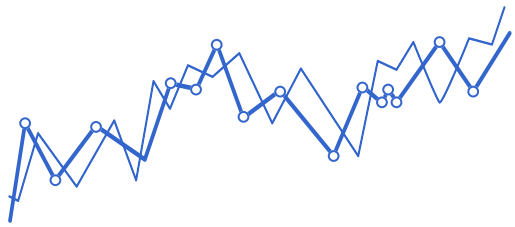
Signal Inference

$\hat{\theta}$ $\hat{\theta}$ $\hat{\theta}$ $\hat{\theta}$ $\hat{\theta}$ $\hat{\theta}$



Statistical Signal Processing

Statistical Signal Processing is an approach that models signals as stochastic processes, using statistical inference to perform estimation of unknown parameters, detection of events, and learning from data for information extraction, noise reduction, and informed decision-making.

Stochastic Signal Modeling

A random variable x maps outcomes to real values. A random vector $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ is a vector of N random variables. A stochastic process $x(t)$ is a time-indexed collection $\{x(t): t \in T\}$ of random variables. In signal processing, $x(t)$ models a random signal, while the discrete noisy measurement $\mathbf{x}$ is a finite realization (sample) of $x(t)$.

Estimation Theory

Estimation theory is a branch of statistics focused on estimating parameter values from measured data with a random component.

Let $\boldsymbol{\theta} = [\theta_0, \dots, \theta_{p-1}]^T \in \mathbb{R}^p$ be unknown true population parameters defining the probability distribution function $p(\mathbf{x}|\boldsymbol{\theta})$ over data vectors $\mathbf{x} \in \mathbb{R}^N$ — the statistical population, the distribution of possible data.

A sample $\mathbf{x} = [x[0], \dots, x[N-1]]$ is a single observed vector drawn from the statistical population, one realization of the random process.

A statistic is any function of $\mathbf{x}$; an estimator is a statistic $g(\mathbf{x})$ producing the estimate $\widehat{\boldsymbol{\theta}} = g(\mathbf{x})$.

Goal: design $g(\mathbf{x})$ so $\widehat{\boldsymbol{\theta}} \approx \boldsymbol{\theta}$.

Frequentist

Frequentist Estimation interprets probability as long-run frequency, treating the parameters $\boldsymbol{\theta}$ as deterministic variables and using $\mathbf{x}$ to construct estimators $\widehat{\boldsymbol{\theta}} = g(\mathbf{x})$.

Bayesian

Bayesian Estimation interprets probability as a degree of belief, treating $\boldsymbol{\theta}$ as a random vector with prior $\pi(\boldsymbol{\theta})$ and uses the posterior $p(\boldsymbol{\theta}|\mathbf{x})$ for estimators.

Basic Statistical Quantities

Let $\mathbf{x} = (x[n])_{n=0}^{N-1} \in \mathbb{R}^N$ and $\mathbf{y} = (y[m])_{m=0}^{M-1} \in \mathbb{R}^M$ be real random vectors, and $x(t), y(t)$ be scalar stochastic processes indexed by t.

mean $$\boldsymbol{\mu}_{\mathbf{x}} = E[\mathbf{x}] = \int_{\mathbb{R}^N} \mathbf{x}\, p(\mathbf{x}) d\mathbf{x} \in \mathbb{R}^N$$

variance $$\sigma_x^2 = \mathrm{var}(x) = \mathrm{cov}(x, x) = E[(x - \mu_x)^2] = E[x^2] - (E[x])^2 \in \mathbb{R}$$

cross-covariance

$$\mathbf{C}_{\mathbf{xy}} = \mathrm{cov}(\mathbf{x}, \mathbf{y}) = E\left[(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}})(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}})^T\right] = E[\mathbf{x}\mathbf{y}^T] - \boldsymbol{\mu}_{\mathbf{x}}\boldsymbol{\mu}_{\mathbf{y}}^T \in \mathbb{R}^{N \times M}$$

$$\mathrm{cov}\big(x(t), y(t+\tau)\big) = E\left[\big(x(t) - \mu_x(t)\big)\Big(y(t+\tau) - \mu_y(t+\tau)\Big)\right] \in \mathbb{R}$$

x and y are uncorrelated iff $\mathrm{cov}(x, y) = 0 \Leftrightarrow E[xy] = E[x]E[y]$

cross-correlation

$$\mathbf{p}_{\mathbf{xy}} = E[\mathbf{x}\mathbf{y}^T] \in \mathbb{R}^{N \times M}, \qquad p_{xy}(\tau) = E[x(t)y(t+\tau)] \in \mathbb{R}$$

auto-covariance

$$\mathbf{C}_{\mathbf{x}} = \mathrm{cov}(\mathbf{x}, \mathbf{x}) = E[(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}})(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}})^T] = E[\mathbf{x}\mathbf{x}^T] - \boldsymbol{\mu}_{\mathbf{x}}\boldsymbol{\mu}_{\mathbf{x}}^T \in \mathbb{R}^{N \times N}$$

$$\mathrm{cov}\big(x(t), x(t+\tau)\big) = E\big[\big(x(t) - \mu_x(t)\big)\big(x(t+\tau) - \mu_x(t+\tau)\big)\big] \in \mathbb{R}$$

auto-correlation

$$\mathbf{R}_{\mathbf{xx}} = E[\mathbf{x}\mathbf{x}^T] \in \mathbb{R}^{N \times N}, \qquad R_{xx}(\tau) = E[x(t)x(t+\tau)] \in \mathbb{R}$$

* Under ergodicity and wide-sense stationarity, a single observation from the stochastic process can be used to estimate population parameters.

Wide Sense Stationarity (WSS)

A stationary process is a stochastic process whose statistical properties, such as mean and variance, do not change over time. A stochastic process $x(t)$ is wide sense stationary if:

(1) Its mean is constant: $$\mu_x(t) = E[x(t)] = \mu_x, \qquad \forall t$$

(2) Its autocorrelation depends only on the lag τ: $$R_{xx}(t, t+\tau) = E[x(t)x(t+\tau)] = R_{xx}(\tau)$$

Ergodicity

An ergodic stochastic process is a process where the time average of a random variable over a long period converges to the ensemble average, which is the average over all possible realizations of the process.

Independent and Identically Distributed (i.i.d.)

A sequence of random variables $\{x[n]\}$ is i.i.d. if each $x[n]$ has the same probability distribution and all $x[n]$ are mutually independent. This implies the joint probability factorizes as:

for independent random variables $$p(\mathbf{x}) = \prod_{n=0}^{N-1} p(x[n])$$

* Two indepndent random variables are always uncorrelated, but uncorrelation only implies independence for Gaussian (normal distribution) random variables.

Multivariate Gaussian Distribution

A random vector $\mathbf{x} \in \mathbb{R}^N$ follows a multivariate normal (Gaussian) distribution with mean $\boldsymbol{\mu} \in \mathbb{R}^N$ and covariance matrix $\mathbf{C} \in \mathbb{R}^{N\times N}$, which a symmetric positive definite matrix, denoted

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{C})$$

if its probability density function is

$$p(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n |\mathbf{C}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \mathbf{C}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

* Noise is usually modeled as Gaussian because, by the Central Limit Theorem, sums of many independent disturbances approximate a Gaussian distribution.

Classical (Frequentist) Signal Model

A common model in classical (frequentist) estimation assumes:

$$x[n] = s[n;\boldsymbol{\theta}] + w[n], \qquad n = 0, \dots, N-1$$

- $x[n]$: the n-th sample of the observed noisy measurement vector $\mathbf{x}$
- $s[n;\boldsymbol{\theta}]$: known deterministic signal form depending on $\boldsymbol{\theta}$
- $w[n]$: the n-th sample of the additive noise vector $\mathbf{w} \in \mathbb{R}^N$

White Gaussian Noise

A zero-mean Gaussian vector $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$ with covariance matrix $\sigma^2\mathbf{I}$. Entries $w[n]$ are i.i.d.

Colored Gaussian Noise

A zero-mean Gaussian vector $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$ with covariance matrix $\mathbf{C}$ and correlated entries.

* Estimators are random variables themselves, so we can compute their expected value and variance to analyze their performance.

Bias

The bias of an estimator measures the difference between its expected value and the true parameter.

$$b(\hat{\theta}) = E(\hat{\theta}) - \theta$$

Mean Squared Error (MSE)

Mean Squared Error is the expected squared difference between the estimator and the true parameter. It can be decomposed into the sum of the variance of the estimator and the square of its bias:

$$\mathrm{MSE}(\hat{\theta}) = E\left[(\theta - \hat{\theta})^2\right] = \mathrm{var}(\hat{\theta}) + b^2(\hat{\theta})$$

* An unbiased estimator is an estimator with zero bias, one whose expected value is equal to the true value of the population parameter being estimated.

Minimum Variance Unbiased Estimator (MVUE)

An estimator is called the Minimum Variance Unbiased Estimator if it is unbiased and has the smallest variance among all unbiased estimators. In other words, no other unbiased estimator has a lower variance.

Likelihood Function

The likelihood function is a function of the unknown parameter that assigns to each parameter value the probability or probability density of the observed data, treating the data as fixed.

$$\theta \longmapsto \mathcal{L}(\theta; \mathbf{x}) = p(\mathbf{x}|\theta)$$

Fisher Information

Fisher information quantifies how much information an observable random vector carries about an unknown parameter in a probability model. It measures how sharply the log-likelihood function curves around the true parameter value: a sharper curve implies more information and lower variance for unbiased estimators. Formally, Fisher information is defined as the variance of the score function, or equivalently, the expectation of the observed information. Under regularity conditions:

$$I(\theta) = \mathrm{var}\left(\frac{\partial}{\partial\theta}\ln\mathcal{L}(\theta;\mathbf{x})\right) = E\left[\left(\frac{\partial}{\partial\theta}\ln\mathcal{L}(\theta;\mathbf{x})\right)^2\right]$$

variance of the score function

$$I(\theta) = -E\left[\frac{\partial^2}{\partial\theta^2}\ln\mathcal{L}(\theta;\mathbf{x})\right] \qquad E\left[\frac{\partial}{\partial\theta}\ln\mathcal{L}(\theta;\mathbf{x})\right] = 0$$

expectation of the observed information

Cramér-Rao Lower Bound (CRLB)

The CRLB gives a lower bound on the variance of any unbiased estimator of a parameter. It shows that estimator variance is at least the inverse of the Fisher information.

$$\mathrm{var}(\hat{\theta}) \geq \frac{1}{I(\theta)} \qquad \mathrm{cov}(\hat{\boldsymbol{\theta}}) \succcurlyeq \mathbf{I}(\boldsymbol{\theta})^{-1}$$

Iff the score function can be written as:

$$\frac{\partial}{\partial\theta}\ln\mathcal{L}(\theta;\mathbf{x}) = I(\theta)(g(\mathbf{x})-\theta) \qquad \frac{\partial}{\partial\boldsymbol{\theta}}\ln\mathcal{L}(\boldsymbol{\theta};\mathbf{x}) = \mathbf{I}(\boldsymbol{\theta})(\hat{\boldsymbol{\theta}}-\boldsymbol{\theta})$$

then $\hat{\theta} = g(\mathbf{x})$ achieves the bound and is the MVUE.

Efficient Estimator

An efficient estimator minimizes a chosen loss function, a function that measures how undesirable estimation errors are based on their size, typically the mean squared error. Specifically, an unbiased estimator is efficient if its variance attains the Cramér-Rao Lower Bound, which corresponds to achieving the minimum possible mean squared error among all unbiased estimators under this criterion. An efficient estimator need not exist, but if it does, it is the MVUE.

CRLB for Signals in White Gaussian Noise

$$\mathbf{x}\sim\mathcal{N}(\mathbf{s}(\theta),\sigma^2\mathbf{I}) \quad \mathbf{x} = [x[0],\dots,x[N-1]]^T \quad \mathbf{s}(\theta) = [s[0;\theta],\dots,s[N-1;\theta]]^T$$

$$\ln\mathcal{L}(\theta;\mathbf{x}) = -\frac{N}{2}\ln\mathcal{L}(2\pi\sigma^2) - \frac{1}{2\sigma^2}\|\mathbf{x}-\mathbf{s}(\theta)\|^2 \qquad \text{log-likelihood}$$

$$\frac{\partial}{\partial\theta}\ln\mathcal{L}(\theta;\mathbf{x}) = \frac{1}{\sigma^2}(\mathbf{x}-\mathbf{s}(\theta))^T\frac{\partial\mathbf{s}(\theta)}{\partial\theta} \qquad \text{score function}$$

$$f(\theta) = h(\mathbf{s}(\theta)) \qquad \frac{\partial f}{\partial\theta} = \left(\frac{\partial h}{\partial\mathbf{s}}\right)^T\frac{\partial\mathbf{s}}{\partial\theta}$$

$$\frac{\partial^2}{\partial\theta^2}\ln\mathcal{L}(\theta;\mathbf{x}) = \frac{1}{\sigma^2}\left[\left\|\frac{\partial\mathbf{s}(\theta)}{\partial\theta}\right\|^2 - (\mathbf{x}-\mathbf{s})^T\frac{\partial^2\mathbf{s}}{\partial\theta^2}\right] \qquad \text{observed information}$$

$$I(\theta) = E\left[-\frac{\partial^2}{\partial\theta^2}\ln\mathcal{L}(\theta;\mathbf{x})\right] = \frac{1}{\sigma^2}\left\|\frac{\partial\mathbf{s}(\theta)}{\partial\theta}\right\|^2 \Rightarrow \mathrm{var}(\hat{\theta}) \geq \frac{1}{I(\theta)} \qquad \text{fisher information}$$

$$\mathrm{var}(\hat{\theta}) \geq \frac{\sigma^2}{\left\|\frac{\partial\mathbf{s}(\theta)}{\partial\theta}\right\|^2} = \frac{\sigma^2}{\sum_{n=0}^{N-1}\left(\frac{\partial s[n;\theta]}{\partial\theta}\right)^2}$$

General CRLB for Signals in White Gaussian Noise

BLUE (Best Linear Unbiased Estimator)

Consider a linear model in the unknown parameter vector, with colored Gaussian noise with known covariance. We want the best linear unbiased estimator of the parameter. The Gauss-Markov theorem guarantees that this estimator, called the BLUE, exists and is unique. It also achieves the Cramér-Rao Lower Bound when the noise is Gaussian.

$$\mathbf{x} = \mathbf{H}\boldsymbol{\theta} + \mathbf{w}, \quad \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$$

$$\frac{\partial}{\partial \boldsymbol{\theta}} \ln \mathcal{L}(\boldsymbol{\theta}; \mathbf{x}) = \mathbf{H}^T \mathbf{C}^{-1} (\mathbf{x} - \mathbf{H}\boldsymbol{\theta})$$

$$-\frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \ln \mathcal{L}(\boldsymbol{\theta}; \mathbf{x}) = \mathbf{H}^T \mathbf{C}^{-1} \mathbf{H}$$

$$\operatorname{cov}(\widehat{\boldsymbol{\theta}}) = (\mathbf{H}^T \mathbf{C}^{-1} \mathbf{H})^{-1}$$

$$\widehat{\boldsymbol{\theta}} \sim \mathcal{N}(\boldsymbol{\theta}, (\mathbf{H}^T \mathbf{C}^{-1} \mathbf{H})^{-1})$$

$$\widehat{\boldsymbol{\theta}} = (\mathbf{H}^T \mathbf{C}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{C}^{-1} \mathbf{x}$$

Maximum Likelihood Estimator (MLE)

The Maximum Likelihood Estimator maximizes the likelihood function of the parameters given the observed data:

$$\widehat{\boldsymbol{\theta}}_{\text{MLE}} = \operatorname*{argmax}_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; \mathbf{x})$$

Asymptotically, as the number of samples $N \longrightarrow \infty$:

(1) Asymptotically Unbiased $\quad E[\widehat{\boldsymbol{\theta}}_{\text{MLE}}] \to \boldsymbol{\theta}$

(2) Asymptotically Efficient $\quad \operatorname{cov}(\widehat{\boldsymbol{\theta}}_{\text{MLE}}) \to \mathbf{I}(\boldsymbol{\theta})^{-1}$

(3) Asymptotically Gaussian $\quad \widehat{\boldsymbol{\theta}}_{\text{MLE}} \to \widehat{\boldsymbol{\theta}}_{\text{MLE}} \sim \mathcal{N}(\boldsymbol{\theta}, \mathbf{I}(\boldsymbol{\theta})^{-1})$

The MLE is functionally invariant under reparameterization:

$$\boldsymbol{\alpha} = g(\boldsymbol{\theta}) \Longrightarrow \widehat{\boldsymbol{\alpha}}_{\text{MLE}} = g(\widehat{\boldsymbol{\theta}}_{\text{MLE}})$$

* For linear models with guassian noise, the MLE coincides with the BLUE.

Least Squares

When the additive noise is modeled as a deterministic error, Least Squares estimates the parameter vector by minimizing the sum of squared unknown deterministic errors:

$$\widehat{\boldsymbol{\theta}}_{\text{LS}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \|\boldsymbol{\epsilon}\|^2$$

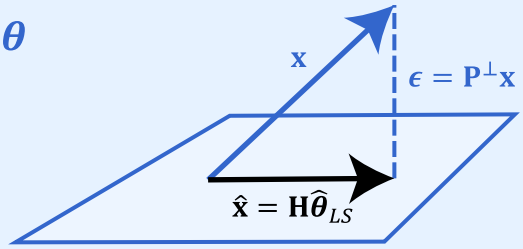
where the error is defined as: $\boldsymbol{\epsilon} = \mathbf{x} - \mathbf{s}(\theta)$

Ordinary Least Squares is a special case of Least Squares where the model is linear in the parameters: $\boldsymbol{\epsilon} = \mathbf{x} - \mathbf{H}\boldsymbol{\theta}$

Ordinary Least Squares (OLS)

To solve the linear least squares problem, we project the observed vector onto the column space of the model matrix. This projection gives the closest point in the subspace spanned by the model. The projection matrix is symmetric and idempotent, and its complement projects onto the orthogonal subspace. The residual (error) is orthogonal to the column space and lies entirely in the complement subspace.

$$\widehat{\boldsymbol{\theta}}_{\mathrm{LS}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \|\mathbf{x} - \mathbf{H}\boldsymbol{\theta}\|^2 \qquad \boldsymbol{\epsilon} = \mathbf{x} - \mathbf{H}\boldsymbol{\theta}$$



$$\mathbf{H}^T\boldsymbol{\epsilon} = \mathbf{H}^T(\mathbf{x} - \mathbf{H}\boldsymbol{\theta}) = \mathbf{0}$$

$$\widehat{\boldsymbol{\theta}}_{\mathrm{LS}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{x} \qquad \hat{\mathbf{x}} = \mathbf{H}\widehat{\boldsymbol{\theta}}_{LS} = \mathbf{P}\mathbf{x}$$

$$\mathbf{P} = \mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T \qquad \mathbf{x} = \mathbf{P}\mathbf{x} + \mathbf{P}^{\perp}\mathbf{x} \qquad \boldsymbol{\epsilon} = \mathbf{P}^{\perp}\mathbf{x} = (\mathbf{I} - \mathbf{P})\mathbf{x}$$

$$J(\boldsymbol{\theta}) = \|\boldsymbol{\epsilon}\|^2 = \boldsymbol{\epsilon}^T\boldsymbol{\epsilon} \qquad J_{\min}(\widehat{\boldsymbol{\theta}}_{\mathrm{LS}}) = \|\mathbf{x}\|^2 - \mathbf{x}^T\mathbf{H}\widehat{\boldsymbol{\theta}}_{\mathrm{LS}}$$

Adaptive Filter

An adaptive filter is a linear digital filter whose parameters are automatically adjusted based on the input signal. This self-adjusting capability allows it to adapt to changing signal characteristics and optimize its performance over time, often by minimizing an error signal.

Recursive Least Squares (RLS)

The RLS algorithm minimizes the weighted sum of squared errors with exponential weighting. It updates adaptive filter coefficients recursively to achieve faster convergence compared to OLS.

Bayesian Mean Squared Error (BMSE)

In frequentist estimation, minimizing the mean square error (MSE) requires complete second-order statistics, which are often unknown, making the estimator unrealizable. Bayesian estimation treats the unknown parameters as a random vector with a prior distribution π(θ). The Bayesian mean square error (BMSE) is the expected squared error averaged over both the data and the prior. It does not depend on the true value of the parameter, only on the prior and likelihood:

$$\mathrm{BMSE}(\hat{\theta}) = E_{\mathbf{x},\boldsymbol{\theta}}\left[\left\|\widehat{\boldsymbol{\theta}}(\mathbf{x}) - \boldsymbol{\theta}\right\|^2\right] = \iint \left\|\widehat{\boldsymbol{\theta}}(\mathbf{x}) - \boldsymbol{\theta}\right\|^2 p(\mathbf{x}, \boldsymbol{\theta})\, d\mathbf{x}\, d\boldsymbol{\theta}$$

Wiener Filter

The Wiener filter is a linear adaptive filter that minimizes the bayesian mean squared error (BMSE) between the filter output and the desired signal. It is designed under the assumption that the input and desired signals are stationary and second-order statistics are known.

$$e(n) = d(n) - y(n) = y(n) = d(n) - \mathbf{w}^T\mathbf{u}(n)$$

$$J(\mathbf{w}) = E[e^2(n)] = E[(d(n) - \mathbf{w}^T\mathbf{u}(n))^2] = \sigma_d^2 - 2\mathbf{p}^T\mathbf{w} + \mathbf{w}^T\mathbf{R}\mathbf{w}$$

$$\nabla J(\mathbf{w}) = -2\mathbf{p} + 2\mathbf{R}\mathbf{w} = 0 \Rightarrow \mathbf{w}_{\text{opt}} = \mathbf{R}^{-1}\mathbf{p}$$

Gradient descent is used to iteratively approach the Wiener solution.

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu[\mathbf{p} - \mathbf{R}\mathbf{w}(n)] \qquad \mu < 2/\lambda_{\max}$$

To guarantee convergence, the learning rate has to be less than 2 over the the maximum eigenvalue of the auto-correlation matrix.

Note: $\mathbf{R} = E[\mathbf{u}(n)\mathbf{u}^T(n)]$ $\mathbf{p} = E[d(n)\mathbf{u}(n)]$
the auto-correlation matrix is toeplitz and symmetric.

Kalman Filter

The Kalman filter is an adaptive filter used for tracking the state of a linear dynamical system when the underlying stochastic process is not wide-sense stationary. It generalizes the Wiener filter to time-varying signals and uses a recursive Bayesian approach to estimate the hidden state. The dicrete time system is expressed by the State-space model:

$$\mathbf{x}(n+1) = \mathbf{F}\mathbf{x}(n) + \boldsymbol{v}_1(n) \qquad \text{state equation}$$

(The models how the state evolves over time with process noise.)

$$\mathbf{y}(n) = \mathbf{C}\mathbf{x}(n) + \boldsymbol{v}_2(n) \qquad \text{measurement equation}$$

(This models how noisy measurements are generated from the state.)

Unlike RLS, which is determinstic, or Wiener filters, which assumes stationarity, the Kalman filter treats the state as a random vector with a prior and adaptively tracks the optimal solution over time.

Statistical Learning Theory

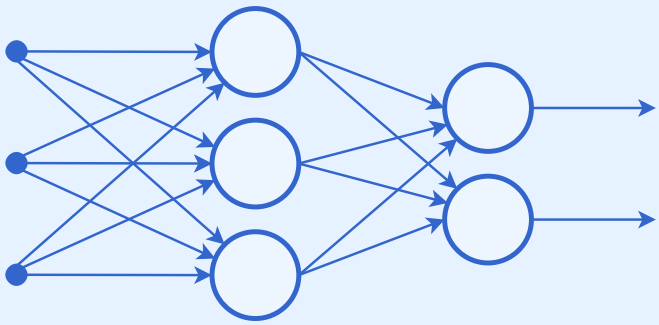
Statistical learning theory deals with the statistical inference problem of finding a predictive function based on data. Supervised learning involves learning from a training set of data. Depending on the type of output, supervised learning is to either do regression or classification. Classification assigns inputs to one of a finite set of classes.

Neural Network (Multi-Layer Perceptron)

Data is linearly separable if a linear function can perfectly divide classes, forming a flat decision boundary. When this is not the case, we use a multi-layer perceptron (MLP) to model nonlinear boundaries.

An MLP consists of an input layer, hidden layers, and an output layer. Each layer applies a linear transformation (weights and biases) followed by a nonlinear activation function (e.g., sigmoid, ReLU).

The data flows feedforward, and the output is compared to the target using a cost function. Through backpropagation and gradient descent, the network updates parameters.



The activation functions are essential for learning nonlinear patterns; otherwise, the network acts as a linear model. The Universal Approximation Theorem states that with at least one hidden layer and nonlinearity, an MLP can approximate any continuous function. For each layer $j = 2,3,\dots,L$:

input layer $\mathbf{v}_1 = \mathbf{x}$, $\mathbf{v}_j = \boldsymbol{\sigma}\left(\mathbf{W}_j \mathbf{v}_{j-1} + \mathbf{b}_j\right)$, $\mathbf{y} = \mathbf{v}_L$ output layer

$$\boldsymbol{\delta}_L = (\mathbf{d} - \mathbf{v}_L) \circ \mathbf{v}_L \circ (1 - \mathbf{v}_L) \qquad \boldsymbol{\delta}_j = (\mathbf{W}_{j+1}^{\top} \boldsymbol{\delta}_{j+1}) \circ \mathbf{v}_j \circ (1 - \mathbf{v}_j)$$

$$\mathbf{W}_j \leftarrow \mathbf{W}_j + \eta \cdot \boldsymbol{\delta}_j \cdot \mathbf{v}_{j-1}^{\top}, \qquad \mathbf{b}_j \leftarrow \mathbf{b}_j + \eta \cdot \boldsymbol{\delta}_j$$

$\circ$ - denotes elementwise multiplication (Hadamard product)

$$\boldsymbol{\sigma}(\mathbf{z}) = \frac{1}{1 + e^{-\mathbf{z}}}$$ sigmoid activaton function applied element-wise

Radial Basis Function (RBF) Network

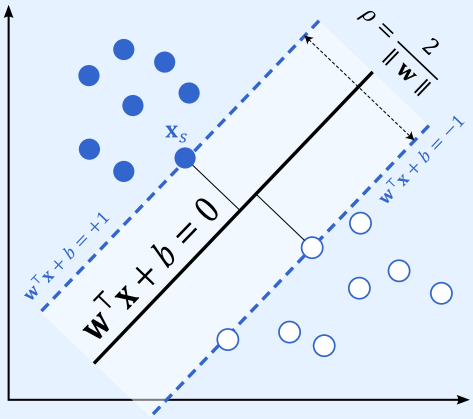
An RBF network is a feedforward neural network with an input layer, one hidden layer of K units (K < N), and an output layer. Each hidden unit applies a radial basis function, which is a function whose value depends only on the distance from some fixed point.

$$\phi(r) = exp\left(-\frac{Kr^2}{\left(\max_{i,j} \|\mathbf{x}_i - \mathbf{x}_j\|\right)^2}\right) \qquad F(\mathbf{x}_i) = \sum_{j=1}^{K} w_j \, \phi\left(\|\mathbf{x}_i - \boldsymbol{\mu}_j\|\right) = d_i$$

$$\mathbf{w} = (\boldsymbol{\Phi}^T \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^T \mathbf{d}\,, \qquad \boldsymbol{\Phi} = \left[\phi\left(\|\mathbf{x}_i - \boldsymbol{\mu}_j\|\right)\right]_{i=1,\dots,N}^{j=1,\dots,K}$$

Support Vector Machine (SVM)

A Support Vector Machine (SVM) is a supervised learning algorithm used for binary classification. It aims to find the optimal hyperplane that separates two classes with the maximum margin — the widest possible distance between the closest points of each class. It can handle both linearly and nonlinearly separable data by using kernels, and it balances accuracy and model complexity through a regularization mechanism. Support Vector Machines aim to find the best separating hyperplane between two classes by maximizing the margin — the distance between the closest points from each class. When data isn't perfectly separable, slack variables and a regularization term allow some classification errors (soft margin). The optimization problem is reformulated in its dual form using Lagrange multipliers, making it depend only on the support vectors, vectors closesest to the boundary. For nonlinear data, kernels are used to implicitly map inputs into higher-dimensional spaces without explicit transformation. The final decision function is a weighted sum over the support vectors, those with non-zero multipliers , and its sign determines the predicted class.



$$\max_{\boldsymbol{\alpha}} \quad Q(\boldsymbol{\alpha}) = \sum_{i\ \ 1}^{N} \alpha_i - \frac{1}{2} \sum_{i\ \ 1}^{N} \sum_{j\ \ 1}^{N} \alpha_i\, \alpha_j d_i d_j\, k(\mathbf{x}_i, \mathbf{x}_j)$$

$$k(\mathbf{x}_i, \mathbf{x}_j) = \boldsymbol{\phi}(\mathbf{x}_i)^\top \boldsymbol{\phi}(\mathbf{x}_j)$$

$$\mathbf{w}_o = \sum_{s=1}^{N_s} \alpha_s\, d_s\, \boldsymbol{\phi}(\mathbf{x}_s), \qquad b_o = d_s - \sum_{r=1}^{N_s} \alpha_r\, d_r\, k(\mathbf{x}_r, \mathbf{x}_s)$$

$$g(\mathbf{x}) = \sum_{s \in \mathrm{SV}} \alpha_s\, d_s\, k(\mathbf{x}_s, \mathbf{x}) + b_0 \qquad 0 \le \alpha_i \le C, \quad \sum_{i=1}^{N} \alpha_i\, d_i = 0$$

K-Means Clustering

K-means is a type of unsupervised learning that partitions data into K clusters by minimizing the within-cluster sum of squares. Algorithm:

(1) Initialize K centroids.
(2) Assign each point to the nearest centroid.
(3) Update centroids as the mean of assigned points.
(4) Repeat until convergence.

$$D = \frac{1}{N} \sum_{i=1}^{k} \sum_{\mathbf{x} \in C_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2$$

Average Distortion (mean sum of squares)

Detection Theory

Detection theory is a branch of statistical inference focused on deciding which hypothesis best explains observed data, especially when the data is noisy or uncertain. It aims to design decision rules that minimize errors or expected costs in identifying the true hypothesis.

Binary Hypothesis Testing

In binary hypothesis testing, the goal is to decide, based on an observation, which of two possible hypotheses is true. H_1 (Hypothesis 1) represents one possible state or condition, while H_0 (null Hypothesis 0) represents the other. The four probabilities associated are:

probability of detectoin

$$P(H_1|H_1) = P_D = \int_R p(\mathbf{x}|H_1)d\mathbf{x}$$

probability of false alarm

$$P(H_1|H_0) = P_{FA} = \int_R p(\mathbf{x}|H_0)d\mathbf{x}$$

probability of miss

$$P(H_0|H_1) = P_M = 1 - P_D$$

$$P(H_0|H_0) = 1 - P_{FA}$$

* The region R is the decision region where we decide in favor of hypothesis H_1 If the observed data x lies inside R, we decide H_1 and H_0 otherwise.

Neyman–Pearson Lemma

The Neyman-Pearson lemma provides a way to construct the most powerful test that maximizes the probability of detection while keeping the probability of false alarms under a fixed allowable level using the likelihood ratio test. To maximize the probability of detection:

$$\max_R P_D = \int_R p(\mathbf{x}|H_1)d\mathbf{x} \quad \text{s.t.} \quad P_{FA} = \int_R p(\mathbf{x}|H_0)d\mathbf{x} = \alpha$$

Solution: $L(\mathbf{x}) = \frac{p(\mathbf{x}|H_1)}{p(\mathbf{x}|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \lambda \qquad P_{FA} = \int_{\{\mathbf{x}:L(\mathbf{x})>\lambda\}} p(\mathbf{x}; H_0)\, d\mathbf{x} = \alpha$

Bayesian Risk and Multiple Hypothesis Testing

In multiple hypothesis testing, Bayes risk is the expected cost of decisions, combining the likelihood of each hypothesis, the decision rule's errors, and their associated costs weighted by prior probabilities. To minimize Bayes risk, we choose the hypothesis that minimizes the expected cost given the observed data.

Bayes Risk:

$$R = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} C_{ij}\, P(H_i|H_j) P(H_j)$$