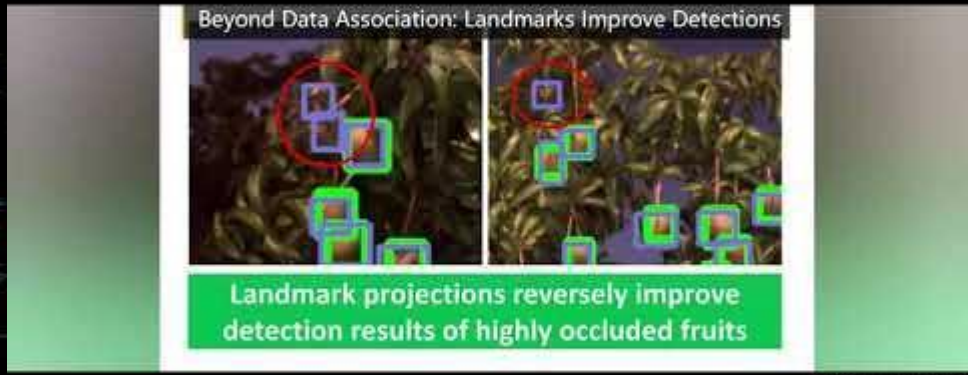

What's happening at mission control?



# Fruit counting, geometry, topology, dynamics, and metrics





- *Liu et al., "Monocular camera based fruit counting and mapping with semantic data association." IEEE Robotics and Automation Letters 4, no. 3 (2019): 2296-2303.*
- J. Das *et al.*, "Devices, systems, and methods for automated monitoring enabling precision agriculture," *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, Gothenburg, Sweden, 2015, pp. 462-469, doi: 10.1109/CoASE.2015.7294123.
- D. Orol et al., "An aerial phytobiopsy system: Design, evaluation, and lessons learned," 2017 International Conference on Unmanned Aircraft Systems (ICUAS), Miami, FL, USA, 2017, pp. 188-195.

# Desert tree dynamics

PI: Dr. Luiza Aparecido

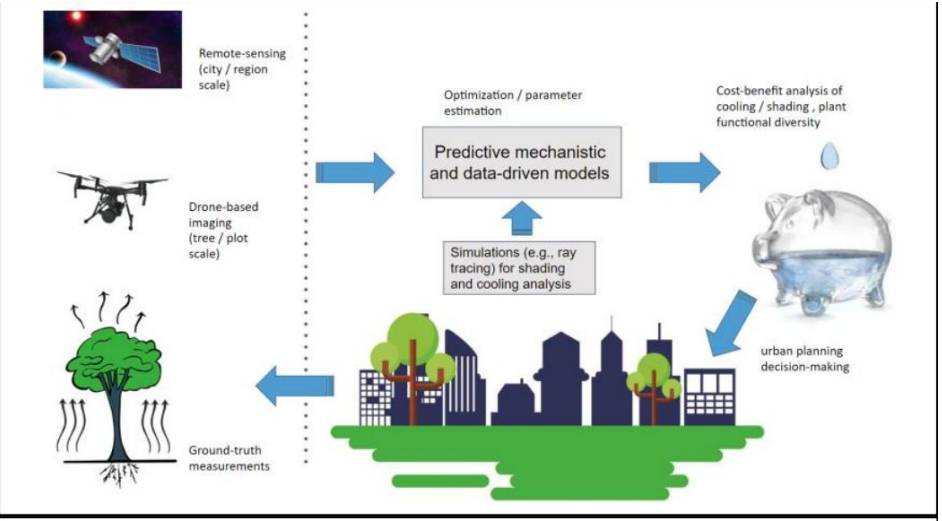
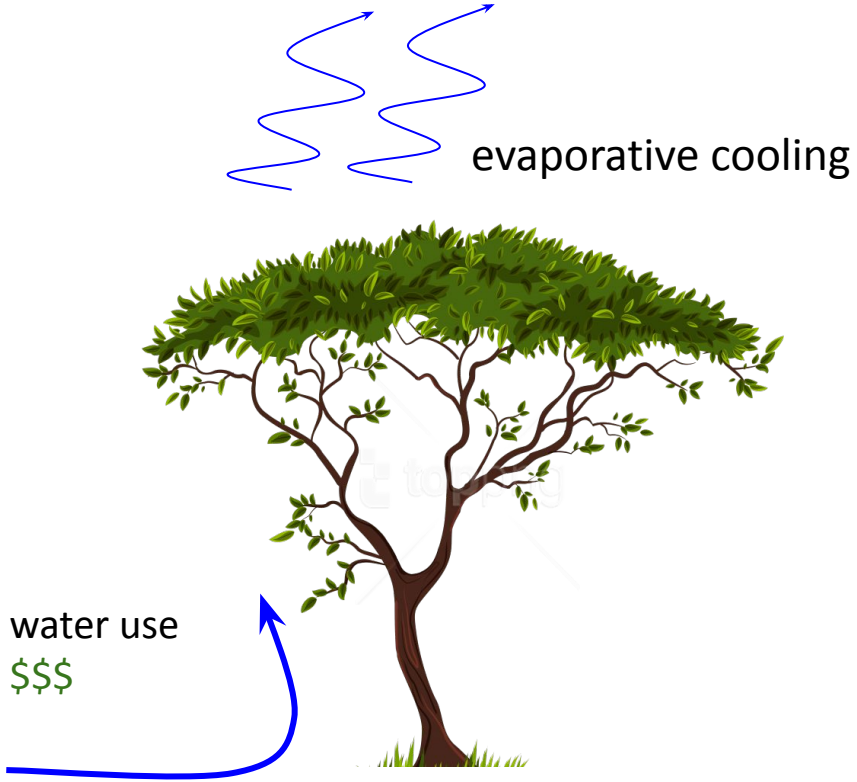


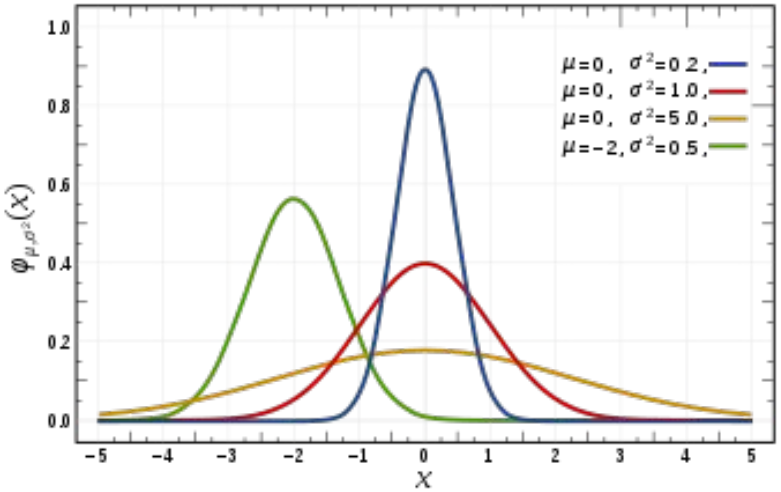

*FIGURE 1. Illustration representing the type of measurements, analyses and products expected from this proposal. In summary, we expect that through ground and aerial measurements of urban plant water use and shading we can deliver valuable information and tools to be used in urban planning decision-making.*




How do tree species differ in cooling relative to water use in urban drylands?

# Uncertainty and probability theory

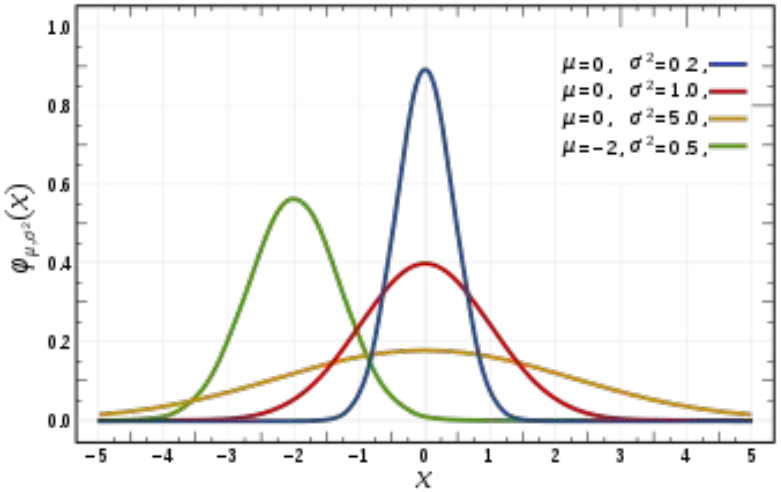
- fundamental role of uncertainty in AI
- probability theory is the calculus of reasoning with uncertainty




$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}$$

# Uncertainty and probability theory

- fundamental role of uncertainty in AI
- probability theory is the calculus of reasoning with uncertainty




$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\boldsymbol{\Sigma}|^{1/2}}\exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

# Uncertainty and probability theory

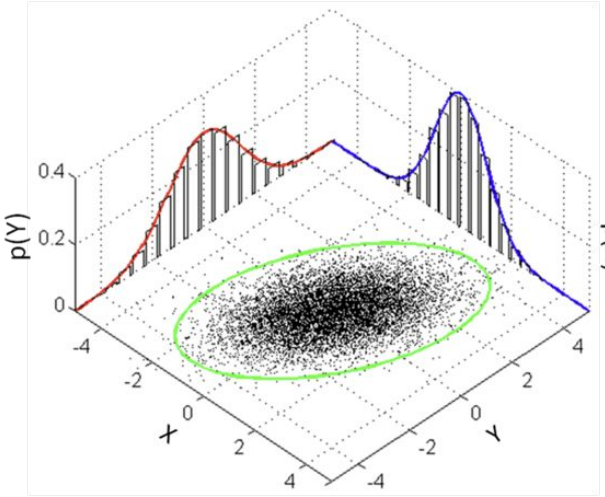
Many algorithms are designed as if knowledge is perfect, but it rarely is.

There are almost always things that are unknown, or not precisely known.

Examples:

- bus schedule
- quickest way to the airport
- sensors - joint positions

An agent making optimal decisions must take into account uncertainty




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

**sum rule** $$p(X) = \sum_{Y} p(X, Y)$$

**product rule** $$p(X, Y) = p(Y|X)p(X)$$

# Uncertainty and probability theory

Probability the precise representation of knowledge and uncertainty

Probability theory: how to optimally update your knowledge based on new information

Decision theory: probability theory + utility theory how to use this information to achieve maximum expected utility

$$\mathbb{E}[f] = \sum_x p(x) f(x)$$

“weighted average of all possible values of the variable. The weight here means the probability of the random variable taking a specific value.”

# Uncertainty in computer vision

https://docs.scipy.org/doc/numpy-1.14.0/reference/routines.random.html

Probability the precise representation of knowledge and uncertainty

Probability theory: how to optimally update your knowledge based on new information

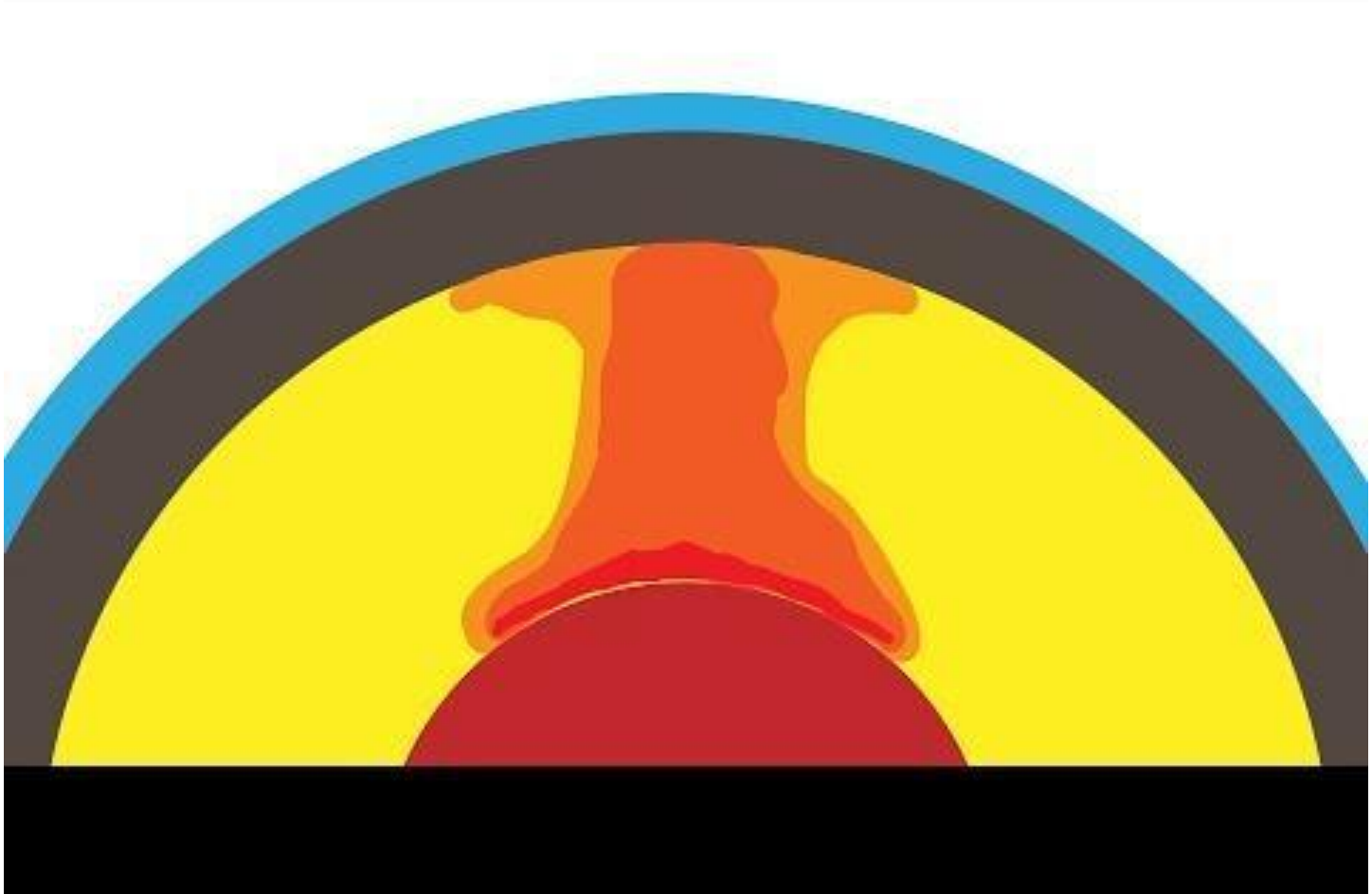
Decision theory: probability theory + utility theory how to use this information to achieve maximum expected utility

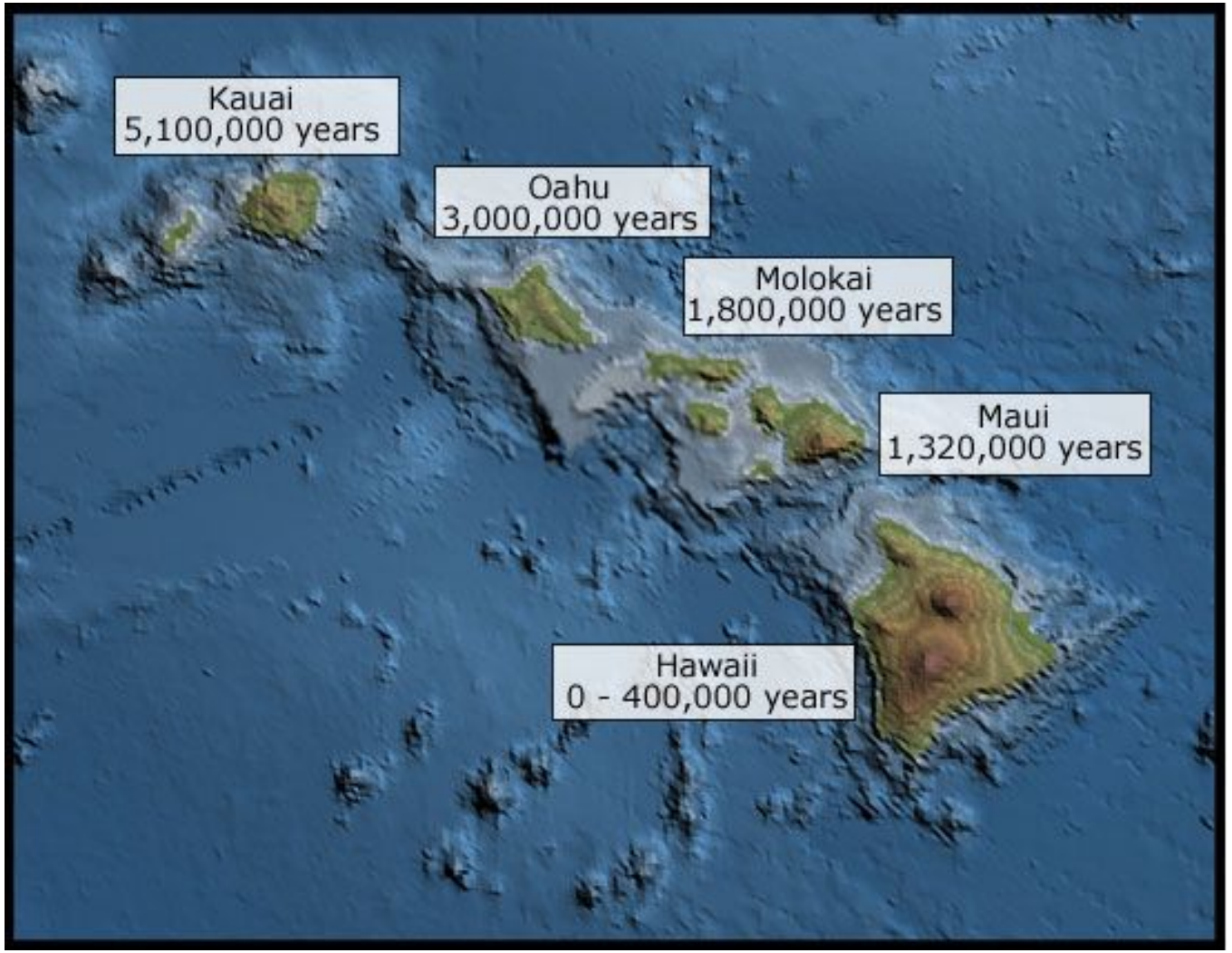
# Linear regression revisited

1) Least-squares estimation
2) Maximum likelihood estimation

The eight main islands are in order from northwest to southeast:

1. Niihau
2. Kauai
3. Oahu
4. Molokai
5. Lānai
6. Kahoolawe
7. Maui
8. Hawaii.




Kauai
5,100,000 years
Oahu
3,000,000 years
Molokai
1,800,000 years
Maui
1,320,000 years
Hawaii
0 - 400,000 years

# 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

$$y = mx + c$$

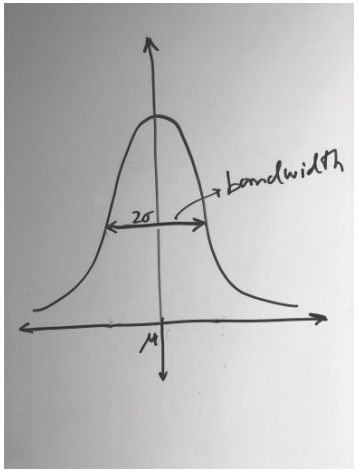
Linear model, how many parameters?

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

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

$$y = mx + c$$

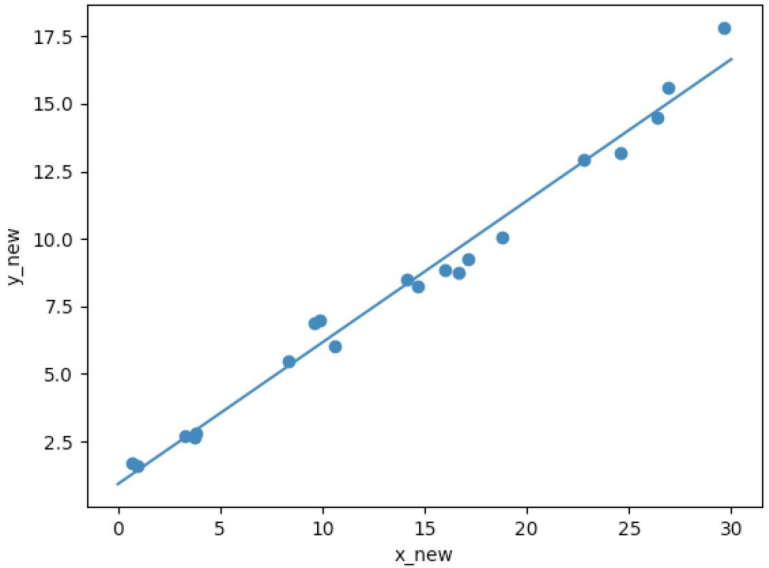
Linear model, how many parameters?

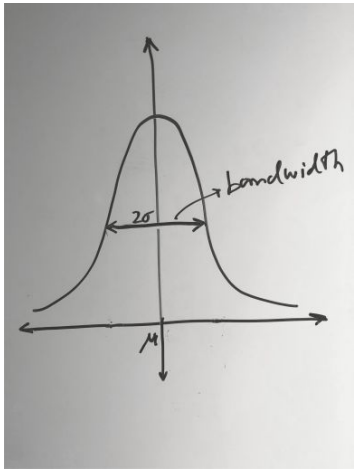
$$y = mx + c + \epsilon$$

Observations from a linear model, with noise

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

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



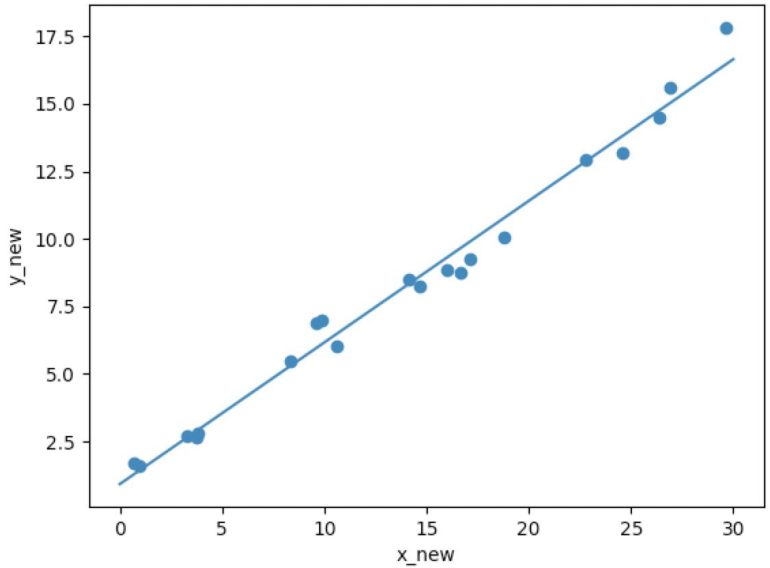
# Parameter Estimation

Estimate parameters for a linear model, given observations with noise

$$y = mx + c$$

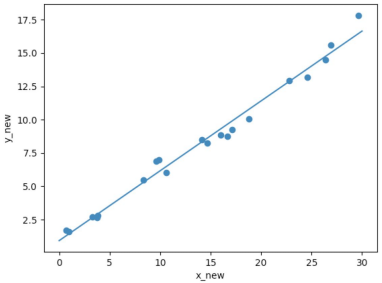
$$y = w_0 1 + w_1 x$$

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



# Parameter Estimation

Estimate parameters for a linear model, given observations with noise




$$y = mx + c$$

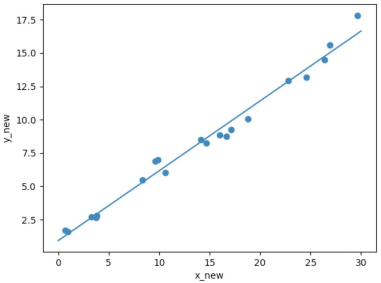
$$y = w_0 1 + w_1 x$$

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

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

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

# Parameter Estimation

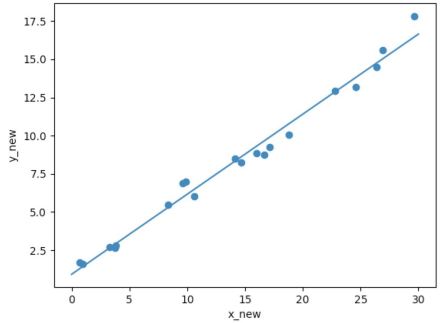



$$y = \mathbf{w}^T \mathbf{x} + \epsilon$$

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

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

# Least squares parameter estimation




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

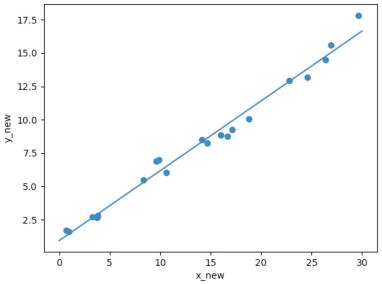
sum of square error

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

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

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

# Least squares parameter estimation




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

sum of square error

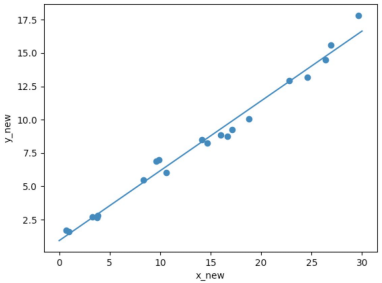
$$X = \begin{bmatrix} x_0^1 & x_1^1 \\ x_0^2 & x_1^2 \\ \vdots & \vdots \\ x_0^N & x_1^N \end{bmatrix} = \begin{bmatrix} 1 & x^1 \\ 1 & x^2 \\ \vdots & \vdots \\ 1 & x^N \end{bmatrix}$$

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

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

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

# Least squares parameter estimation




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

sum of square error

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

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

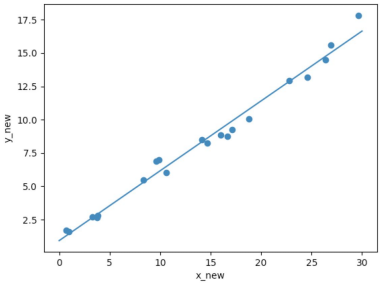
$$\arg\min_{\mathbf{w}} \sum_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2$$

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

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

Moore-Penrose pseudo inverse

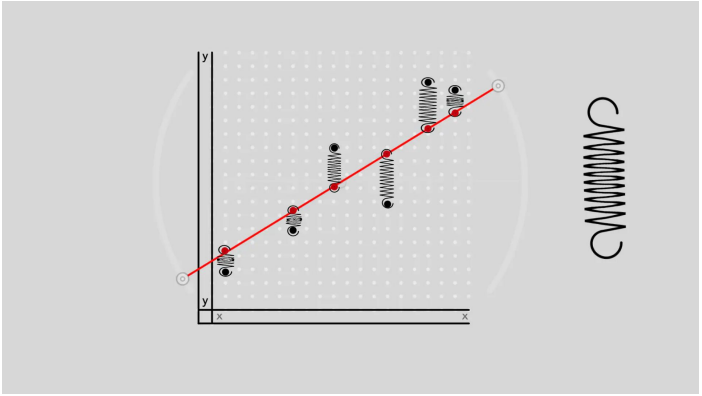
# Least squares parameter estimation




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

sum of square error

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




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

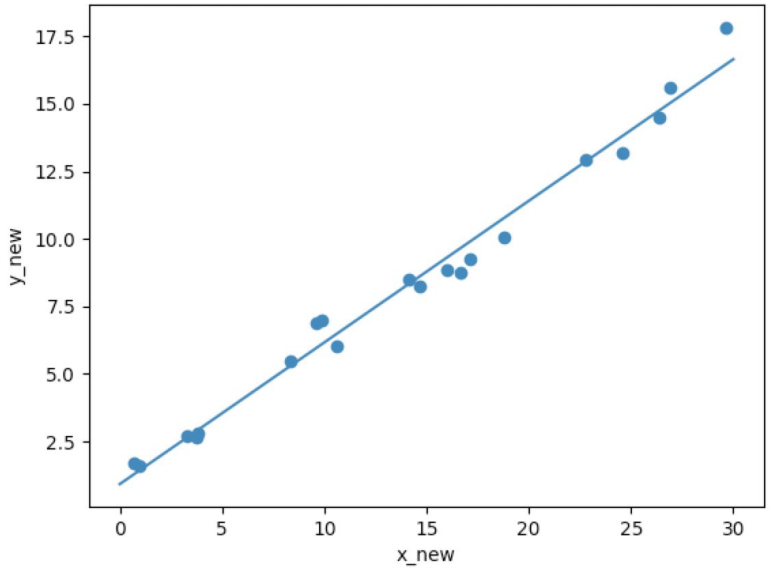
$$\mathbf{w} = (X^T X)^{-1} X^T Y$$

Moore-Penrose pseudo inverse

# Maximum Likelihood Estimation

$$\mathcal{L}(\mathbf{w}; X, Y) = \prod_{i=1}^{N} p(y^i | \mathbf{x}^i, \mathbf{w}, \sigma^2)$$

$$= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} exp^{-\frac{(y^i - \mathbf{w}^T \mathbf{x}^i)^2}{2\sigma^2}}$$

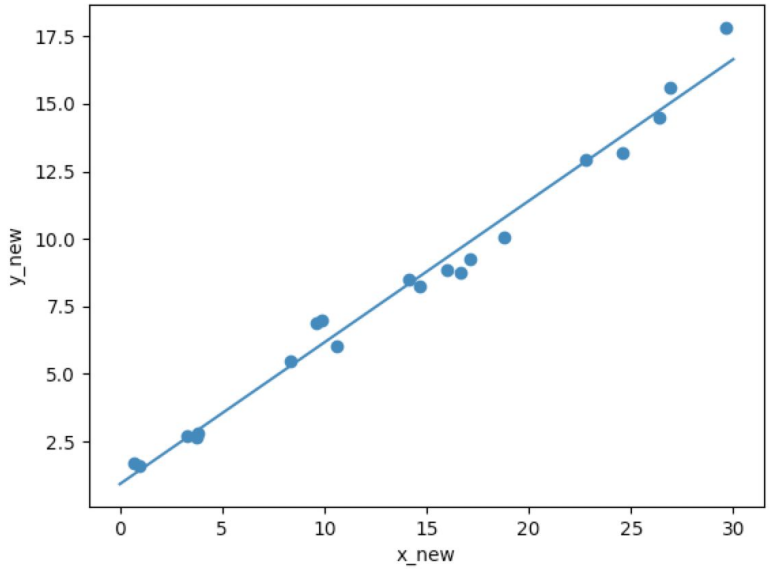


# Maximum Likelihood Estimation

**Likelihood function:**

$$\mathcal{L}(\mathbf{w}; X, Y) = \prod_{i=1}^{N} p(y^i | \mathbf{x}^i, \mathbf{w}, \sigma^2)$$

$$= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} exp^{-\frac{(y^i - \mathbf{w}^T \mathbf{x}^i)^2}{2\sigma^2}}$$



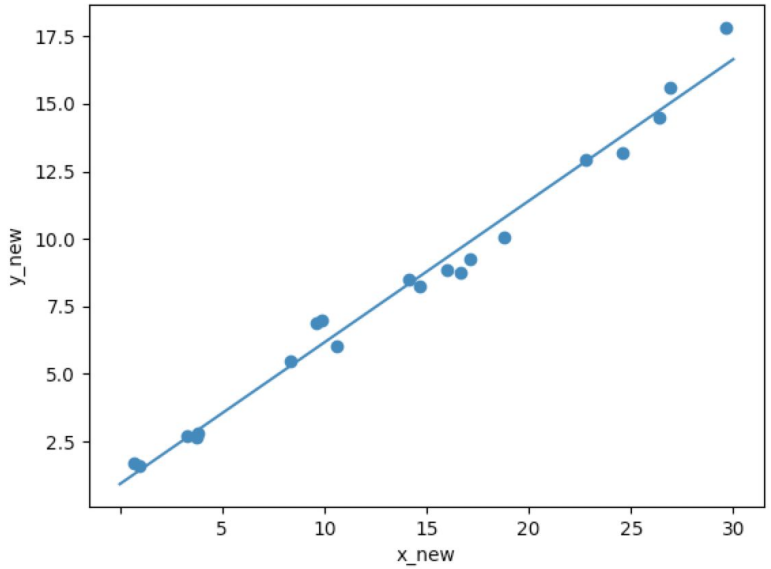

**Log-likelihood function:**

$$\mathcal{L}_{log}(\mathbf{w}; X, Y) = \ln \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} exp^{-\frac{(y^i - \mathbf{w}^T \mathbf{x}^i)^2}{2\sigma^2}}$$

# Maximum Likelihood Estimation

$$\mathcal{L}_{log}(\mathbf{w}; X, Y) = \ln \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} exp^{-\frac{(y^i - \mathbf{w}^T \mathbf{x}^i)^2}{2\sigma^2}}$$

$$\ln \text{ a.b } = \ln \text{ a } + \ln \text{ b}$$



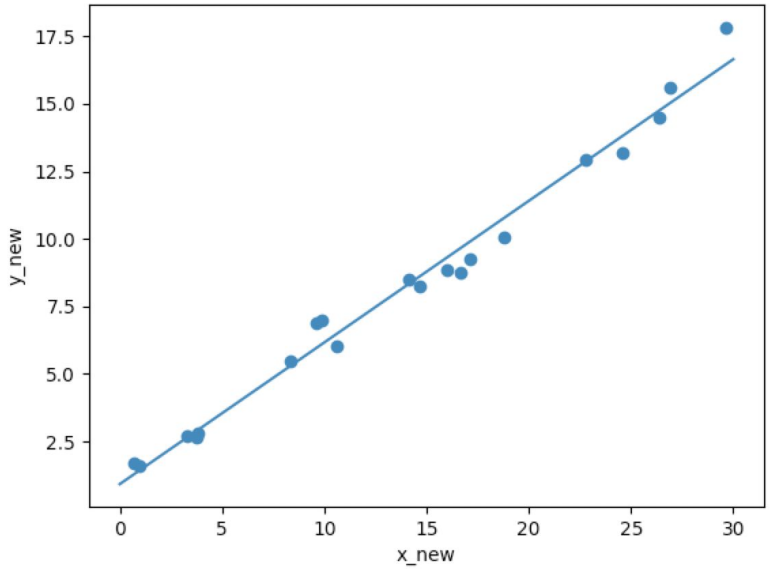

$$\mathcal{L}_{log}(\mathbf{w}; X, Y) = \sum_{i=1}^{N} \left( \ln \frac{1}{\sqrt{2\pi\sigma^2}} + \ln exp^{-\frac{(y^i - \mathbf{w}^T \mathbf{x}^i)^2}{2\sigma^2}} \right) \quad (16)$$

$$= \sum_{i=1}^{N} \ln \frac{1}{\sqrt{2\pi\sigma^2}} + \sum_{i=1}^{N} \ln exp^{-\frac{(y^i - \mathbf{w}^T \mathbf{x}^i)^2}{2\sigma^2}} ) \quad (17)$$

# Maximum Likelihood Estimation

$$\mathcal{L}_{log}(\mathbf{w}; X, Y) = \sum_{i=1}^{N} \left( \ln \frac{1}{\sqrt{2\pi\sigma^2}} + \ln exp^{-\frac{(y^i - \mathbf{w}^T \mathbf{x}^i)^2}{2\sigma^2}} \right)$$

$$= \sum_{i=1}^{N} \ln \frac{1}{\sqrt{2\pi\sigma^2}} + \sum_{i=1}^{N} \ln exp^{-\frac{(y^i - \mathbf{w}^T \mathbf{x}^i)^2}{2\sigma^2}} )$$




$$\mathcal{L}_{log}(\mathbf{w}; X, Y) = C - \frac{1}{2\sigma^2} \sum_{i=1}^{N} (y^i - \mathbf{w}^T \mathbf{x}^i)^2 \qquad (18)$$

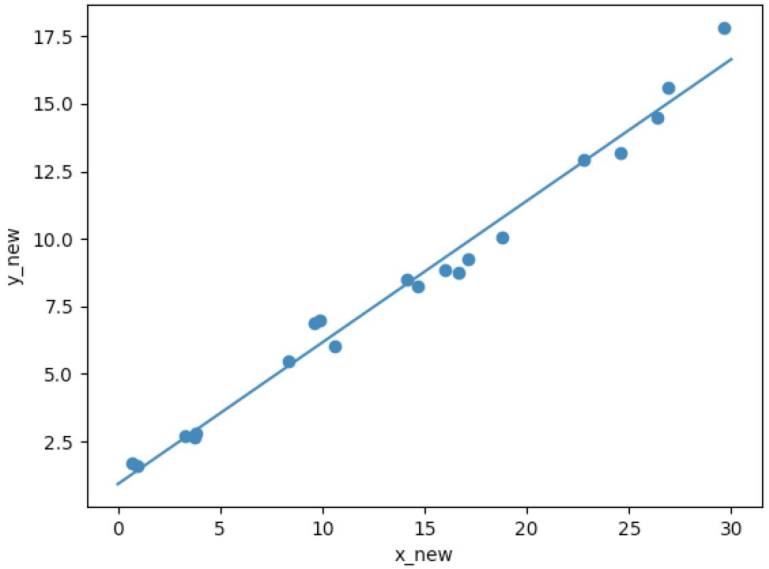
# Maximum Likelihood Estimation

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

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \quad \mathcal{L}_{log}(\mathbf{w}; X, Y)$$

**Maximum Likelihood Estimation**

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




above optimization is equivalent to minimizing the negative log likelihood

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

**Maximum Likelihood Estimation ⇔ Least Squares Estimation**

# Next class

Least squares and maximum likelihood estimation (contd..)

Linear dynamical systems …