ML2025

Dima Bykhovsky

March 19, 2025

Contents

Chapter 1

Introduction

1.1 Data types

Goal: Define notation of data.

1.1.1 Basic

Typically, the data types of interest are:

Numerical

Binary Used for binary information represented by $\{0, 1\}$ or $\{\text{True}, \text{False}\}$. Typically used for binary classification problems.

Integer Typically used to describe the data with limited number of possible numerical descriptors. The most popular representations used signed or unsigned numbers with 8, 16, 32 or 64 bit representation.

Real The basic numerical representation. Standard representations are 32 or 64 bit for CPU and 8(new!), 16, 32 bit for GPU.

Categorical

Nominal Variables with values selected from a group of categories, while not having any kind of natural order. Example: car type

Ordinal A categorical variable whose categories can be meaningfully ordered.
Example: age, grade of exam.

1.1.2 Signals and time-series

Two main categories:

- Discrete-time signals that are representation of physical continuous-time prototype signal. Signals typically have constant sampling frequency, and typically handled by signal processing techniques. Example: Voltage measurement is a signal and once-an-hour power-meter measurements.
- Time-series are typically derived from a time-stamped discrete-time origin in social sciences. Sometimes have an arbitrary sample times.
  Example: economical parameters.

1.1.3 Dataset

The basic dataset includes matrix $\mathbf{X} \in \mathscr{R}^{M \times N}$ (M rows and N columns) and vector $\mathbf{y} \in \mathscr{R}^{M}$.
A single dataset entry is a vector

$$\mathbf{x}_i^T = \begin{bmatrix} x_{i1} & x_{i2} & \cdots & x_{iN} \end{bmatrix}, \tag{1.1}$$

where i is the number of *features* (raw) and N if the dimension (number of columns), $\mathbf{x}_i \in \mathscr{R}^N$ and $x_{ij} \in \mathscr{R}$. All the values of M entries are organized in a matrix form,

$$\mathbf{X} = \begin{bmatrix} — & \mathbf{x}_1^T & — \\ — & \mathbf{x}_2^T & — \\ — & \vdots & — \\ — & \mathbf{x}_M^T & — \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1N} \\ x_{21} & x_{22} & \cdots & x_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ x_{M1} & x_{M2} & \cdots & x_{MN} \end{bmatrix} \tag{1.2}$$

1.2 Tasks

Typical related task:

- Prediction or regression, $\mathbf{y}$ is quantitative (Fig. 1.1).
- Classification, $\mathbf{y}$ is categorical.
- Segmentation
- Anomaly detection
- Simulation
- Clustering, no $\mathbf{y}$ is provided - it is learned from dataset.
- Signal processing tasks: noise removal, smoothing (filling missing values), event/condition detection.

1.3 Basic workflow

The basic ML/DL workflow is presented in Fig. 1.2. The workflow parts are:

- Data: available data
- Model: basic assumptions about the hidden pattern within the data
- Model training: minimization of the loss functions to derive the most appropriate parameters.
- Performance assessment according predefined metrics.

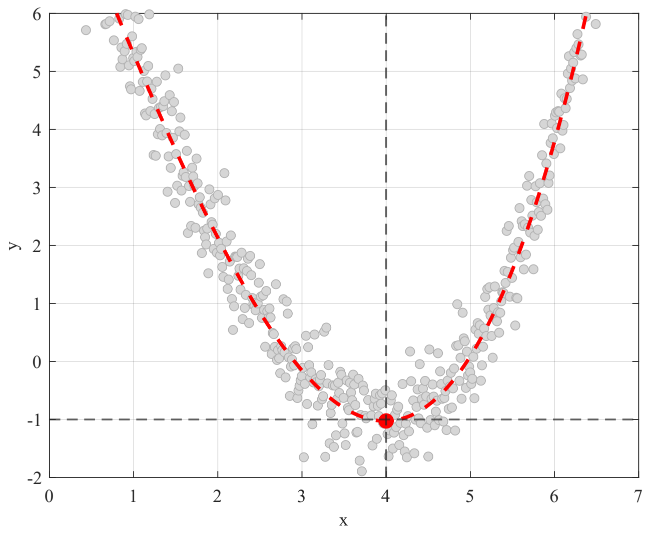


Figure 1.1: Regression example: what is the value of y for given x?

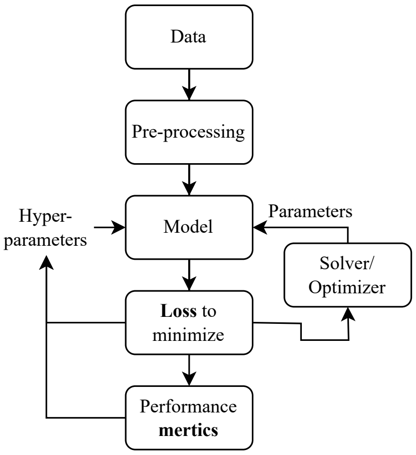


Figure 1.2: Workflow

Baseline The basic end-to-end workflow implementation is called baseline.

1.4 Model

We assume that there is an underling problem (e.g., regression and classification) formulation is of the form

$$y = f(\mathbf{x}) + \epsilon \tag{1.3}$$

where the values of $\mathbf{x}$ (scalar or vector) and y are known (it is the dataset) and ϵ is some irreducible noise. Sometimes, zero-mean noise is assumed.
The goal is to find the function $f(\cdot)$. The way to define the $f(\cdot;\mathbf{w})$ is termed *model* that depends on some model parameters vector $\mathbf{w}$. The process of finding regression solution by a set of parameters $\mathbf{w}$ is called learning, such as the resulting model can provide output

$$\hat{y}_0 = f(\mathbf{x}_0;\mathbf{w}) \tag{1.4}$$

for some new data $\mathbf{x}_0$.

Parameters vs hyper-parameters

Model parameters: Model parameters are learned directly from a dataset.
Hyper-parameters: Model parameters that are not learned directly from a dataset are called **hyper-parameters**. They are learned in in-direct way during cross-validation process in the follow.

Parametric vs non-parametric models

There are two main classes of models: parametric and non-parametric, summarized in Table 1.1.

1.5 Loss Function

The parameters $\mathbf{w}$ are minimum of some function, that is termed loss or cost function.

1.5.1 Loss Function Minimization

Goal: Minimum of the loss function for a given model.

Closed-form solution A closed-form solution for $\mathbf{w}$ is a solution that is based on basic mathematical functions. For example, a "normal equation" is a solution for linear regression/classification.

Local-minimum gradient-based iterative algorithms This family of algorithms is applicable only for convex (preferably strictly convex) loss functions. For example, gradient descent (GD) and its modifications (e.g., stochastic GD) are used to evaluate NN parameters. Another example is the Newton-Raphson algorithm.

- Some advanced algorithms under this category also employ (require) second-order derivative $\frac{\partial^2}{\partial \mathbf{w}}\mathscr{L}$ for faster convergence.
- If either derivative is not available as a closed-form expression, it is evaluated numerically.

Global optimizers The goal of global optimizers is to find a global minimum of non-convex function. These algorithms may be gradient-free, first-derivative or second-derivative. The complexity of these algorithms is significantly higher than the local optimizer and can be prohibitive for more than a few hundred variables in $\mathbf{X}$.

Table 1.1: Comparison of parametric and non-parametric models.

Aspect	Parametric	Non-parametric
Dependence on number of parameters on dataset size	Fixed	Flexible
Interpretability	Yes	No
Underlying data assumptions	Yes	No
Risk	Underfitting due to rigid structure	Overfitting due to high flexibility
Dataset size	Smaller	Best for larger
Complexity	Often fast	Often complex
Examples	Linear regression	k-NN, trees

1.6 Metrics

Metrics are performance indicators of the model. Sometimes, the minimum of the loss function is a metric, e.g. mean squared error (MSE).

Chapter 2

Least-squares and Linear Regression

Goal:
- The goal of the least squares (LS) method is to minimize MSE (or RMSE) between the given data and the parametric model.
- Define and analyze a model that is based on a linear relation between data and the outcome.
- Find the linear model parameters by LS.

2.1 Uni-variate Linear LS

2.1.1 Definition

The simplest sub-case is the (random) experiment that produces a set of M points (or measurements), $\{x_k, y_k\}_{k=1}^{M}$ [?].The **linear model** is

$$y = f(x; w_0, w_1) = w_0 + w_1 x + \epsilon, \tag{2.1}$$

where w_0 and w_1 are the model weights (or parameters) and ϵ is zero-mean noise.The model outcomes (predictions) are

$$\hat{y}_k = f(x_k; w_0, w_1) = w_0 + w_1 x_k, \tag{2.2}$$

where $\hat{y}_k$ is the prediction outcome of x_k.The performance **metric** is mean-square error (MSE) that is given by

$$\begin{aligned} J_{mse}(w_0, w_1) &= \frac{1}{M}\sum_{k=1}^{M}(y_k - \hat{y}_k)^2 \\ &= \frac{1}{M}\sum_{k=1}^{M} e_k^2 \end{aligned} \tag{2.3}$$

or root-MSE (RMSE)

$$J_{rmse}(w_0, w_1) = \sqrt{J_{mse}(w_0, w_1)}. \tag{2.4}$$

Note, sometimes MSE is termed as sum of squared errors (SSE).For both of these metrics, the corresponding **loss** (or cost) function to minimize is

$$\begin{aligned} \mathscr{L}(w_0, w_1) &= \sum_{k=1}^{M}(y_k - \hat{y}_k)^2 \\ &= \sum_{k=1}^{M}(y_k - w_0 - w_1 x_k)^2 \end{aligned} \tag{2.5}$$

since either root and/or constant multiplication does not change the desired minimum,

$$\begin{aligned} w_0, w_1 &= \arg\min_{w_0, w_1} J_{mse}(w_0, w_1) \\ &= \arg\min_{w_0, w_1} J_{rmse}(w_0, w_1) \\ &= \arg\min_{w_0, w_1} \mathscr{L}(w_0, w_1) \end{aligned} \tag{2.6}$$

Note that loss function and performance metrics does not have to be the same.

2.1.2 Minimization

This minimum is given by a solution of the set of equations,

$$\begin{cases} \dfrac{\partial}{\partial w_0}\mathscr{L}(w_0, w_1) = 0 \\ \dfrac{\partial}{\partial w_1}\mathscr{L}(w_0, w_1) = 0 \end{cases} \tag{2.7}$$

The resulting equations are

$$\begin{cases} 2\displaystyle\sum_{k=1}^{M}(y_k - w_0 - w_1 x_k)\cdot(-1) = 0 \\ 2\displaystyle\sum_{k=1}^{M}(y_k - w_0 - w_1 x_k)\cdot(-x_k) = 0 \end{cases} \tag{2.8}$$

After some basic algebraic manipulations, the resulting set of equations is

$$\begin{cases} w_0 M & +w_1 \sum_{k=1}^{M} x_k = \sum_{k=1}^{M} y_k \\ w_0 \sum_{k=1}^{M} x_k & +w_1 \sum_{k=1}^{M} x_k^2 = \sum_{k=1}^{M} x_k y_k \end{cases} \tag{2.9}$$

This set of equations is termed *normal equation*.The interesting and numerically stable form of the numerical solution is by usage of average estimation by mean,

$$E[\mathbf{z}] = \bar{\mathbf{z}} = \frac{1}{N}\sum_{k=1}^{N} z_k \tag{2.10}$$

$$\operatorname{Var}[\mathbf{z}] = \overline{\mathbf{z}^2} - \bar{\mathbf{z}}^2 \tag{2.11}$$

$$\operatorname{Cov}[\mathbf{x}, \mathbf{y}] = \overline{\mathbf{xy}} - \bar{\mathbf{x}}\bar{\mathbf{y}} \tag{2.12}$$

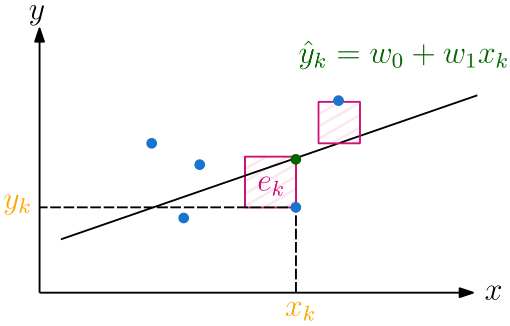


Figure 2.1: Linear regression visualization. The goal is to minimize the total area $\sum_k e_k^2$ of the rectangles.

The resulting prediction is

$$\hat{\mathbf{y}} = E[\mathbf{y}] + \frac{\text{Cov}[\mathbf{x},\mathbf{y}]}{\text{Var}[\mathbf{x}]}(\mathbf{x} - E[\mathbf{x}]) \tag{2.13}$$

This is *probabilistic* result.Notes:

- $\text{Var}[\mathbf{x}] \neq 0$ requirement.
- $E[\mathbf{y}] = E[\mathbf{x}] = 0 \Rightarrow w_0 = 0$.

Concluding notes:

- The resulting model is also termed as linear regression, linear trend-line and linear prediction.
- The straightforward solution may result in ill-conditioned matrix. Reformulation of the solution can result in a better numerical stability, e.g. [**?**, Ch. 5, Question 5, pp. 260]. There are more accurate algorithms than just multiply inverse matrix.
- For numerical stability, the variance of x_k samples is required to be non-zero (distinct x_k values).

2.2 Vector/Matrix Notation

2.2.1 Uni-variate model

To improve the mathematical representation, vector notation can be used. This time, the points $\{x_k, y_k\}_{k=1}^M$ are organized into vectors, with a few additional ones, as follows,

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_M \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix}, \quad \mathbf{1}_M = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \in \mathbb{R}^M, \ \mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} \tag{2.14}$$

The resulting model notation is

$$\hat{\mathbf{y}} = f(\mathbf{X};\mathbf{w}) = \mathbf{1}_M w_0 + \mathbf{x} w_1 = \mathbf{X}\mathbf{w}, \tag{2.15}$$

where $\mathbf{X} = \begin{bmatrix} \mathbf{1}_M & \mathbf{x} \end{bmatrix} \in \mathbb{R}^{M\times 2}$ and $\mathbf{w} = \begin{bmatrix} w_0 & w_1 \end{bmatrix}^T$.The corresponding loss functions is

$$\begin{aligned} \mathscr{L}(\mathbf{w}) &= (\mathbf{y} - \hat{\mathbf{y}})^T(\mathbf{y} - \hat{\mathbf{y}}) = \|\mathbf{y} - \hat{\mathbf{y}}\|^2 \\ &= (\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 \end{aligned} \tag{2.16}$$

and the corresponding optimal minimum (Eq. (2.6)) results from the solution of normal equation (matrix form)

$$\nabla_{\mathbf{w}}\mathscr{L}(\cdot) = -\mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = 0 \tag{2.17}$$

and is given by

$$\begin{gathered} \mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = \mathbf{X}^T\mathbf{y} - \mathbf{X}^T\mathbf{X}\mathbf{w} = 0 \\ \mathbf{X}^T\mathbf{y} = \left(\mathbf{X}^T\mathbf{X}\right)\mathbf{w} \end{gathered}$$

Finally,

$$\mathbf{w}_{opt} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y} \tag{2.18}$$

2.2.2 Multivariate LS

For the multivariate N-dimensional formulation,

$$\mathbf{X} = \begin{bmatrix} \mathbf{1} & \mathbf{x}_1 & \cdots & \mathbf{x}_N \end{bmatrix} \in \mathbb{R}^{M\times(N+1)} \tag{2.19}$$

$$\mathbf{w} = \begin{bmatrix} w_0 & w_1 & \cdots & w_N \end{bmatrix}^T \in \mathbb{R}^{N+1} \tag{2.20}$$

All the LS discussion on$\hat{\mathbf{y}} = \mathbf{X}\mathbf{w}$is the same independent from the number of variables.

Dataset

All the data rows in $(\mathbf{X},\mathbf{y})$ are called dataset.The matrix $\mathbf{X}$ is assumed *full-rank*, i.e. columns are linearly independent.

Moore–Penrose inverse (pseudo-inverse)

Moore–Penrose inverse is the extension of an ordinary inverse matrix for none-rectangular matrices,

$$\mathbf{X}^+ = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T, \tag{2.21}$$

such that

$$\mathbf{X}^+\mathbf{X} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{X} = \mathbf{I}$$

Note, the by-definition implementation of $\mathbf{X}^+$ may have numerical stability problems with $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}$.All the modern programming languages have numerically-stable and efficient implementation of pseudo-inverse calculations.The common numerical notation is

$$\mathbf{w}_{opt} = \mathbf{X}^+\mathbf{y} \tag{2.22}$$

Implementation note: there are numerically optimized algorithms for $\mathbf{w}_{opt}$, such as:

1. `lsqminnorm` (Matlab)
2. Python, `numpy.linalg.lstsq` and`scipy.linalg.lstsq`

Projection matrix

The model output is given by

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{X}\mathbf{w} = \mathbf{X}\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y} \\ &= \mathbf{X}\mathbf{X}^{+}\mathbf{y} = \mathbf{P}\mathbf{y}\end{aligned} \tag{2.23}$$

where

$$\mathbf{P} = \mathbf{X}\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T \tag{2.24}$$

is a *projection* matrix, i.e. projection of $\mathbf{y}$ into a base derived from $\mathbf{X}$.Important properties of the matrix $\mathbf{P}$:

- Symmetric $\mathbf{P} = \mathbf{P}^T$,
- Idempotent $\mathbf{P} = \mathbf{P}^2$,
- Orthogonality, $\mathbf{P} \perp (\mathbf{I} - \mathbf{P})$
  Proof. $\mathbf{P}(\mathbf{I} - \mathbf{P}) = \mathbf{P} - \mathbf{P}^2 = \mathbf{0}$.
- $\mathbf{I} - \mathbf{P}$ is also projection matrix.

Model error

The model error is

$$\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - \mathbf{P}\mathbf{y} = (\mathbf{I} - \mathbf{P})\,\mathbf{y}, \tag{2.25}$$

such that$\mathscr{L}(\mathbf{w}) = \overline{\mathbf{e}^2}$.

Error and data orthogonality

$$\mathbf{e} \perp \mathbf{X} \Rightarrow \mathbf{X}^T\mathbf{e} = \mathbf{0} \tag{2.26}$$

Proof:

$$\begin{aligned}\mathbf{X}^T\mathbf{e} = \mathbf{X}^T\mathbf{y} - \mathbf{X}^T\mathbf{X}\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y} \\ \mathbf{X}^T\mathbf{y} - \mathbf{X}^T\mathbf{y} = \mathbf{0}\end{aligned} \tag{2.27}$$

Error and prediction orthogonality

$$\mathbf{e} \perp \hat{\mathbf{y}} \Rightarrow \hat{\mathbf{y}}^T\mathbf{e} = \mathbf{e}^T\hat{\mathbf{y}} = 0 \tag{2.28}$$

Proof:

$$\begin{aligned}\hat{\mathbf{y}}^T\mathbf{e} &= \mathbf{y}^T\mathbf{P}\left(\mathbf{I} - \mathbf{P}\right)\mathbf{y} \\ &= \mathbf{y}^T\mathbf{P}\mathbf{y} - \mathbf{y}^T\mathbf{P}\mathbf{P}\mathbf{y} \\ &= \mathbf{y}^T\mathbf{P}\mathbf{y} - \mathbf{y}^T\mathbf{P}\mathbf{y} = 0\end{aligned} \tag{2.29}$$

The interesting outcome of this property is a relation between error and prediction,

$$\|\mathbf{y}\|^2 = \|\hat{\mathbf{y}}\|^2 + \|\mathbf{e}\|^2 \tag{2.30}$$

Proof.

$$\begin{aligned}\|\mathbf{y}\|^2 &= \|\hat{\mathbf{y}} + \mathbf{e}\|^2 \\ &= (\hat{\mathbf{y}} + \mathbf{e})^T(\hat{\mathbf{y}} + \mathbf{e}) \\ &= \hat{\mathbf{y}}^T\hat{\mathbf{y}} + \mathbf{e}^T\mathbf{e}\end{aligned} \tag{2.31}$$

Average error

The average error is zero-mean,

$$\begin{aligned}\bar{\mathbf{e}} &= \frac{1}{M}\sum_{k=1}^{M} e_k \\ &= \sum_{k=1}^{M} e_k = \mathbf{1}^T\mathbf{e} = 0\end{aligned} \tag{2.32}$$

Proof.

$$\begin{aligned}&\mathbf{X}^T\underbrace{(\mathbf{y} - \mathbf{X}\mathbf{w})}_{\mathbf{e}} = 0 \\ \Rightarrow &\begin{bmatrix}\mathbf{1}^T \\ \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_N^T\end{bmatrix}\mathbf{e} = \begin{bmatrix}\mathbf{1}^T\mathbf{e} \\ \vdots\end{bmatrix} = 0\end{aligned} \tag{2.33}$$

The interesting consequence is

$$\begin{aligned}\bar{\mathbf{y}} &= \bar{\hat{\mathbf{y}}} \qquad &(2.34\text{a}) \\ &= w_0 + w_1\bar{\mathbf{x}}_1 + \cdots + w_N\bar{\mathbf{x}}_N \qquad &(2.34\text{b})\end{aligned}$$

Proof.

$$\begin{aligned}\bar{\mathbf{y}} &= \overline{\hat{\mathbf{y}} + \mathbf{e}} \\ &= \bar{\hat{\mathbf{y}}} + \bar{\mathbf{e}}\end{aligned} \tag{2.35}$$

Error distribution

The values of the error vector $\mathbf{e}$ are assumed to be normally distributed, due to Central Limit Theorem (CLT). Typically, this assumption is not need in ML, but it is important for statistical analysis for small values of M.

MSE

The reduced expression for the resulting minimal loss is

$$\begin{aligned}\mathscr{L}_{min} &= \sum_{k=1}^{M} y_k^2 - \sum_{j=0}^{N} w_j\mathbf{y}^T\mathbf{x}_j \\ &= \mathbf{y}^T\mathbf{y} - \mathbf{y}^T\mathbf{X}\mathbf{w}\end{aligned} \tag{2.36}$$

Proof.

$$\begin{aligned}mse_{min} &= \mathbf{e}^T\mathbf{e} \\ &= (\mathbf{y} - \hat{\mathbf{y}})^T\mathbf{e} \\ &= \mathbf{y}^T\mathbf{e} - \cancel{\hat{\mathbf{y}}^T\mathbf{e}} \\ &= \mathbf{y}^T(\mathbf{y} - \mathbf{X}\mathbf{w}) \\ &= \mathbf{y}^T\mathbf{y} - \underbrace{\mathbf{y}^T\begin{bmatrix}\mathbf{1} & \mathbf{x}_1 & \cdots & \mathbf{x}_N\end{bmatrix}}_{\mathscr{R}^{1\times(N+1)}}\mathbf{w}\end{aligned} \tag{2.37}$$

The MSE or RMSE evaluation from the loss is straightforward.

Chapter 3

Basic Signal Analysis

Goal: This chapter introduces the fundamental concepts and methods for analyzing and estimating (learning) parameters of a discrete-time sinusoidal signal observed in additive noise.

3.1 Signal Preliminaries

A general continuous-time cosine signal can be written as

$$\begin{aligned} y(t) &= A\cos(2\pi F_0 t + \theta) + \epsilon(t), \\ &= A\cos(\Omega_0 t + \theta) + \epsilon(t), \end{aligned} \tag{3.1}$$

where

- $A > 0$ is the amplitude,
- $-\pi < \theta \leq \pi$ is the phase,
- F_0 is the frequency in Hz,
- $\Omega_0 = 2\pi F_0$ is the radial frequency in rad/sec,
- $\epsilon(t)$ is zero-mean additive noise.

The only assumption for the additive noise is that it is zero-mean,

$$\sum_n \epsilon[n] = 0. \tag{3.2}$$

No additional assumptions, such as Gaussianity, are applied; however, the special case of additive white Gaussian noise (AWGN) is is further refined as tips for selected topics.For the further analysis, we use the sampled version $x[n]$ of the continuous-time signal $x(t)$, sampled with frequency $F_s = 1/T$,

$$\begin{aligned} y[n] &= y(nT) \\ &= A\cos(\omega_0 n + \theta) + \epsilon[n] \quad n = 0, \ldots, L-1, \end{aligned} \tag{3.3}$$

where

$$\omega_0 = 2\pi F_0 T = 2\pi \frac{F_0}{F_s} \tag{3.4}$$

is the angular frequency (measured in rad) derived from the analog frequency F_0 and L is the resulting number of samples.In order to accurately reproduce a cosine signal, the Nyquist criterion demands $F_0 < F_s/2$, which implies $\omega_0 < \pi$. This requirement can be easily illustrated by the following example.Consider two signals:

$$x_1(t) = \cos(0.6\pi t), \quad x_2(t) = \cos(2.6\pi t)$$

Sampling with $F_s = 1$ Hz results in two identical signals,

$$x_1[n] = \cos(0.6\pi n),$$

$$x_2[n] = \cos(2.6\pi n) = \cos(0.6\pi n + 2\pi n) = x_1[n].$$

This phenomenon is called aliasing.Note, when $\omega_0 = 0$ the signal is the DC level, $y(t) = y[n] = A\cos(\theta)$.Therefore, the sampling frequency requirement is $0 \leqslant \omega < \pi$. This relation holds for all the following discussions and derivations.The energy of the signal $x[n]$ is defined as

$$E_{\mathbf{x}} = \|\mathbf{x}\|^2 = \sum_n x^2[n], \tag{3.5}$$

where $\mathbf{x}$ is the vector of samples of the signal $x[n]$.The corresponding power is

$$P_{\mathbf{x}} = \frac{1}{N} E_{\mathbf{x}} = \frac{1}{N}\|\mathbf{x}\|^2. \tag{3.6}$$

3.2 Amplitude estimation

Goal: Find the amplitude of a sinusoidal signal in noise that best fits the model in a least squares (LS) sense.

Given a signal model with a known frequency ω_0,

$$y[n] = A\cos(\omega_0 n) + \epsilon[n] \quad n = 0, \ldots, L-1 \tag{3.7}$$

the goal is to estimate the amplitude A that best fits a provided model.Technically, we are looking for the value of A that minimizes the squared error,

$$\mathscr{L}(A) = \sum_n (y[n] - A\cos(\omega_0 n))^2. \tag{3.8}$$

In the linear LS regression formulation, we define the corresponding parameters $\mathbf{y}, \mathbf{X}$ and $\mathbf{w}$. First, the required weight is $\mathbf{w} = A$.The matrix $\mathbf{X}$ is formed by samples of the signal$\{\cos(\omega_0 n)\}_{n=0}^{L-1}$,

$$\mathbf{X} = \begin{bmatrix} 1 & \cos(\omega_0) & \cos(2\omega_0) & \cdots & \cos\big((L-1)\omega_0\big) \end{bmatrix}^T \tag{3.9}$$

Finally, $\mathbf{y}$ is the vector of samples of $\{y[n]\}_{n=0}^{L-1}$. The resulting solution is straightforward,

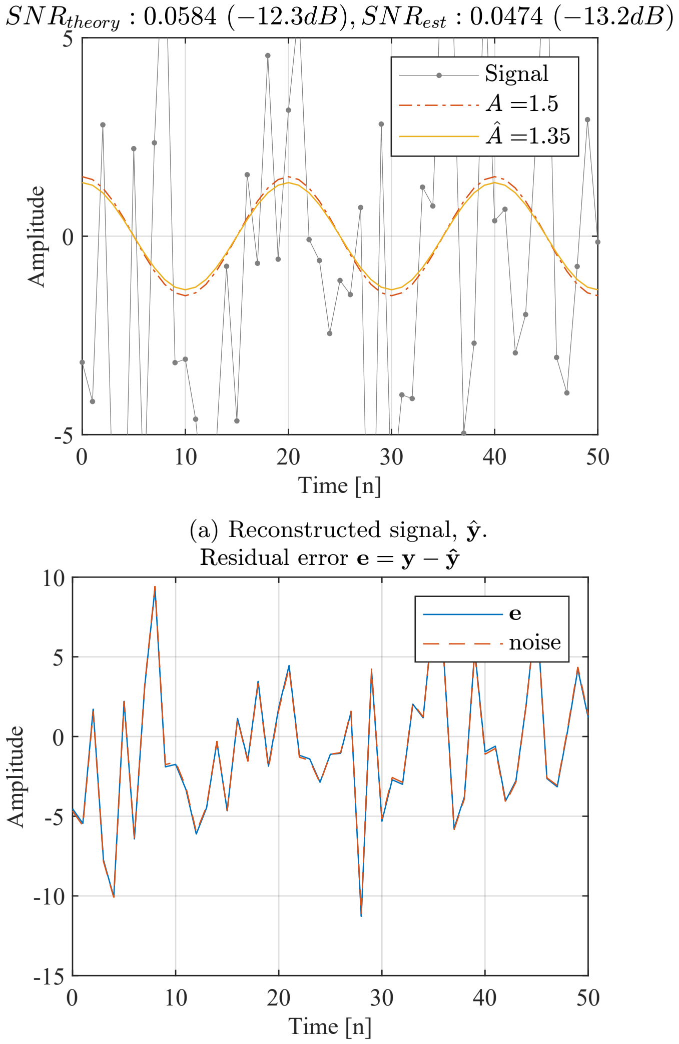
$$\begin{aligned} \hat{A} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \\ &= \frac{\sum_n x[n]y[n]}{\sum_n x^2[n]} \\ &= \frac{\sum_n y[n]\cos(\omega_0 n)}{\sum_n \cos^2(\omega_0 n)} \end{aligned} \tag{3.10}$$

If we substitute the model into the resulting solution,

$$\begin{aligned}\hat{A} &= \frac{\sum_n x[n]\left(Ax[n]+\epsilon[n]\right)}{\sum_n x^2[n]} \\ &= A + \frac{\sum_n x[n]\epsilon[n]}{\sum_n x^2[n]},\end{aligned} \tag{3.11}$$

it produces a true value of A with some additive noise.The resulting residual error is given by

$$\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}}. \tag{3.12}$$



(a) Reconstructed signal, $\hat{\mathbf{y}}$.

(b) Residual error. Ideally, if the model was perfect, the residual would be equal to the added noise.

Figure 3.1: Example of the cosine signal amplitude estimation.

Since $\mathbf{e} \perp \hat{\mathbf{y}}$ the power/energy terms can be decomposed as follows,

$$\|\mathbf{y}\|^2 = \|\hat{\mathbf{y}}\|^2 + \|\mathbf{e}\|^2, \quad P_{\mathbf{y}} = P_{\hat{\mathbf{y}}} + P_{\mathbf{e}}. \tag{3.13}$$

An interesting interpretation of this result is estimated signal to noise ratio (SNR), defined as

$$\widehat{SNR} = \frac{\left\|\hat{\mathbf{y}}^2\right\|}{\left\|\mathbf{e}^2\right\|} \tag{3.14}$$

Moreover, due to zero-mean property of the noise, the estimated variance of the noise is

$$\hat{\sigma}_\epsilon^2 = \frac{1}{L}\|\hat{\mathbf{e}}\|^2. \tag{3.15}$$

The following example (Fig. 3.1) uses a synthetic cosine signal of length $L = 51$ samples, angular frequency$\omega_0 = 0.1\pi$ and amplitude $A = 1.5$. Gaussian noise with standard deviation $\sigma = 5$ is then added to create a noisy observation. A least-squares regression is applied to estimate the amplitude, yielding $\hat{\sigma}_\epsilon = 4.43$.

3.3 Amplitude and phase estimation

Goal: Find amplitude and phase of a sinusoidal signal in noise.

The following analysis is provided for the more general model,

$$y[n] = A\cos\left(\omega_0 n + \theta\right) + \epsilon[n] \quad n = 0, \dots, L-1, \tag{3.16}$$

with two unknown parameters, the amplitude A and the phase θ.The linear LS reformulation of the signal model

$$\hat{y}[n] = A\cos\left(\omega_0 n + \theta\right) \tag{3.17}$$

involves the use of trigonometric identities to express the cosine with a phase shift as a linear combination of sine and cosine signals,

$$A\cos\left(\omega_0 n + \theta\right) = w_c\cos(\omega_0 n) + w_s\sin(\omega_0 n), \tag{3.18}$$

where

$$\begin{aligned} w_c &= A\cos(\theta) \\ w_s &= -A\sin(\theta). \end{aligned} \tag{3.19}$$

This transforms the problem into a two-parameter linear LS problem in terms of w_c and w_s [?].The resulting LS formulation involves a two-valued vector of linear coefficients,$\mathbf{w} = \begin{bmatrix} w_c & w_s \end{bmatrix}^T$, the vector $\mathbf{y}$ of samples of $y[n]$, and the matrix $\mathbf{X}$ of dimensions $L \times 2$ that is given by

$$\mathbf{X} = \begin{bmatrix} 1 & 0 \\ \cos(\omega_0) & \sin(\omega_0) \\ \cos(2\omega_0) & \sin(2\omega_0) \\ \vdots & \vdots \\ \cos\big((L-1)\omega_0\big) & \sin\big((L-1)\omega_0\big) \end{bmatrix}. \tag{3.20}$$

Once $\hat{\mathbf{w}}$ has been found, the amplitude and phase can be recovered from

$$\begin{aligned} A &= \sqrt{w_c^2 + w_c^2} \\ \theta &= -\arctan\left(\frac{w_c}{w_s}\right) \end{aligned} \tag{3.21}$$

SNR and noise variance interpretations are similar to in the previous model in Eqs. (3.14) and (3.15).The numerical example is presented in Fig. 3.2. The configuration is similar to the previous figure, expect the lower noise variance, $\sigma = 1.5$. Nevertheless, there is a decrease in performance, since two parameters are estimated simultaneously.

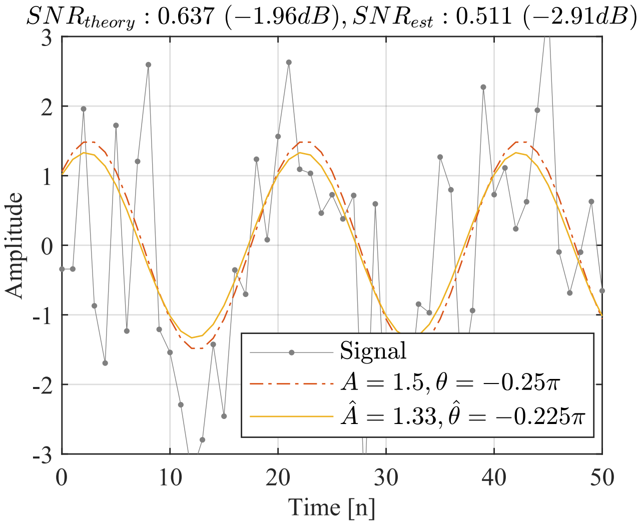


Figure 3.2: Example of the cosine signal amplitude and phase estimation.

Implementation Tip

- This estimation procedure is optimal in the maximum likelihood (ML) sense under additive white Gaussian noise (AWGN) and achieves the Cramér-Rao lower bound (CRLB) [?, ?].
- The theoretical lower bound (also termed Cramer-Rao lower bound(CRLB)) on the average estimation accuracy of w_c, w_s is given by [?, Eqs. (5.47-48)]

$$\operatorname{Var}\left[\hat{w}_{c,s}\right] \gtrsim \frac{2\sigma^2}{L} \tag{3.22}$$

  This bound is the tightest for the AWGN case and is less accurate for other noise distributions.
- The approximated estimation variance can be easily evaluated by Monte-Carlo simulations for any set of parameters and any distribution of interest.

3.4 Frequency estimation

Goal: If the frequency ω_0 is also unknown, it can be estimated by searching for the$\hat{\omega}_0$ that best fits a sinusoidal model for the observed data, i.e., that minimizes the residual error norm or maximizes the reconstructed signal energy.

Since ω_0 is unknown, the matrix $\mathbf{X}$ may be parameterized as a frequency-dependent one, $\mathbf{X}(\omega)$.Here, the estimated signal is frequency-dependent

$$\hat{\mathbf{y}}(\omega) = \mathbf{X}(\omega)\mathbf{w}(\omega), \tag{3.23}$$

where $\mathbf{w}(\omega)$ are the estimated parameters w_c and w_s at that frequency. The corresponding frequency-dependent residual error is given by

$$\mathbf{e}(\omega) = \mathbf{y} - \hat{\mathbf{y}}(\omega). \tag{3.24}$$

Since the error $\mathbf{e}(\omega)$ is orthogonal to $\hat{\mathbf{y}}$,

$$\|\mathbf{y}\|^2 = \|\hat{\mathbf{y}}(\omega)\|^2 + \|\mathbf{e}(\omega)\|^2. \tag{3.25}$$

To find the frequency that best represents the data, we seek the one that maximizes the energy of the reconstructed signal (or equivalently minimizes the residual error), as mentioned above

$$\hat{\omega_0} = \arg\min_{\omega} \|\mathbf{e}(\omega)\|^2 = \arg\max_{\omega} \|\hat{\mathbf{y}}(\omega)\|^2. \tag{3.26}$$

Note, this optimization problem can be challenging because the objective function may exhibit multiple local maxima/minima. Therefore, an appropriate numerical global optimization method is required.Once $\hat{\omega}_0$ has been found, the amplitude and phase are estimated using the corresponding linear LS solution $\mathbf{w}(\omega_0)$. This solution also results in SNR and noise variance estimations, as in Eqs. (3.14) and (3.15).

Tip: Interpretation in Terms of the Periodogram
The function

$$P(\omega) = \frac{1}{L}\|\hat{\mathbf{y}}(\omega)\|^2 \tag{3.27}$$

as a function of ω is termed a periodogram that is a frequency-dependent measure of signal power that approximates the power spectral density (PSD) of the signal. By scanning over frequencies, the ω that yields the maximum periodogram value is taken as the frequency estimate, ω_0.A numerical example of the signal with additive white Gaussian noise (AWGN), and with the parameters $A = 1.5, \omega_0 = 0.1\pi, \theta = -\pi/4$ and $\sigma_\epsilon^2 = 1$, is presented in Fig. 3.3.First, periodogram peak is found (Fig. 3.3a).Than, the subsequent amplitude/phase estimation result is presented (Fig. 3.3b).

Tip: Theoretical performance bounds Under AWGN assumption, theoretical SNR is given by

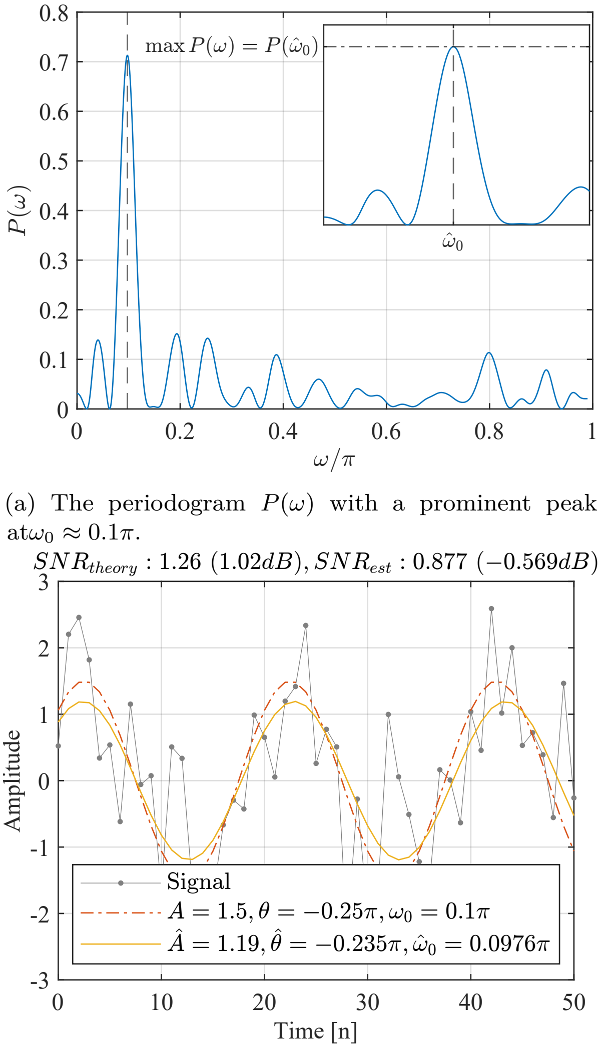
$$SNR = \frac{A^2}{2\sigma^2} \tag{3.28}$$

and the corresponding CRLB on the estimation variances are [?]

$$\operatorname{Var}\left[\hat{A}\right] \geqslant \frac{2\sigma^2}{L} \quad [V^2] \tag{3.29}$$

$$\operatorname{Var}[\hat{\omega}_0] \geqslant \frac{12}{SNR \times L(L^2-1)} \approx \frac{12}{SNR \times L^3} \quad \left[\left(\frac{rad}{sample}\right)^2\right] \tag{3.30}$$

$$\operatorname{Var}\left[\hat{\theta}\right] \geqslant \frac{2(2L-1)}{SNR \times L(L+1)} \approx \frac{4}{SNR \times L} \quad [rad^2] \tag{3.31}$$



(a) The periodogram $P(\omega)$ with a prominent peak at$\omega_0 \approx 0.1\pi$.

(b) Reconstracted signal.

Figure 3.3: The reconstruction in (b) uses the estimated amplitude, phase, and angular frequency$(\hat{A}, \hat{\theta}, \hat{\omega}_0)$found by maximizing the periodogram in (a).

For analog frequency $F_0 = \frac{\omega_0}{2\pi} F_s$,

$$\mathrm{Var}[F_0] = \mathrm{Var}[\omega_0] \left(\frac{F_s}{2\pi}\right)^2 \quad [Hz^2] \tag{3.32}$$

In practice, for short data lengths or non-Gaussian noise, these bounds provide only approximate guides to achievable performance.

3.5 Harmonic Signal Analysis

A particularly important class of signals encountered in many practical applications is the *harmonic* or *periodic* signal. Such a signal can be expressed as a sum of cosine terms whose frequencies are integer multiples (harmonics) of a fundamental frequency ω_0.

$$y[n] = A_0 + \sum_{m=1}^{M} A_m \cos(m\omega_0 n + \theta_m), \tag{3.33}$$

where:

- A_0 is the constant (DC) component,
- A_m and θ_m represent the amplitude and phase of the m-th harmonic,
- ω_0 is the fundamental angular frequency,
- $m\omega_0$ corresponds to the frequency of the m-th harmonic,
- and M is the number of harmonics in the model.

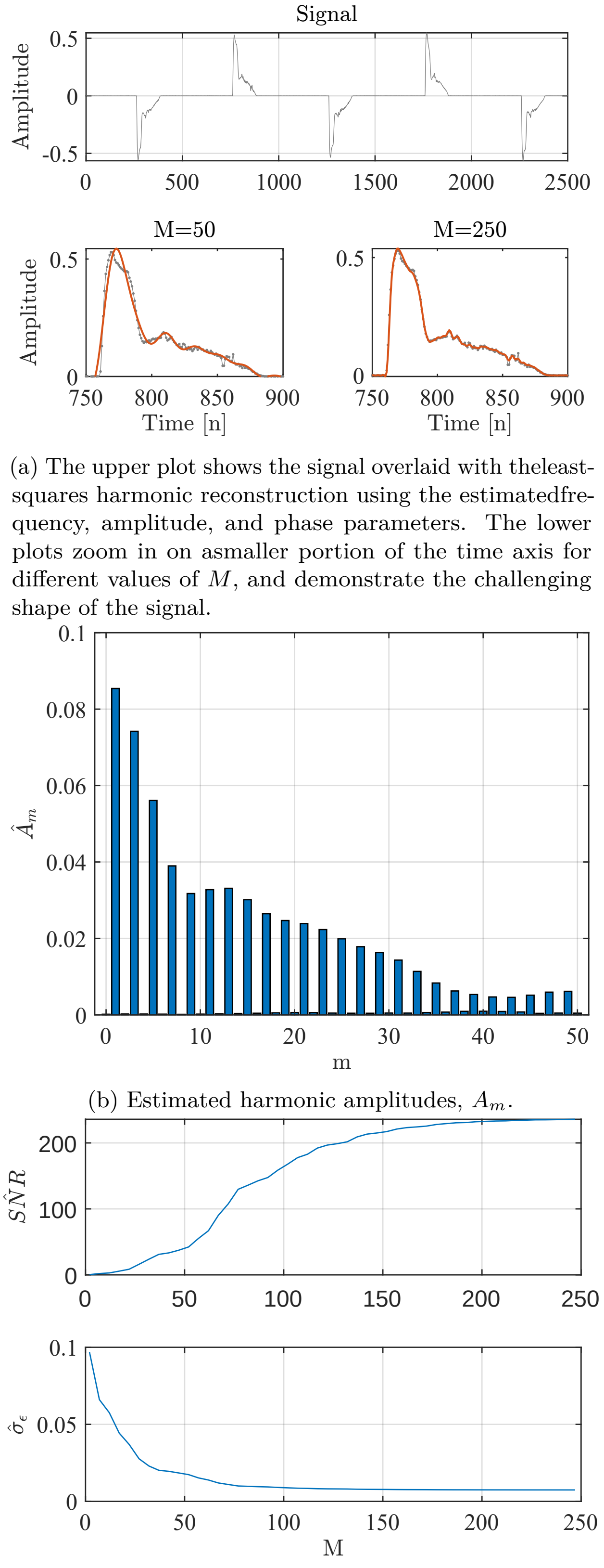
Given ω_0, the model is linear in terms of the unknown parameters $\{A_m, \theta_m\}$ for each harmonic $m = 1, \ldots, M$. Similar to the single-frequency case, the LS matrix $\mathbf{X}$ is constructed with columns corresponding to $\cos(m\omega_0 n)$ and $\sin(m\omega_0 n)$ for $m = 1, \ldots, M$, plus a column of ones for the DC component.Each pair (A_m, θ_m) can be recovered from the LS estimated cosine and sine coefficients in the manner described for single-frequency amplitude-phase estimation. The resulting SNR and noise variance estimates are similar to those described in the previous sections.The model order M (number of harmonics) is a hyper-parameter that should be chosen carefully. Too few harmonics can fail to capture essential signal structure, while too many may overfit noise. The maximum value of M is bounded by the Nyquist criterion, $M < \pi/\omega_0$.If ω_0 is not known, the approach that is described in the frequency estimation section can also be applied here. Once $\hat{\omega}_0$ has been determined from a maximum of the harmonic periodogram,

$$P_h(\omega) = \frac{1}{L} \sum_{m=1}^{M} \left\| \mathbf{y}(m\omega) \right\|^2, \tag{3.34}$$

the harmonic amplitudes and phases can be estimated via LS at this frequency [?].Total harmonic distortion (THD) is a measure commonly used in electrical engineering, audio processing, and other fields to quantify how much the harmonic components of a signal differ from a pure sinusoid at the fundamental frequency. It is defined as the ratio of the root-sum-square of the harmonic amplitudes and the amplitude of the fundamental frequency,

$$THD = \frac{\sqrt{\sum_{m=2}^{M} A_m^2}}{A_1}. \tag{3.35}$$

A lower THD value indicates that the signal is closer to a pure sinusoidal shape, whereas a higher THD signifies a stronger presence of higher-order harmonics.The example is the sampled current of a switch-mode power supply in a 50Hz network sampled at a 50kHz frequency [?]. Figure 3.4a shows a reconstruction of the signal with $M = 250$ harmonics.The estimated amplitudes $\hat{A}_m$ are shown (Fig. 3.4b) as a function ofthe harmonic index m, including the DC term at $m = 0$.A larger magnitude indicates a more prominent harmonic component.The first non-DC harmonic amplitude $m = 1$ corresponds tothe fundamental frequency, ω_0, while higher indices capture additionalharmonics in the signal. The estimated fundamental frequency is 50.104Hz with the corresponding THD of about 1.6.Figure 3.4c shows estimated SNR (top)and the noise standard deviation (bottom) vary as the number ofharmonics M in the model increases.**Tip:**



(a) The upper plot shows the signal overlaid with theleast-squares harmonic reconstruction using the estimatedfrequency, amplitude, and phase parameters. The lower plots zoom in on asmaller portion of the time axis for different values of M, and demonstrate the challenging shape of the signal.

(b) Estimated harmonic amplitudes, A_m.

(c) Estimated SNR and noise std, $\hat{\sigma}_\epsilon$.

Figure 3.4: Example of a harmonic signal analysis.

The frequency estimator is an effective ML estimator with known analytical CRLB [?].

3.6 Discrete Fourier Transform (DFT)

The discrete Fourier transform (DFT) can be viewed as a systematic way of decomposing a finite-length signal into a sum of harmonically related sinusoids. In fact, it is a special case of the harmonic signal representation discussed earlier. Specifically, setting the fundamental angular frequency to $\omega_0 = \frac{2\pi}{N}$ and using $N \geq L-1$ harmonics, the harmonic model reduces exactly to a DFT decomposition that provides a natural harmonic decomposition of the signal into N harmonics that are evenly spaced in frequency.DFT representation assumes that any arbitrary, finite-time signal $y[n]$ may be represented as a sum of sinusoidal signals,

$$y[n] = \sum_{k=0}^{N-1} A_k \cos\left(k\frac{2\pi}{N}n + \theta_k\right), \quad n = 0, \ldots, L-1 \tag{3.36}$$

When $N \geq L$, the DFT allows for perfect reconstruction of the signal using its harmonic representation:

$$\mathbf{y} = \mathbf{X}\hat{\mathbf{w}}.$$

It is also worth noting the symmetry in the DFT,

$$\begin{aligned} \cos\big((N-k)\Delta\omega\big) &= \cos\big(k\Delta\omega\big) \\ \sin\big((N-k)\Delta\omega\big) &= -\sin\big(k\Delta\omega\big), \end{aligned} \tag{3.37}$$

resulting in redundant information for frequencies $k\omega_0$ above and below π,

$$\begin{aligned} A_k &= A_{N-k} \\ \theta_k &= -\theta_{N-k} \end{aligned}. \tag{3.38}$$

As a result, only frequencies $k\omega_0 \leq \pi$ need to be considered uniquely.

3.6.1 Single frequency analysis

Consider a signal $y[n]$ assume a discrete frequency $\omega_0 = \frac{2\pi}{N}k$ is given. To estimate amplitude and phase at this predefined frequency, we can form a matrix $\mathbf{X}$,

$$\mathbf{X}_1 = \begin{bmatrix} \mathbf{x}_c & \mathbf{x}_s \end{bmatrix},$$

where

$$\mathbf{x}_c = \begin{bmatrix} \cos\left(2\pi\frac{k}{N}\cdot 0\right) \\ \cos\left(2\pi\frac{k}{N}\cdot 1\right) \\ \vdots \\ \cos\left(2\pi\frac{k}{N}(L-1)\right) \end{bmatrix} \text{and} \mathbf{x}_s = \begin{bmatrix} \sin\left(2\pi\frac{k}{N}\cdot 0\right) \\ \sin\left(2\pi\frac{k}{N}\cdot 1\right) \\ \vdots \\ \sin\left(2\pi\frac{k}{N}(L-1)\right) \end{bmatrix}.$$

By evaluating $\mathbf{X}_1^T\mathbf{X}_1$, we find that the sine and cosine columns form an orthogonal basis for this single frequency, with

$$\mathbf{x}_c^T\mathbf{x}_c = \frac{N}{2}, \tag{3.39a}$$

$$\mathbf{x}_s^T\mathbf{x}_s = \frac{N}{2}, \tag{3.39b}$$

$$\mathbf{x}_c^T\mathbf{x}_s = 0. \tag{3.39c}$$

Stacking these results for all $k = 0, \ldots, N-1$ yields the complete DFT matrix forms a complete orthogonal basis for the L-sample signal space.The further discussion of $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}$ matrix properties may be found in Examples 4.2 and 8.5 in [?]. Moreover, since $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}$ takes a particularly simple diagonal form, and the least squares solution $\hat{\mathbf{w}}$ for the parameters $w_{c,k}$ and $w_{s,k}$ (corresponding to amplitude and phase components at ω_k) is

$$w_{c,k} = \frac{2}{N}\sum_{n=0}^{L-1} y[n]\cos\left(2\pi\frac{k}{N}n\right), \tag{3.40a}$$

$$w_{s,k} = \frac{2}{N}\sum_{n=0}^{L-1} y[n]\sin\left(2\pi\frac{k}{N}n\right). \tag{3.40b}$$

Tip The fast Fourier transform (FFT) algorithm efficiently computes $Y[k]$, providing $A_k = |Y[k]|/N$ and $\theta_k = \angle(Y[k])$ with significantly lower memory requirements and complexity than direct calculation in Eq. (3.40). When only a single frequency value is of interest, Goertzel algorithm is more efficient method for the task. Moreover, it can be used for computationally effective peaking of the maximum in Eq. (3.26).

3.6.2 Power Spectral Density

The power of a signal of the form

$$x_k[n] = A_k\cos\left(k\frac{2\pi}{N}n + \theta_k\right) \tag{3.41}$$

is

$$P_{\mathbf{y}_k} = \frac{1}{L}\|\mathbf{y}_k\|^2 = \frac{A_k^2}{2}. \tag{3.42}$$

This value is known as the power spectral density (PSD) at the frequency $\omega = k\frac{2\pi}{N}$.The corresponding squared magnitude values $A_k^2/2$ are known as the discrete-frequency periodogram (Eq. (3.27)), and this is the basic method for the PSD estimation of a signal.Plotting such a periodogram gives a frequency-domain representation of the signal's power distribution, highlighting which frequencies carry the most power.DFT is energy conservation transform (Parseval's Theorem) that states the relation

$$\sum_{k=0}^{N-1} A_k^2 = \frac{1}{L}\|\mathbf{y}\|^2. \tag{3.43}$$

3.6.3 Spectral Spreading and Leakage

In an idealized setting, a pure cosine signal has a perfectly defined frequency representation. For instance, consider the discrete-time signal,

$$x[n] = A\cos\left(k_0\frac{2\pi}{L}n\right),\ k_0 \in \{1, \cdots, L-1\} \tag{3.44}$$

where k_0 is the frequency index. The Fourier transform of this signal yields a single spectral component at frequency $w_0 = k_0\frac{2\pi}{L}$, such that the spectral amplitude A_k at each value of k is given by

$$A_k = \begin{cases} \frac{A}{2} & k = k_0, N - k_0 \\ 0 & \text{otherwise} \end{cases}. \tag{3.45}$$

Under these conditions, the signal's spectral representation seems to be strictly localized at the specific frequency ω_k, with no energy distributed elsewhere in the spectrum.However, practical scenarios deviate from this ideal case.In particular, if a denser frequency grid is employed (i.e. $N > L$)or the frequency varies continuously (as in Eq. (3.23)), the resulting spectral distribution can differ substantially from the discrete, single-peak ideal (Fig. 3.5).This difference arises because, in general, $\mathbf{X}(\omega)^T\mathbf{X}(\omega)$ is not orthogonal as in Eq. (3.39).As a result, two effects are introduced:

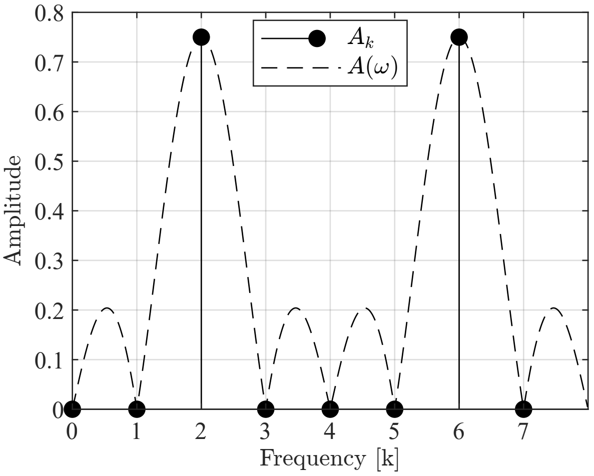


Figure 3.5: Illustration of a single-frequency cosine signal's spectrum under ideal assumptions (discrete, integer-multiple frequencies) versus practical conditions (denser frequency grid or non-integer frequencies). Note how the ideal single peak broadens and additional low-level components appear, highlighting the effects of spectral spreading and leakage.

- The main frequency peak broadens, resulting in "spectral spreading".
- Additional frequency components emerge beyond the broadened main peak, termed "spectral leakage."

3.7 Summary

The summary of the presented approach is shown in Table 3.1.The presented approach involves a design of matrix $\mathbf{X}$ and using LS to estimate unknown parameters.The key addressed task are as follows.

Amplitude Estimation With a known frequency ω_0, the amplitude$A$ is found via LS. The resulting residuals provide noise variance and SNR estimates.**Amplitude and Phase Estimation:** For known ω_0, rewriting

$$A\cos(\omega_0 n + \theta) = w_c \cos(\omega_0 n) + w_s \sin(\omega_0 n)$$

transforms the problem into a two-parameter LS regression.

Frequency Estimation: If ω_0 is unknown, it is found by searching for the frequency that maximizes the fitted signal energy.

Harmonic Signal Analysis: Signals can be expressed as sums of multiple harmonics. Extending the LS approach to multiple harmonics allows estimation of each amplitude and phase. THD quantifies deviations from a pure tone.

Discrete Fourier Transform (DFT): The DFT is a special case of harmonic modeling, decomposing a signal into equally spaced frequency components. Efficiently computed by the FFT, the DFT is central to signal spectral analysis.Although the estimators presented above have been extensively analyzed for the specific case of additive white Gaussian noise (AWGN) in the statistical signal processing literature [?, ?], conducting such an analysis requires a significantly more extensive mathematical framework. Furthermore, it is worth noting that any bias and variance in these estimators can be readily approximated via Monte Carlo simulations under various parameter settings and noise distributions.

Appendices

3.A Single frequency analysis

3.A.1 Theory

- $\mathbf{X}^T\mathbf{X}$ analysis:

$$\mathbf{X}^T\mathbf{X} = \begin{bmatrix} \mathbf{x}_c^T\mathbf{x}_c & \mathbf{x}_s^T\mathbf{x}_c \\ \mathbf{x}_s^T\mathbf{x}_c & \mathbf{x}_s^T\mathbf{x}_s \end{bmatrix} \tag{3.46}$$

with the following values

$$\begin{aligned} \mathbf{x}_c^T\mathbf{x}_c &= \sum_{n=0}^{N-1} \cos^2\left(2\pi\frac{k}{N}n\right) = \frac{N}{2} \\ \mathbf{x}_c^T\mathbf{x}_s &= \sum_{n=0}^{N-1} \cos\left(2\pi\frac{k}{N}n\right)\sin\left(2\pi\frac{k}{N}n\right) = 0 \\ &= \mathbf{x}_s^T\mathbf{x}_c \\ \mathbf{x}_s^T\mathbf{x}_s &= \sum_{n=0}^{N-1} \sin^2\left(2\pi\frac{k}{N}n\right) = \frac{N}{2} \end{aligned} \tag{3.47}$$

- $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}$ analysis: The resulting matrix

$$\left(\mathbf{X}^T\mathbf{X}\right)^{-1} = \begin{bmatrix} \frac{2}{N} & 0 \\ 0 & \frac{2}{N} \end{bmatrix} \tag{3.48}$$

- Finally, $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$ is

$$w_{c,k} = \frac{2}{N}\sum_{n=0}^{L-1} y[n]\cos\left(2\pi\frac{k}{N}n\right) \tag{3.49a}$$

$$w_{s,k} = \frac{2}{N}\sum_{n=0}^{L-1} y[n]\sin\left(2\pi\frac{k}{N}n\right) \tag{3.49b}$$

The orthogonality in more general form is given by

$$\sum_{n=0}^{N-1} \cos\left(2\pi\frac{j}{N}n\right)\cos\left(2\pi\frac{k}{N}n\right) = \frac{N}{2}\delta\left[j-k\right] \tag{3.50}$$

$$\sum_{n=0}^{N-1} \cos\left(2\pi\frac{k}{N}n\right)\sin\left(2\pi\frac{k}{N}n\right) = 0 \quad \forall j,k \tag{3.51}$$

$$\sum_{n=0}^{N-1} \sin\left(2\pi\frac{j}{N}n\right)\sin\left(2\pi\frac{k}{N}n\right) = \frac{N}{2}\delta\left[j-k\right], \tag{3.52}$$

3.1.2 Power

For a more general case of an arbitrary ω values, the signal of the form

$$y[n] = A\cos(\omega_0 n) \tag{3.53}$$

has the ω_0-dependent power,

$$P_{\mathbf{y}} = \frac{A^2}{4L}\left(1 + 2L - \frac{\sin(\omega_0 - 2L\omega_0)}{\sin(\omega_0)}\right), \tag{3.54}$$

that results from the time-limited origin of the signal $y[n]$. For the infinite number of samples, the resulting power converges to a continuous-time power expression,

$$\lim_{L\to\infty} P_{\mathbf{y}} \to \frac{A^2}{2} \tag{3.55}$$

Table 3.1: Comparison and summary of different signal estimation methods.

Task	Parameters	Matrix $\mathbf{X}$	SNR
Amplitude only	A given ω_0	A single column of $\cos(\omega_0 n)$	$\frac{\|\hat{\mathbf{y}}\|^2}{\|\mathbf{e}\|^2}$
Amplitude & phase	A, θ given ω_0	Two columns of $\cos(\omega_0 n)$ and $\sin(\omega_0 n)$	$\frac{\|\hat{\mathbf{y}}\|^2}{\|\mathbf{e}\|^2}$
Frequency estimation	ω_0, A, θ	Frequency-dependent $\cos(\omega n)$ and $\sin(\omega n)$ columns	Maximum of $\frac{\left\|\hat{\mathbf{y}}(\omega)\right\|^2}{\left\|\mathbf{e}(\omega)\right\|^2}$
Fourier series (harmonic decomposition)	$A_0, \{A_m, \theta_m\}_{m=1}^{M}$, possibly ω_0	Harmonic cos/sin columns at multiples of ω_0, $\cos(m\omega_0 n)$, $\sin(m\omega_0 n)$	$\frac{\|\hat{\mathbf{y}}\|^2}{\|\mathbf{e}\|^2}$, can include frequency dependence if ω_0 unknown
DFT	$\{A_k, \theta_k\}_{k=0}^{N-1}$	Multiple pairs of columns $\cos\left(\frac{2\pi k}{N} n\right)$, $\sin\left(\frac{2\pi k}{N} n\right)$ for $k = 0, \dots, N-1$	Not used directly. Perfect reconstruction for $N \geq L$

Chapter 4

Notation

Numbers and indexing

a	Scalar
$\mathbf{a}$	Vector
a_i	Element i of a vector a, indexing starting at 1
$\mathbf{A}$	Matrix
a_{ij}	Element i, j of a matrix $\mathbf{A}$, indexing starting at 1
$\mathscr{R}$	Real numbers domain
$\mathscr{R}^D$	D-dimensional vector
$\mathscr{R}^{D_1 \times D_2}$	matrix of a dimension $D_1 \times D_2$

Datasets

N	Number of features
M	Number of entries in the dataset
K	Number of classes
$\mathbf{w}$	Model parameters
$f(\cdot; \mathbf{w})$	Model
x_{ij}	Singe data value
$\mathbf{x}_i$	Singe data vector, i column number in $\mathbf{X}$
$\mathbf{X}$	Data matrix
$\mathbf{y}$	Target vector for the data in $\mathbf{X}$
$\hat{\mathbf{y}}$	Prediction vector of $\mathbf{y}$
y_i	Target value
$\hat{y}_i$	Predicted target value
$\mathscr{L}(\mathbf{y}, \hat{\mathbf{y}})$	Loss function (vector domain)
$\mathscr{L}(y_i, \hat{y}_i)$	Loss function (scalar domain)
$\mathbf{a}^{[k]}$	Activation of layer k
$\mathbf{z}^{[k]}$	Output of layer k
$g_k(\cdot)$	Activation function of layer k