ASTR8150/PHYS8150
Predictors for Linear & Nonlinear Regression
Gaussian Processes

Fabien Baron

Georgia State University

fbaron@gsu.edu

Fall 2025

Variance and covariance: a reminder

- Variance of a scalar-valued random variable X:
  $\sigma_X^2 = \text{var}(X) = E[(X - E[X])^2] = E[(X - E[X]) \cdot (X - E[X])]$
- Covariance between two scalar-valued random variables X and Y:

$$\text{cov}(X, Y) = E\left[(X - E\left[X\right])\left(Y - E\left[Y\right]\right)\right] = E\left[XY\right] - E\left[X\right]E\left[Y\right]$$

- Covariance matrix of random vector $\boldsymbol{X}$:

$$\Sigma_{\mathbf{XX}} = \begin{bmatrix} E[(X_1 - E[X_1])(X_1 - E[X_1])] & E[(X_1 - E[X_1])(X_2 - E[X_2])] & \cdots & E[(X_1 - E[X_1])(X_n - E[X_n])] \\ E[(X_2 - E[X_2])(X_1 - E[X_1])] & E[(X_2 - E[X_2])(X_2 - E[X_2])] & \cdots & E[(X_2 - E[X_2])(X_n - E[X_n])] \\ \vdots & \vdots & \ddots & \vdots \\ E[(X_n - E[X_n])(X_1 - E[X_1])] & E[(X_n - E[X_n])(X_2 - E[X_2])] & \cdots & E[(X_n - E[X_n])(X_n - E[X_n])] \end{bmatrix}$$
$$= E[(\boldsymbol{X} - E[\boldsymbol{X}])^\top (\boldsymbol{X} - E[\boldsymbol{X}])]$$

- Cross-covariance matrix of random vectors $\boldsymbol{X}$ and $\boldsymbol{Y}$:
  $\Sigma_{\mathbf{XY}} = E[(\boldsymbol{X} - E[\boldsymbol{X}])^\top (\boldsymbol{Y} - E[\boldsymbol{Y}])]$

Linear regression

- $y_i = f(x_i) + \epsilon_i = \theta_1 + \theta_2 x_i + \epsilon_i$ where $\epsilon \sim \mathcal{N}(0, \sigma^2)$
- In matrix notation

$$\begin{aligned} \boldsymbol{Y} &= \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_N \end{bmatrix}^\top \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} + \boldsymbol{\epsilon} \\ &= \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} + \boldsymbol{\epsilon} = \boldsymbol{X}^\top \boldsymbol{\theta} + \boldsymbol{\epsilon} \quad \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}) \end{aligned}$$

- Non-diagonal $\boldsymbol{\Sigma}$ will be used in the case the data points are covariant
- Likelihood of θ:

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{Y}) = \Pr(\boldsymbol{Y}|\boldsymbol{\theta}) = \frac{1}{\sqrt{(2\pi)^N \det(\boldsymbol{\Sigma})}} e^{-\frac{1}{2}(\boldsymbol{Y} - \boldsymbol{X}^\top \boldsymbol{\theta})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{Y} - \boldsymbol{X}^\top \boldsymbol{\theta})}$$

Maximum likelihood solution and predictor

- Likelihood of θ:

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{Y}) = \Pr(\boldsymbol{Y}|\boldsymbol{\theta}) = \frac{1}{\sqrt{(2\pi)^N \det(\boldsymbol{\Sigma})}} e^{-\frac{1}{2}(\boldsymbol{Y}-\boldsymbol{X}^\top\boldsymbol{\theta})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{Y}-\boldsymbol{X}^\top\boldsymbol{\theta})}$$

- Maximum likelihood (take the log, derive with respect to θ) results in the **normal equation** giving the most likely $\boldsymbol{\theta}$:

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{X}\boldsymbol{\Sigma}^{-1}\boldsymbol{X}^\top)^{-1}\boldsymbol{X}\boldsymbol{\Sigma}^{-1}\boldsymbol{Y}$$

- Given $\hat{\boldsymbol{\theta}}$ and any new $\boldsymbol{X}_*$, we can now predict $\boldsymbol{Y}_*$, using the predictor **projection "hat" matrix $\boldsymbol{H}$** defined as:

$$\hat{\boldsymbol{Y}} = (\boldsymbol{X}_*)^\top\hat{\boldsymbol{\theta}} = \underbrace{(\boldsymbol{X}_*)^\top(\boldsymbol{X}\boldsymbol{\Sigma}^{-1}\boldsymbol{X}^\top)^{-1}\boldsymbol{X}\boldsymbol{\Sigma}^{-1}}_{\boldsymbol{H}}\boldsymbol{Y} = \boldsymbol{H}\boldsymbol{Y}$$

- The hat matrix "puts the hat into" $\boldsymbol{Y}$
- Great but not fully Bayesian. What about $\Pr(\hat{\boldsymbol{\theta}})$, or $\Pr(\hat{\boldsymbol{Y}})$?

Marginal and Conditional Gaussians

- Important theorem, used when both the prior and likelihood are normally distributed. If we have:

$$\Pr(\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Lambda})$$
$$\Pr(\boldsymbol{y}|\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}, \boldsymbol{\Sigma})$$

  then we will have

$$\Pr(\boldsymbol{y}) \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{b}, \boldsymbol{\Sigma} + \boldsymbol{A}\boldsymbol{\Lambda}\boldsymbol{A}^\top)$$
$$\Pr(\boldsymbol{x}|\boldsymbol{y}) \sim \mathcal{N}((\boldsymbol{\Lambda}^{-1} + \boldsymbol{A}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{A})^{-1}\left(\boldsymbol{A}^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{b}) + \boldsymbol{\Lambda}^{-1}\boldsymbol{\mu}\right), (\boldsymbol{\Lambda}^{-1} + \boldsymbol{A}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{A})^{-1})$$

- Reference: Bishop "Pattern Recognition and Machine Learning", eqs 2.113 to 2.117

Application to the Linear Regression case

- Somewhat confusing, but we have: $\boldsymbol{y} \rightarrow \boldsymbol{Y}$, $\boldsymbol{x} \rightarrow \boldsymbol{\theta}$, $\boldsymbol{A} \rightarrow \boldsymbol{X}^\top$, $\boldsymbol{b} \rightarrow \mathbf{0}$, and $\Sigma \rightarrow \Sigma$
- We need to set $\Pr(\boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Lambda})$ with possibly the edge case of $\boldsymbol{\mu} = \mathbf{0}$ and $\boldsymbol{\Lambda}^{-1} \rightarrow \mathbf{0}$ to simulate a nearly uniform prior.
- We already have:

$$\Pr(\boldsymbol{Y}|\boldsymbol{\theta}, \boldsymbol{X}) \sim \mathcal{N}(\boldsymbol{X}^\top\boldsymbol{\theta}, \boldsymbol{\Sigma})$$

- The posterior distribution for $\boldsymbol{\theta}$, whose mean is the MAP:

$$\begin{aligned}\Pr(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{X}) &\sim \mathcal{N}((\boldsymbol{\Lambda}^{-1} + \boldsymbol{X}\boldsymbol{\Sigma}^{-1}\boldsymbol{X}^\top)^{-1}\left(\boldsymbol{X}\boldsymbol{\Sigma}^{-1}\boldsymbol{Y} + \boldsymbol{\Lambda}^{-1}\boldsymbol{\mu}\right), (\boldsymbol{\Lambda}^{-1} + \boldsymbol{X}\boldsymbol{\Sigma}^{-1}\boldsymbol{X}^\top)^{-1}) \\ &= \mathcal{N}(\boldsymbol{\Gamma}^{-1}\boldsymbol{X}\boldsymbol{\Sigma}^{-1}\boldsymbol{Y}, \boldsymbol{\Gamma}^{-1}) \text{ for common case } \boldsymbol{\mu} = \mathbf{0}\end{aligned}$$

- And the predictive distribution is (applying the previous slide again):

$$\begin{aligned}\Pr(\boldsymbol{Y}_*|\boldsymbol{X}_*, \boldsymbol{X}, \boldsymbol{Y}) &= \int \Pr(\boldsymbol{Y}_*|\boldsymbol{\theta}, \boldsymbol{X}_*, \boldsymbol{X}, \boldsymbol{Y}) \Pr(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{Y}) d\boldsymbol{\theta} \\ &\sim \mathcal{N}(\boldsymbol{X}_*^\top\boldsymbol{\Gamma}^{-1}\boldsymbol{X}\boldsymbol{\Sigma}^{-1}\boldsymbol{Y}, \boldsymbol{X}_*^\top\boldsymbol{\Gamma}^{-1}\boldsymbol{X}_*)\end{aligned}$$

Beyond Linear Regression: basis functions

- Before we had $\boldsymbol{Y} = \boldsymbol{X}^\top \boldsymbol{\theta}$, but now we'd prefer a more flexible scheme $\boldsymbol{Y} = \phi(\boldsymbol{X})^\top \boldsymbol{\theta}$ where ϕ represent a function basis such as polynomials $\phi(\boldsymbol{X}) = (1, x, x^2, x^3)^\top$. The matrix $\Phi(\boldsymbol{X})$ is the aggregation of columns $\phi(\boldsymbol{X})$ into a matrix.
- Amazingly, the same analysis works, so that the predictive distribution becomes:

$$\Pr(\boldsymbol{Y}_*|\boldsymbol{X}_*, \boldsymbol{X}, \boldsymbol{Y}) \sim \mathcal{N}(\Phi(\boldsymbol{X}_*)^\top \boldsymbol{\Gamma}^{-1} \Phi(\boldsymbol{X}) \boldsymbol{\Sigma}^{-1} \boldsymbol{Y}, \Phi(\boldsymbol{X}_*)^\top \boldsymbol{\Gamma}^{-1} \Phi(\boldsymbol{X}_*))$$

  where $\Gamma = \boldsymbol{\Lambda}^{-1} + \Phi(\boldsymbol{X})^\top \Sigma^{-1} \Phi(\boldsymbol{X})$
- This means we can find predictive distribution for basis functions. But what if we don't want to specify any functional form for these functions?

Gaussian process

- A Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution.
- A Gaussian process is a distribution over functions, rather than over variables
- We define the mean function $m(\boldsymbol{x})$ and the covariance function $k(\boldsymbol{x}, \boldsymbol{x'})$ of a real process $f(\boldsymbol{x})$ and we note $\boldsymbol{f} \sim \mathcal{GP}(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x'}))$ so that:

$$\begin{aligned} m(\boldsymbol{x}) &= E[f(\boldsymbol{x})] \\ k(\boldsymbol{x}, \boldsymbol{x'}) &= E\left[(f(\boldsymbol{x}) - m(\boldsymbol{x}))\left(f(\boldsymbol{x'}) - m(\boldsymbol{x'})\right)\right] \end{aligned}$$

The squared exponential kernel

- The squared exponential is a classic choice for a first exposition to Gaussian processes

$$\mathbf{\Sigma} = cov(f(x), f(x')) = k(x, x') = \sigma_k^2 e^{-\frac{1}{2}\frac{(x-x')^2}{l^2}}$$

- It is parametrized by hyperparameter l, the characteristic length-scale of the process over which correlation is strong, and σ_k, the strength of correlation.
- Samples $\boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{M}, \boldsymbol{K})$ can be generated by :
  1. computing $\mathbf{\Sigma}$ using the kernel expression over a given range for $\boldsymbol{X}$.
  2. computing the Cholesky decomposition $\boldsymbol{L}$ of $\mathbf{\Sigma} = \boldsymbol{L}\boldsymbol{L}^\top$
  3. computing $\boldsymbol{Y} = \boldsymbol{m} + \boldsymbol{L}\boldsymbol{u}$ where $\boldsymbol{u} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
- Demo:

`https://distill.pub/2019/visual-exploration-gaussian-processes/`

Predictions: posterior

- Typical example is data $\boldsymbol{Y} = f(\boldsymbol{X}) + \boldsymbol{\epsilon}$, with $\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \boldsymbol{I})$.
- $\text{cov}(y_i, y_j) = k(x_i, y_j) + \sigma_n^2 \delta_{ij} \rightarrow \text{cov}(\boldsymbol{Y}) = \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}$
- The joint distribution of the measurements and the predictions is:

$$\begin{bmatrix} \boldsymbol{Y} \\ \boldsymbol{f}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I} & \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}_*) \\ \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}) & \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}_*) \end{bmatrix}\right)$$

- The predictive distribution is: $\Pr(\boldsymbol{f}_* | \boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{X}_*) \sim \mathcal{N}\left(\bar{\boldsymbol{f}}_*, \text{cov}(\boldsymbol{f}_*)\right)$, with

$$\bar{\boldsymbol{f}}_* = \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}) \left[\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}\right]^{-1} \boldsymbol{Y}$$

$$\text{cov}(\boldsymbol{f}_*) = \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}_*) - \boldsymbol{K}(\boldsymbol{X}_*, \boldsymbol{X}) \left[\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}\right]^{-1} \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}_*)$$

Marginalization of kernel parameters

- The prior $\Pr(\boldsymbol{f}|\boldsymbol{X}) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}))$ and likelihood $\Pr(\boldsymbol{Y}|\boldsymbol{f}) \sim \mathcal{N}(\boldsymbol{f}(\boldsymbol{X}), \sigma_n^2 \boldsymbol{I})$ give the marginal likelihood over the function values $\Pr(\boldsymbol{Y}|\boldsymbol{X}) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I})$
- This implies the analytic expression:

$$\log Pr(\boldsymbol{Y}|\boldsymbol{X}) = -\frac{1}{2}\boldsymbol{Y}^\top \left[\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}\right]^{-1} \boldsymbol{Y} - \frac{1}{2}\log|\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X})| - \frac{n}{2}\log 2\pi$$

- For kernels with hyperparameters (e.g. σ_k and l for the squared exponential), one minimize this log-marginal likelihood with respect to hyperparameters
- This allows us to find the best parameters supported by the data, and in turn to refine our future predictions.

Going beyond this introduction

- "Gaussian Processes for Machine Learning", Rasmussen & Williams, PDF downloadable here: http://www.gaussianprocess.org/gpml/
- "Pattern Recognition and Machine Learning", Bishop, with examples by contributors in Matlab http://prml.github.io/ or Python https://github.com/ctgk/PRML