Learning with Gaussians

CPSC 440/550: Advanced Machine Learning

cs.ubc.ca/~dsuth/440/24w2

University of British Columbia, on unceded Musqueam land

2024-25 Winter Term 2 (Jan–Apr 2025)

Last time: Multivariate Gaussians

- Continuous density estimation, $d > 1$ with the multivariate Gaussian distribution

$$X \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \quad \text{means} \quad p(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{d}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^{\mathsf{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right)$$

- If $\boldsymbol{\Sigma}$ is a diagonal matrix, product of univariate normals
- $\boldsymbol{\mu}_j$ is $\mathbb{E}[X_j]$; $\boldsymbol{\Sigma}_{jj'}$ gives $\mathrm{Cov}(X_j, X_{j'})$
  - If $\mathrm{Cov}(X_j, X_{j'}) = 0$, then $X_j \perp\!\!\!\perp X_{j'}$ (for jointly-Gaussian variables)
- If $\boldsymbol{\Sigma}$ is singular, "degenerate" Gaussian: $v^{\mathsf{T}} x$ takes a constant value for some v
- $AX + b \sim \mathcal{N}(A\boldsymbol{\mu} + b, A\boldsymbol{\Sigma}A^{\mathsf{T}})$
  - Lets us sample based on $Z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  - Marginalizing: still normal, just ignore the other variables in $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$
  - Conditioning: $x \mid Z = \mathbf{z} \sim \mathcal{N}\left(\boldsymbol{\mu}_x + \boldsymbol{\Sigma}_{xz}\boldsymbol{\Sigma}_z^{-1}(\mathbf{z} - \boldsymbol{\mu}_z), \boldsymbol{\Sigma}_x - \boldsymbol{\Sigma}_{xz}\boldsymbol{\Sigma}_z^{-1}\boldsymbol{\Sigma}_{xz}^{\mathsf{T}}\right)$

- MLE: $\hat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}^{(i)}$, $\hat{\boldsymbol{\Sigma}} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{x}^{(i)} - \boldsymbol{\mu})(\mathbf{x}^{(i)} - \boldsymbol{\mu})^{\mathsf{T}}$

Outline

MAP estimation for mean

- For fixed Σ, conjugate prior for mean is a Gaussian:

$$x^{(i)} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \qquad \mu \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0) \quad \text{implies} \quad \boldsymbol{\mu} \mid \mathbf{X}, \boldsymbol{\Sigma} \sim \mathcal{N}(\boldsymbol{\mu}^+, \boldsymbol{\Sigma}^+),$$

  where

$$\boldsymbol{\Sigma}^+ = (n\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}_0^{-1})^{-1},$$
$$\boldsymbol{\mu}^+ = \Sigma^+(n\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_{\mathsf{MLE}} + \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0) \qquad \text{MAP estimate of } \mu$$

- In special case of $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$ and $\Sigma_0 = \frac{1}{\lambda}\mathbf{I}$, we get

$$\boldsymbol{\Sigma}^+ = \left(\frac{n}{\sigma^2}\mathbf{I} + \lambda\mathbf{I}\right)^{-1} = \frac{1}{\frac{n}{\sigma^2} + \lambda}\mathbf{I},$$
$$\boldsymbol{\mu}^+ = \boldsymbol{\Sigma}^+\left(\frac{n}{\sigma^2}\boldsymbol{\mu}_{\mathsf{MLE}} + \lambda\boldsymbol{\mu}_0\right)$$

- Posterior predictive is $\mathcal{N}(\boldsymbol{\mu}^+, \boldsymbol{\Sigma} + \boldsymbol{\Sigma}^+)$ – take product of $(n+2)$ then marginalize
  - Many Bayesian inference tasks have closed form

MAP Estimation in Multivariate Gaussian (Trace Regularization)

- A common MAP estimate for Σ is

$$\hat{\Sigma} = \mathbf{S} + \lambda \mathbf{I},$$

  where S is the covariance of the data
  - Key advantage: $\hat{\Sigma}$ is strictly positive definite (eigenvalues are at least λ)

- This corresponds to L1 regularization of precision diagonals (see bonus)

$$f(\Theta) = \underbrace{\mathrm{Tr}(\mathbf{S}\boldsymbol{\Theta}) - \log|\boldsymbol{\Theta}|}_{\text{NLL times } 2/n} + \lambda \sum_{j=1}^{d} |\boldsymbol{\Theta}_{jj}|$$

- Note this *doesn't* set Θ_{jj} values to exactly zero
  - Log-determinant term becomes arbitrarily steep as the Θ_{jj} approach 0
  - It's not really the case that "L1 gives sparsity"; it's "L2 + L1 gives sparsity"

Conjugate Priors for Covariance

bonus!

- Trace regularization : not a conjugate prior
- Conjugate prior for Θ with known mean is Wishart distribution
  - A multi-dimensional generalization of the gamma distribution
    - Gamma is a distribution over positive scalars
    - Wishart is a distribution over positive-definite matrices
  - Posterior predictive is a student t distribution
  - Conjugate prior for Σ is inverse-Wishart (equivalent posterior)
- If both μ and Θ are random, conjugate prior is normal-Wishart
  - Normal times Wishart, with a particular dependency among parameters
  - Posterior predictive is again a student t distribution
- Wikipedia has already done a lot of possible homework questions for you:
  - `https://en.wikipedia.org/wiki/Conjugate_prior`

Outline

Generative Classification with Gaussians

- Consider a generative classifier with continuous features:

$$p(y \mid x) \propto p(x, y) = \underbrace{p(x \mid y)}_{\text{continuous}} \underbrace{p(y)}_{\text{discrete}}$$

- Model y as a categorical distribution (classification task)
- Previously handled $p(x \mid y)$ with the naive Bayes assumption, $X_i \perp\!\!\!\perp X_j \mid Y$
  - Strong, usually unrealistic assumption

- In Gaussian discriminant analysis (GDA) we assume $X \mid Y$ is Gaussian
  - Classifier asks "which Gaussian makes this $\tilde{\mathbf{x}}$ most likely?"
  - This can model pairwise correlations within each class
    - Doesn't need the naive Bayes assumption

Gaussian Discriminant Analysis (GDA)

- In Gaussian discriminant analysis we assume $X \mid Y$ is Gaussian

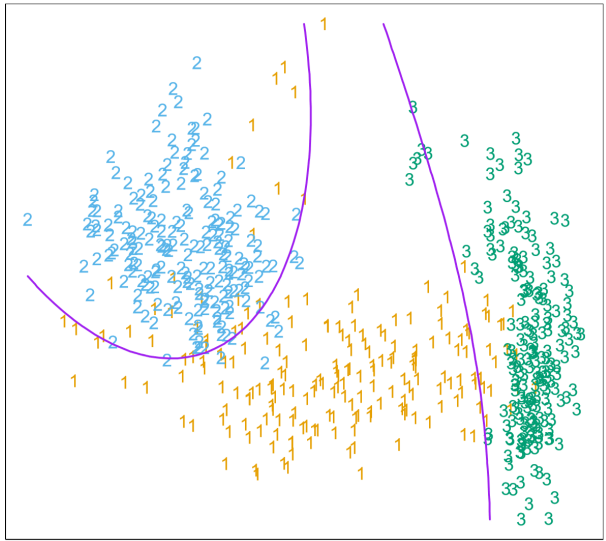
$$p(\mathbf{x}, y = c) = \underbrace{p(y)\, p(\mathbf{x} \mid y = c)}_{\text{product rule}} = \underbrace{\pi_c}_{\Pr(y=c)} \underbrace{p(\mathbf{x} \mid \boldsymbol{\mu_c}, \boldsymbol{\Sigma_c})}_{\text{Gaussian pdf}}$$

- Classify based on

$$\begin{aligned}\arg\max_c p(\tilde{y} = c \mid \tilde{\mathbf{x}}) &= \arg\max_c \log p(\tilde{y} = c, \tilde{\mathbf{x}}) \\ &= \arg\max_c \ \log \pi_c - \frac{1}{2} \log |\boldsymbol{\Sigma_c}| - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu_c})^\mathsf{T} \boldsymbol{\Sigma_c}^{-1} (\mathbf{x} - \boldsymbol{\mu_c})\end{aligned}$$

- With general choices for $\boldsymbol{\mu_c}$ and $\boldsymbol{\Sigma_c}$, we're taking the max of k quadratics
  - Means that the decision boundary will be zeros of a quadratic ("quadric surface")
  - Leads to the equivalent name quadratic discriminant analysis (QDA)
- Fitting GDA=QDA: fit π_c as categorical, fit Gaussian for each subset with $y^{(i)} = c$

GDA=QDA example



https://web.stanford.edu/~hastie/Papers/ESLII.pdf

Special case: Linear Discriminant Analysis (LDA)

- A common special case: constrain $\boldsymbol{\Sigma_c} = \boldsymbol{\Sigma}$ for all c
- Means that we classify as

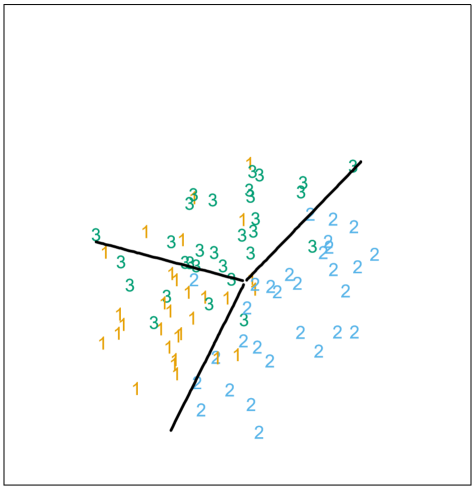
$$\begin{aligned}
\arg\max_c p(y = c \mid x) &= \arg\max_c \ \log \pi_c - \tfrac{1}{2} \log |\boldsymbol{\Sigma}| - \tfrac{1}{2} (\mathbf{x} - \boldsymbol{\mu_c})^\mathsf{T} \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu_c}) \\
&= \arg\max_c \ \log \pi_c - \tfrac{1}{2} \mathbf{x}^\mathsf{T} \boldsymbol{\Sigma}^{-1} \mathbf{x} + \boldsymbol{\mu_c}^\mathsf{T} \boldsymbol{\Sigma}^{-1} \mathbf{x} - \tfrac{1}{2} \boldsymbol{\mu_c}^\mathsf{T} \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu_c} \\
&= \arg\max_c \ \underbrace{(\boldsymbol{\Sigma}^{-1} \boldsymbol{\mu_c})^\mathsf{T}}_{w_c} \mathbf{x} + \underbrace{\log \pi_c - \tfrac{1}{2} \boldsymbol{\mu_c}^\mathsf{T} \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu_c}}_{b_c}
\end{aligned}$$

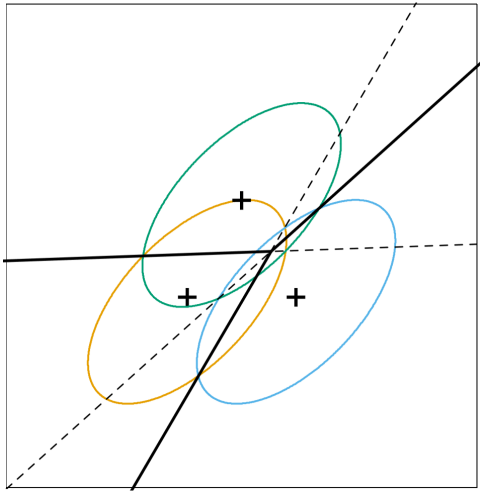
  so this is a linear classifier!
  - Behaves (asymptotically) optimally if the assumptions are true: $X \mid y \sim \mathcal{N}(\boldsymbol{\mu}_y, \boldsymbol{\Sigma})$
  - May be terrible if these assumptions aren't true
- MLE in this model is simple: $\boldsymbol{\mu_c}$ is mean of the points with $y^{(i)} = c$,

$$\boldsymbol{\Sigma} \text{ is } \frac{1}{n} \sum_{i=1}^{n} \left(\mathbf{x}^{(i)} - \boldsymbol{\mu}_{y^{(i)}} \right) \left(\mathbf{x}^{(i)} - \boldsymbol{\mu}_{y^{(i)}} \right)^\mathsf{T}$$

LDA example

- Example of fitting linear discriminant analysis (LDA) to a 3-class problem:





https://web.stanford.edu/~hastie/Papers/ESLII.pdf

LDA and nearest neighbour

bonus!

- LDA classifies according to

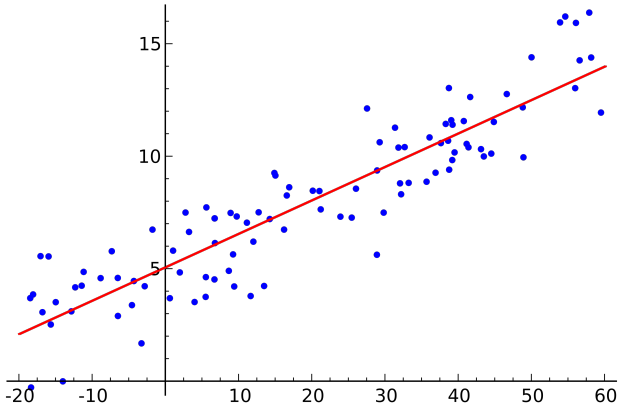
$$\begin{aligned}
&\arg\max_c \ (\boldsymbol{\Sigma}^{-\frac{1}{2}}\boldsymbol{\mu}_c)^\mathsf{T}(\boldsymbol{\Sigma}^{-\frac{1}{2}}\mathbf{x}) - \tfrac{1}{2}(\boldsymbol{\Sigma}^{-\frac{1}{2}}\boldsymbol{\mu}_c)^\mathsf{T}(\boldsymbol{\Sigma}^{-\frac{1}{2}}\boldsymbol{\mu}_c) + \log \pi_c \\
&\qquad = \arg\max_c \ -\tfrac{1}{2}\|\boldsymbol{\Sigma}^{-\frac{1}{2}}\mathbf{x}\|^2 + (\boldsymbol{\Sigma}^{-\frac{1}{2}}\boldsymbol{\mu}_c)^\mathsf{T}(\boldsymbol{\Sigma}^{-\frac{1}{2}}\mathbf{x}) - \tfrac{1}{2}\|\boldsymbol{\Sigma}^{-\frac{1}{2}}\boldsymbol{\mu}_c\|^2 + \log \pi_c \\
&\qquad = \arg\min_c \|\boldsymbol{\Sigma}^{-\frac{1}{2}}(\mathbf{x} - \boldsymbol{\mu}_c)\|^2 - 2\log \pi_c
\end{aligned}$$

- If π_c are constant (all $\frac{1}{k}$) and $\boldsymbol{\Sigma} = \sigma^2 \mathbf{I}$, this picks the closest class mean
- With constant π_c but general $\boldsymbol{\Sigma}$, picks closest class mean in Mahalanobis distance

Outline

Regression with Gaussians

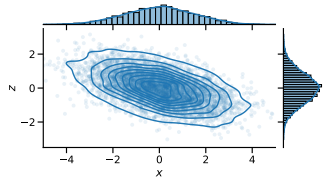
- In regression, Y is continuous



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

Generative Regression with Multivariate Normal

- With continuous features, we could model $p(x, y)$ as a multivariate Gaussian



- Training could use the closed-form MLE/MAP for multivariate Gaussian
- We obtain a univariate Gaussian $p(y \mid x)$ using conditioning formula,

$$Y \mid X = \mathbf{x} \sim \mathcal{N}\left(\boldsymbol{\mu}_y + \boldsymbol{\Sigma}_{yx}\boldsymbol{\Sigma}_x^{-1}(\mathbf{x} - \boldsymbol{\mu}_x), \sigma_y^2 - \boldsymbol{\Sigma}_{yx}\boldsymbol{\Sigma}_x^{-1}\boldsymbol{\Sigma}_{yx}^{\mathsf{T}}\right)$$

- The conditional mean is a linear function, $w^{\mathsf{T}}\mathbf{x} + b$
- Could extend to multiple outputs, with correlations given based on $\boldsymbol{\Sigma}_y$

- Problem: what if X isn't really normal?

Bayesian Linear Regression

- Linear regression with Gaussian likelihood and prior,

$$Y \mid X = \mathbf{x} \sim \mathcal{N}(w^\mathsf{T}\mathbf{x}, \sigma^2) \qquad w \sim \mathcal{N}(0, \lambda^{-1}\mathbf{I})$$

- MAP estimate is ridge regression (L2-regularized least squares)
- Can use Gaussian identities to work out that the posterior has the form

$$w \mid (\mathbf{X}, \mathbf{y}) \sim \mathcal{N}\left(w_{\mathsf{MAP}}, \left(\frac{1}{\sigma^2}\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda\mathbf{I}\right)^{-1}\right),$$

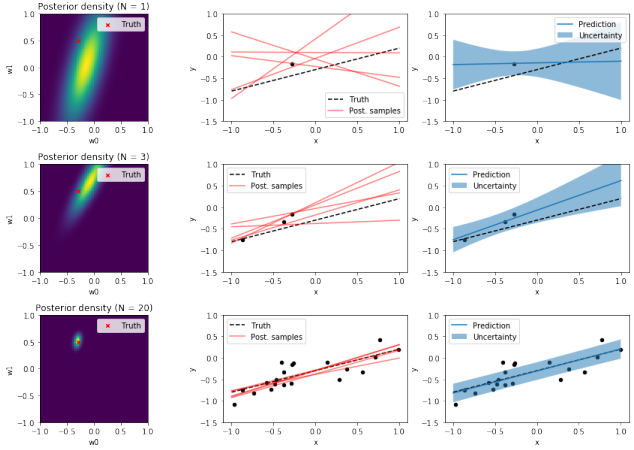
  which is a multivariate Gaussian centred at $w_{\mathsf{MAP}} = \left(\mathbf{X}^\mathsf{T}\mathbf{X} + \frac{\lambda}{\sigma^2}\mathbf{I}_d\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}$
  - The variance tells us how much variation we have around the MAP estimate
    - In other models, the posterior mode (MAP) is often not the posterior mean
- By more Gaussian identities, the posterior predictive has the form

$$\tilde{y} \mid (\mathbf{X}, \mathbf{y}, \tilde{x}) \sim \mathcal{N}\left(w_{\mathsf{MAP}}^\mathsf{T}\tilde{x},\ \sigma^2 + \tilde{x}^\mathsf{T}\left(\frac{1}{\sigma^2}\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda\mathbf{I}\right)^{-1}\tilde{x}\right)$$

- Posterior predictive mode=mean again the MAP prediction in this model
  - Working with the full posterior predictive gives us variance of predictions

Bayesian Linear Regression

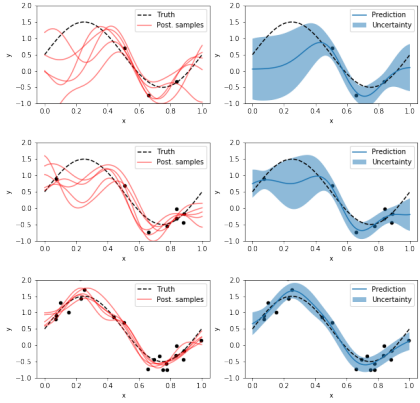
- Bayesian perspective gives us variability in w and predictions:



http://krasserm.github.io/2019/02/23/bayesian-linear-regression

Bayesian Linear Regression

- Bayesian linear regression with Gaussian RBFs as features:



http://krasserm.github.io/2019/02/23/bayesian-linear-regression

- We have not only a prediction, but Bayesian inference gives "error bars"
  - Gives an idea of "where model is confident" and where it is not

Digression: Gaussian Processes

bonus!

- In CPSC 340 you saw the kernel trick:
  - Rewrites L2-regularized least squares linear/prediction in terms of inner products
  - Allows us to efficiently use some exponential-sized or infinite-sized feature sets

- We can use kernel trick on posterior in Gaussian likelihood/prior model
  - Allows us to efficiently use some large or infinite-sized feature sets
  - Posterior in this case can be written as a Gaussian process (GP)

- Notation: a stochastic process is an infinite collection of random variables
- In a Gaussian process, any finite subcollection is jointly Gaussian
  - Defined in terms of a mean function and a covariance function
    - The set of possible covariance functions is the set of possible kernel functions
  - A popular book on this topic if you want to read more:
    Rasmussen/Williams, Gaussian Processes for Machine Learning

- We'll assume we have explicit features, but you could use kernels/GPs instead

Summary

- Gaussian discriminant analysis and special case linear discriminant analysis
  - Generative classifier where $x \mid y$ is multivariate normal
- Bayesian Linear Regression
  - Gaussian conditional likelihood and Gaussian prior gives Gaussian posterior
  - Posterior predictive is also Gaussian ("regression with error bars")

- Next time: choosing priors