CSC 311: Introduction to Machine Learning

Lecture 8 - Probabilistic Models Pt. II, PCA

Roger Grosse Chris Maddison Juhan Bae Silviu Pitis

University of Toronto, Fall 2020

Recap

- Last week took a probabilistic perspective on parameter estimation.
- We modeled a biased coin as a Bernoulli random variable with parameter θ, which we estimated using:
    - maximum likelihood estimation:
      $\hat{\theta}_{\mathrm{ML}} = \max_{\theta} p(\mathcal{D} \,|\, \theta)$
    - expected Bayesian posterior:
      $\mathbb{E}[\theta \,|\, \mathcal{D}]$ where $p(\theta \,|\, \mathcal{D}) \propto p(\boldsymbol{\theta}) p(\mathcal{D} \,|\, \theta)$ by Bayes' Rule.
    - Maximum a-posteriori (MAP) estimation:
      $\hat{\theta}_{\mathrm{MAP}} = \arg\max_{\theta} \; p(\theta \,|\, \mathcal{D})$
- We also saw parameter estimation in context of a Naïve Bayes classifier.
- Today we will continuing developing the probabilistic perspective:
    - Gaussian Discriminant Analysis: Use Gaussian generative model of the data for classification
    - Principal Component Analysis: Simplify a Gaussian model by projected it onto a lower dimensional subspace

Gaussian Discriminant Analysis

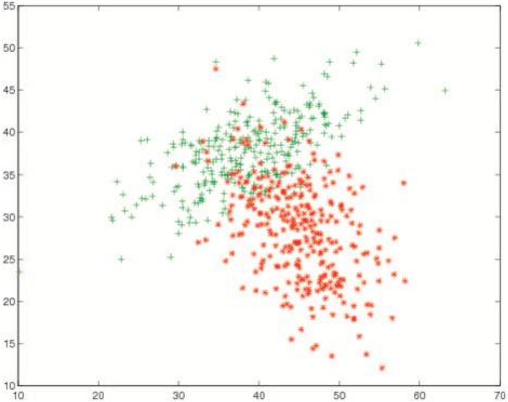
- Generative model for classification
- Instead of trying to separate classes, try to model what each class "looks like": $p(\mathbf{x} \,|\, t = k)$.
- Recall $p(\mathbf{x} \,|\, t = k)$ may be very complex for high dimensional data:

$$p(x_1, \cdots, x_d, t) = p(x_1|x_2, \cdots, x_d, t) \cdots p(x_{d-1}|x_d, t)p(x_d, t)$$

- Naive bayes used a conditional independence assumption. What else could we do? Choose a simple distribution.
- Next, we will discuss fitting Gaussian distributions to our data.

Classification: Diabetes Example

- Observation per patient: White blood cell count & glucose value.



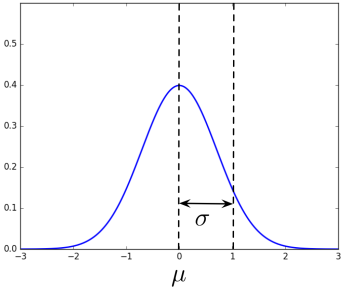
- $p(\mathbf{x} \mid t = k)$ for each class is shaped like an ellipse
  $\Longrightarrow$ we model each class as a multivariate Gaussian

Univariate Gaussian distribution

- Recall the Gaussian, or normal, distribution:

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

- The Central Limit Theorem says that sums of lots of independent random variables are approximately Gaussian.
- In machine learning, we use Gaussians a lot because they make the calculations easy.



Multivariate Data

- Multiple measurements (sensors)
- D inputs/features/attributes
- N instances/observations/examples

$$\mathbf{X} = \begin{bmatrix} [\mathbf{x}^{(1)}]^\top \\ [\mathbf{x}^{(2)}]^\top \\ \vdots \\ [\mathbf{x}^{(N)}]^\top \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_D^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdots & x_D^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{(N)} & x_2^{(N)} & \cdots & x_D^{(N)} \end{bmatrix}$$

Multivariate Mean and Covariance

- Mean

$$\boldsymbol{\mu} = \mathbb{E}[\mathbf{x}] = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_d \end{pmatrix}$$

- Covariance

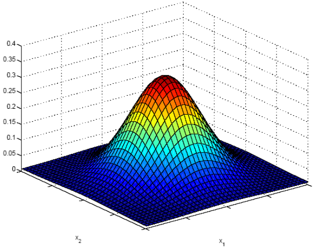
$$\boldsymbol{\Sigma} = \mathrm{Cov}(\mathbf{x}) = \mathbb{E}[(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^\top] = \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1D} \\ \sigma_{12} & \sigma_2^2 & \cdots & \sigma_{2D} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{D1} & \sigma_{D2} & \cdots & \sigma_D^2 \end{pmatrix}$$

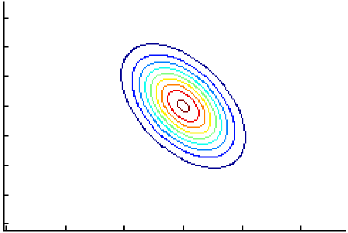
- The statistics ($\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$) uniquely define a multivariate Gaussian (or multivariate Normal) distribution, denoted $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ or $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$
  - This is not true for distributions in general!

Multivariate Gaussian Distribution

- Normally distributed variable $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ has distribution:

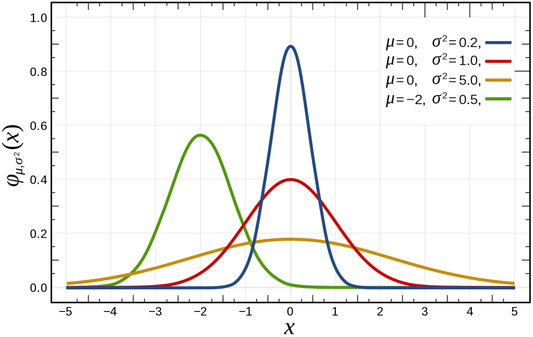
$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$





Gaussian Intuition: (Multivariate) Shift + Scale

- Recall that in the univariate case, all normal distributions are shaped like the standard normal distribution
- The densities are related to the standard normal by a shift (μ), a scale (or stretch, or dilation) σ, and a normalization factor



Gaussian Intuition: (Multivariate) Shift + Scale

- The same intuition applies in the multivariate case.
- We can think of the multivariate Gaussian as a shifted and "scaled" version of the standard multivariate normal distribution.
    - The standard multivariate normal has $\boldsymbol{\mu} = \mathbf{0}$ and $\boldsymbol{\Sigma} = \mathbf{I}$
- Multivariate analog of the shift is simple: it's a vector $\boldsymbol{\mu}$
- But what about the scale?
    - In the univariate case, the scale factor was the square root of the variance: $\sigma = \sqrt{\sigma^2}$
    - But in the multivariate case, the covariance $\boldsymbol{\Sigma}$ is a matrix! Does $\boldsymbol{\Sigma}^{\frac{1}{2}}$ exist, and can we scale by it?

Multivariate Scaling (Intuitive)

(optional draw-on slide for intuition)

We call a matrix “positive definite” if it scales the space in **orthogonal** directions. The univariate analog is positive scalar $\alpha > 0$.
Consider, e.g., how these two matrices transform the orthogonal vectors:

Consider matrix:	$\begin{pmatrix} 2 & 0 \\ 0 & 0.5 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$
Consider action on:	$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \perp \begin{pmatrix} 0 \\ 1 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 1 \end{pmatrix} \perp \begin{pmatrix} 1 \\ -1 \end{pmatrix}$
Draw action on slide:		

Notice: both matrices are symmetric!

Multivariate Scaling (Formal) (details optional)

We summarize some definitions/results from linear algebra (without proof). Knowing them is optional, but they may help with intuition (esp. for PCA).

- **Definition.** Symmetric matrix A is positive semidefinite if $\mathbf{x}^\top \mathbf{A}\mathbf{x} \geq 0$ for all non-zero $\mathbf{x}$. It is positive definite if $\mathbf{x}^\top \mathbf{A}\mathbf{x} > 0$ for all non-zero $\mathbf{x}$.
  - Any positive definite matrix is positive semidefinite.
  - Positive definite matrices have positive eigenvalues, and positive semidefinite matrices have non-negative eigenvalues.
  - For any matrix $\mathbf{X}$, $\mathbf{X}^\top \mathbf{X}$ and $\mathbf{X}\mathbf{X}^\top$ are positive semidefinite.
- **Theorem** (*Unique Positive Square Root*)**.** Let $\mathbf{A}$ be a positive semidefinite real matrix. Then there is a unique positive semidefinite matrix $\mathbf{B}$ such that $\mathbf{A} = \mathbf{B}^\top \mathbf{B} = \mathbf{B}\mathbf{B}$. We call $\mathbf{A}^{\frac{1}{2}} \triangleq \mathbf{B}$ the positive square root of $\mathbf{A}$.
- **Theorem** (*Spectral Theorem*)**.** The following are equivalent for $\mathbf{A} \in \mathbb{R}^{d \times d}$:
  1. $\mathbf{A}$ is symmetric.
  2. $\mathbb{R}^D$ has an orthonormal basis consisting of the eigenvectors of $\mathbf{A}$.
  3. There exists orthogonal matrix $\mathbf{Q}$ and diagonal matrix $\mathbf{\Lambda}$ such that $\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$. This is called the spectral decomposition of $\mathbf{A}$.
     - The columns of $\mathbf{Q}$ are (unit) eigenvectors of $\mathbf{A}$.

Properties of $\boldsymbol{\Sigma}$

Key properties of $\boldsymbol{\Sigma}$:

1. $\boldsymbol{\Sigma}$ is positive semidefinite (and therefore symmetric).
2. For a distribution with density, $\boldsymbol{\Sigma}$ is positive definite.

Other properties (optional / for reference):

3. $\boldsymbol{\Sigma} = \mathbb{E}[\mathbf{x}\mathbf{x}^\top] - \boldsymbol{\mu}\boldsymbol{\mu}^\top$ (generalizes $\mathrm{Var}(x) = \mathbb{E}[x^2] - \mu^2$))
4. $\mathrm{Cov}(\mathbf{A}\mathbf{x} + \mathbf{b}) = \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}^\top$ (generalizes $\mathrm{Var}(ax + b) = a^2 \,\mathrm{Var}(x)$))

So here is the "scale" intuition:

- For positive definite $\boldsymbol{\Sigma}$, consider its unique positive square root $\boldsymbol{\Sigma}^{\frac{1}{2}}$.
- $\boldsymbol{\Sigma}^{\frac{1}{2}}$ is also positive definite, so by the Real Spectral Theorem, it "scales" the space in orthogonal directions (its eigenvectors) by its eigenvalues.
- So we can think of $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ as $\mathcal{N}(\mathbf{0}, \mathbf{I})$ shifted by $\boldsymbol{\mu}$ and "scaled" by $\boldsymbol{\Sigma}^{\frac{1}{2}}$!
  - Note that if $\boldsymbol{\Sigma} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^T$, $\boldsymbol{\Sigma}^{\frac{1}{2}} = \mathbf{Q}\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{Q}^T$
- Lets see some examples...

Bivariate Gaussian

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \boldsymbol{\Sigma} = 0.5 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \boldsymbol{\Sigma} = 2 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

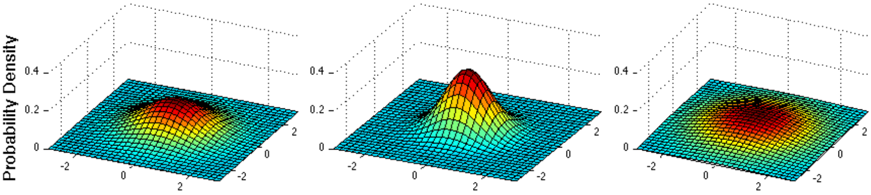


Figure: Probability density function

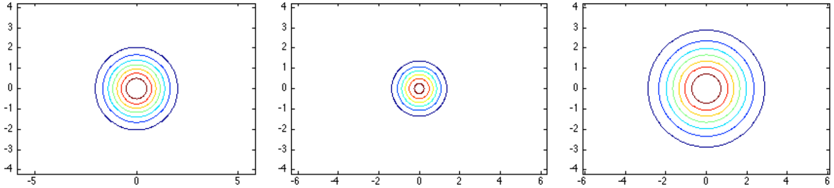


Figure: Contour plot of the pdf

Bivariate Gaussian

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$$

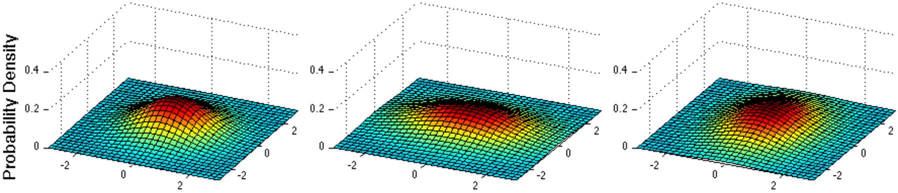


Figure: Probability density function

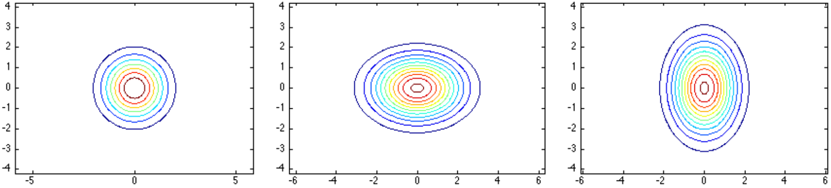


Figure: Contour plot of the pdf

Bivariate Gaussian

$$\mathbf{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \mathbf{\Sigma} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix} \qquad \mathbf{\Sigma} = \begin{pmatrix} 1 & 0.8 \\ 0.8 & 1 \end{pmatrix}$$

$$= \mathbf{Q}_1 \begin{pmatrix} 1.5 & 0. \\ 0. & 0.5 \end{pmatrix} \mathbf{Q}_1^\top \qquad = \mathbf{Q}_2 \begin{pmatrix} 1.8 & 0. \\ 0. & 0.2 \end{pmatrix} \mathbf{Q}_2^\top$$

Test your intuition: Does $Q_1 = Q_2$?

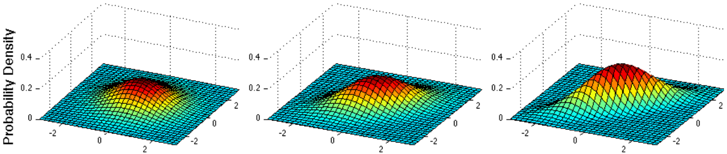


Figure: Probability density function

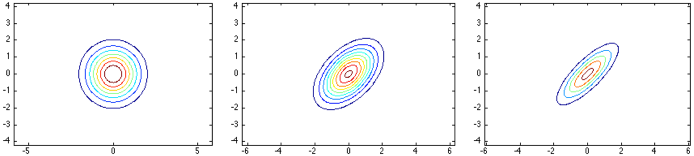


Figure: Contour plot of the pdf

Bivariate Gaussian

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & -0.5 \\ -0.5 & 1 \end{pmatrix}$$

$$= \mathbf{Q}_1 \begin{pmatrix} 1.5 & 0. \\ 0. & 0.5 \end{pmatrix} \mathbf{Q}_1^\top \qquad = \mathbf{Q}_2 \begin{pmatrix} \lambda_1 & 0. \\ 0. & \lambda_2 \end{pmatrix} \mathbf{Q}_2^\top$$

Test your intuition: Does $Q_1 = Q_2$? What are λ_1 and λ_2?

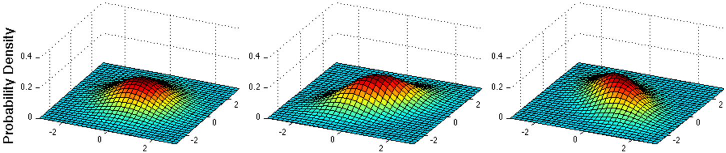


Figure: Probability density function

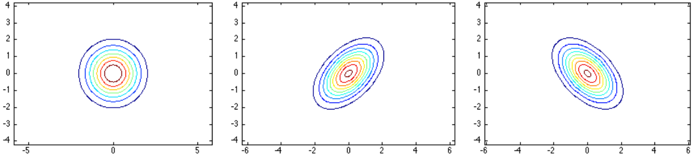
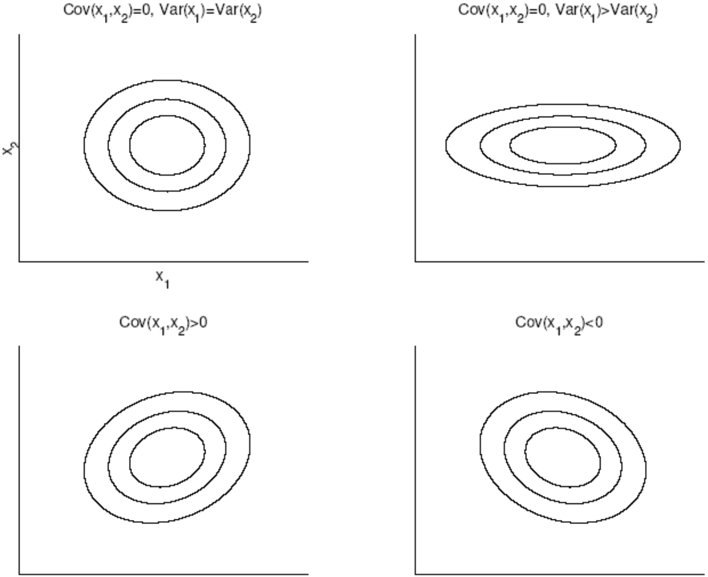
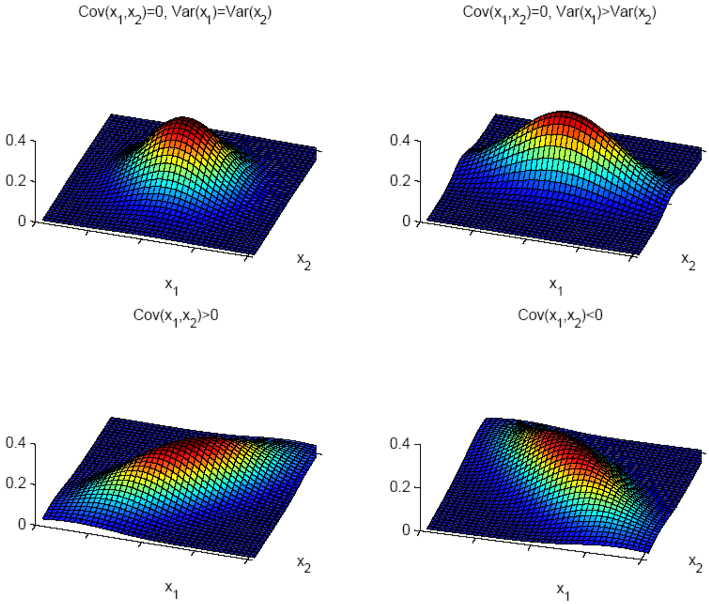


Figure: Contour plot of the pdf

Bivariate Gaussian



Bivariate Gaussian



$\mathrm{Cov}(x_1,x_2)=0$, $\mathrm{Var}(x_1)=\mathrm{Var}(x_2)$

$\mathrm{Cov}(x_1,x_2)=0$, $\mathrm{Var}(x_1)>\mathrm{Var}(x_2)$

$\mathrm{Cov}(x_1,x_2)>0$

$\mathrm{Cov}(x_1,x_2)<0$

Gaussian Maximum Likelihood

- Suppose we want to model the distribution of highest and lowest temperatures in Toronto in March, and we've recorded the following observations ☹

  (-2.5,-7.5)   (-9.9,-14.9)   (-12.1,-17.5)   (-8.9,-13.9)   (-6.0,-11.1)

- Assume they're drawn from a Gaussian distribution with mean $\boldsymbol{\mu}$, and covariance $\boldsymbol{\Sigma}$. We want to estimate these using data.
- Log-likelihood function:

$$\begin{aligned}\ell(\boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \log \prod_{i=1}^{N} \left[\frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{ -\frac{1}{2}(\mathbf{x}^{(i)} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu}) \right\} \right] \\ &= \sum_{i=1}^{N} \log \left[\frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{ -\frac{1}{2}(\mathbf{x}^{(i)} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu}) \right\} \right] \\ &= \sum_{i=1}^{N} \underbrace{-\log (2\pi)^{d/2}}_{\text{constant}} - \log |\boldsymbol{\Sigma}|^{1/2} - \frac{1}{2}(\mathbf{x}^{(i)} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu})\end{aligned}$$

Optional intuition building: why does $|\boldsymbol{\Sigma}|^{1/2}$ show up in the Gaussian density $p(\mathbf{x})$?

Hint: determinant is product of eigenvalues

Gaussian Maximum Likelihood

- Maximize the log-likelihood by setting the derivative to zero:

$$\begin{aligned} 0 = \frac{\mathrm{d}\ell}{\mathrm{d}\boldsymbol{\mu}} &= -\sum_{i=1}^{N} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\mu}} \frac{1}{2}(\mathbf{x}^{(i)} - \boldsymbol{\mu})^{T} \boldsymbol{\Sigma}^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu}) \\ &= -\sum_{i=1}^{N} \boldsymbol{\Sigma}^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu}) = 0 \end{aligned}$$

- Here we use the identity $\nabla_{\mathbf{x}} \mathbf{x}^{\top} \mathbf{A} \mathbf{x} = 2\mathbf{A}\mathbf{x}$ (see the multivariable calculus note from Lecture 2).
- Solving we get $\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}^{(i)}$. In general, “hat” means estimator
- This is just the sample mean of the observed values, or the empirical mean.

Gaussian Maximum Likelihood

- We can do a similar calculation for the covariance matrix Σ (we skip the details).
- Setting the *partial* derivatives to zero, just like before, we get:

$$
\begin{aligned}
0 = \frac{\partial \ell}{\partial \boldsymbol{\Sigma}} \implies \hat{\boldsymbol{\Sigma}} &= \frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}})(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}})^\top \\
&= \frac{1}{N} (\mathbf{X} - \mathbf{1}\boldsymbol{\mu}^\top)^\top (\mathbf{X} - \mathbf{1}\boldsymbol{\mu}^\top)
\end{aligned}
$$

  where $\mathbf{1}$ is an N-dimensional vector of 1s.

- This is called the empirical covariance and comes up quite often (e.g., PCA soon!)
- Derivation in multivariate case is tedious. No need to worry about it. But it is good practice to derive this in one dimension. See supplement (next slide).

Supplement: MLE for univariate Gaussian

$$
\begin{aligned}
0 = \frac{\partial \ell}{\partial \mu} &= -\frac{1}{\sigma^2} \sum_{i=1}^{N} \mathbf{x}^{(i)} - \mu \\
0 = \frac{\partial \ell}{\partial \sigma} &= \frac{\partial}{\partial \sigma} \left[\sum_{i=1}^{N} -\frac{1}{2} \log 2\pi - \log \sigma - \frac{1}{2\sigma^2} (\mathbf{x}^{(i)} - \mu)^2 \right] \\
&= \sum_{i=1}^{N} -\frac{1}{2} \frac{\partial}{\partial \sigma} \log 2\pi - \frac{\partial}{\partial \sigma} \log \sigma - \frac{\partial}{\partial \sigma} \frac{1}{2\sigma} (\mathbf{x}^{(i)} - \mu)^2 \\
&= \sum_{i=1}^{N} 0 - \frac{1}{\sigma} + \frac{1}{\sigma^3} (\mathbf{x}^{(i)} - \mu)^2 \\
&= -\frac{N}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^{N} (\mathbf{x}^{(i)} - \mu)^2
\end{aligned}
$$

$$
\begin{aligned}
\hat{\mu}_{\text{ML}} &= \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}^{(i)} \\
\hat{\sigma}_{\text{ML}} &= \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}^{(i)} - \mu)^2}
\end{aligned}
$$

Gaussian Discriminant Analysis (Gaussian Bayes Classifier)

- Gaussian Discriminant Analysis in its general form assumes that $p(\mathbf{x} \,|\, t)$ is distributed according to a multivariate Gaussian distribution
- Multivariate Gaussian distribution:

$$p(\mathbf{x} \,|\, t = k) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_k|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k)\right]$$

  where $|\boldsymbol{\Sigma}_k|$ is the determinant of $\boldsymbol{\Sigma}_k$, and d is dimension of $\mathbf{x}$

- Each class k has a mean vector $\boldsymbol{\mu}_k$ and a covariance matrix $\boldsymbol{\Sigma}_k$
- Each $\boldsymbol{\Sigma}_k$ has $\mathcal{O}(d^2)$ parameters - could be hard to estimate

Gaussian Discriminant Analysis (Gaussian Bayes Classifier)

- GDA (GBC) decision boundary is based on class posterior.
- Make decisions by comparing class probabilities:

$$
\begin{aligned}
\log p(t_k|\mathbf{x}) &= \log p(\mathbf{x}|t_k) + \log p(t_k) - \log p(\mathbf{x}) \\
&= -\frac{d}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}_k^{-1}| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k) \\
&\quad + \log p(t_k) - \log p(\mathbf{x})
\end{aligned}
$$

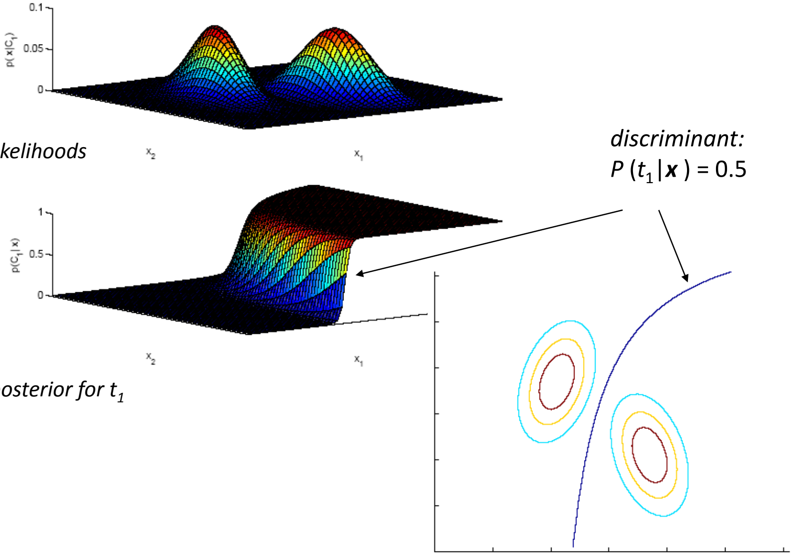
- Decision boundary ($\log p(t_k|\mathbf{x}) = \log p(t_l|\mathbf{x})$):

$$(\mathbf{x}-\boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k) = (\mathbf{x}-\boldsymbol{\mu}_\ell)^T\boldsymbol{\Sigma}_\ell^{-1}(\mathbf{x}-\boldsymbol{\mu}_\ell) + C_{k,l}$$

$$\mathbf{x}^T\boldsymbol{\Sigma}_k^{-1}\mathbf{x} - 2\boldsymbol{\mu}_k^T\boldsymbol{\Sigma}_k^{-1}\mathbf{x} = \mathbf{x}^T\boldsymbol{\Sigma}_\ell^{-1}\mathbf{x} - 2\boldsymbol{\mu}_\ell^T\boldsymbol{\Sigma}_\ell^{-1}\mathbf{x} + C_{k,l}$$

- Quadratic relation in $\mathbf{x} \implies$ quadratic (conic) decision boundary
- So sometimes called "Quadratic Discriminant Analysis" (QDA)

Decision Boundary



likelihoods

posterior for t_1

discriminant:
$P(t_1|\boldsymbol{x}) = 0.5$

Learning

- Learn the parameters for each class using maximum likelihood
- Assume the prior is Bernoulli (we have two classes)

$$p(t|\phi) = \phi^t (1-\phi)^{1-t}.$$

- You can compute the MLE in closed form (good exercise!)

$$\begin{aligned}
\hat{\phi} &= \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}[t^{(n)} = 1] \\
\hat{\boldsymbol{\mu}}_k &= \frac{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = k] \cdot \mathbf{x}^{(n)}}{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = k]} \\
\hat{\boldsymbol{\Sigma}}_k &= \frac{1}{\sum_{n=1}^{N} \mathbb{1}[t^{(n)} = k]} \sum_{n=1}^{N} \mathbb{1}[t^{(n)} = k](\mathbf{x}^{(n)} - \hat{\mu}_{t^{(n)}})(\mathbf{x}^{(n)} - \hat{\mu}_{t^{(n)}})^T
\end{aligned}$$

Simplifying the Model

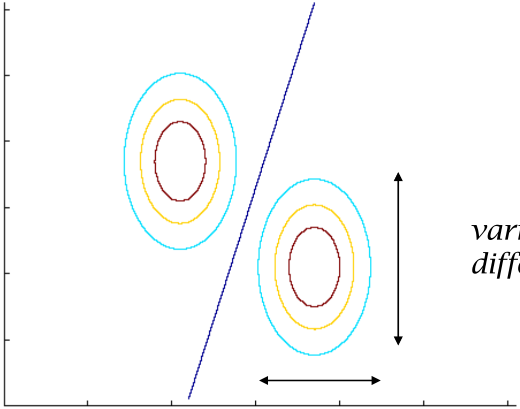
What if $\mathbf{x}$ is high-dimensional?

- For Gaussian Bayes Classifier, if input $\mathbf{x}$ is high-dimensional, then covariance matrix has many parameters $O(d^2)$
- Save some parameters by using a shared covariance for the classes, i.e. $\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}_l$.
- Any other idea you can think of? (next lecture)
- MLE in this case:

$$\hat{\Sigma} = \frac{1}{N} \sum_{n=1}^{N} (\mathbf{x}^{(n)} - \mu_{t^{(n)}})(\mathbf{x}^{(n)} - \mu_{t^{(n)}})^T$$

- Linear decision boundary (at home: verify this mathematically!).
  - In Scikit-Learn this is called "Linear Discriminant Analysis" (LDA)

Decision Boundary: Shared Variances (between Classes)



variances may be different

Gaussian Discriminative Analysis vs Logistic Regression

- Binary classification: If you examine $p(t = 1|\mathbf{x})$ under GDA and assume $\Sigma_0 = \Sigma_1 = \Sigma$, you will find that it looks like this:

$$p(t|\mathbf{x}, \phi, \mu_0, \mu_1, \Sigma) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

  where $\mathbf{w}$ is an appropriate function of $(\phi, \mu_0, \mu_1, \Sigma)$, $\phi = p(t = 1)$.
- GDA is similar to logistic regression (LR), parameter estimates are computed differently.
- When should we prefer GDA to LR, and vice versa?

Gaussian Discriminative Analysis vs Logistic Regression

- GDA is a generative model, LR is a discriminative model.
- GDA makes stronger modeling assumption: assumes class-conditional data is multivariate Gaussian.
- If this is true, GDA is asymptotically efficient (best model in limit of large N)
- But LR is more robust, less sensitive to incorrect modeling assumptions (what loss is it optimizing?)
- Many class-conditional distributions lead to logistic classifier.
- When these distributions are non-Gaussian (true almost always), LR usually beats GDA

Generative models - Recap

- GDA has quadratic (conic) decision boundary.
- With shared covariance, GDA is similar to logistic regression.
- Generative models:
  - Flexible models, easy to add/remove class.
  - Handle missing data naturally.
  - More "natural" way to think about things, but usually doesn't work as well.
- Tries to solve a hard problem (model $p(\mathbf{x})$) in order to solve a easy problem (model $p(t \mid \mathbf{x})$).

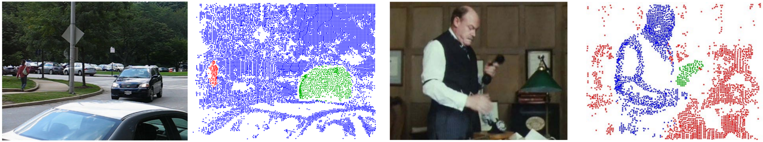
Next up: Unsupervised learning with PCA!

Unsupervised Learning: Motivating Examples

- Some examples of situations where you'd use unsupervised learning
  - You want to understand how a scientific field has changed over time. You want to take a large database of papers and model how the distribution of topics changes from year to year. But what are the topics?
  - You're a biologist studying animal behavior, so you want to infer a high-level description of their behavior from video. You don't know the set of behaviors ahead of time.
  - You want to reduce your energy consumption, so you take a time series of your energy consumption over time, and try to break it down into separate components (refrigerator, washing machine, etc.).
- Common theme: you have some data, and you want to infer the structure underlying the data.
- This structure is latent, which means it's never observed.

Motivating Examples

- Determine groups of people in image above
  - based on clothing styles, gender, age, etc



- Determine moving objects in videos

PCA Overview

- We now turn to the first unsupervised learning algorithm for this course: principal component analysis (PCA)
- Dimensionality reduction: map data to a lower dimensional space
  - Save computation/memory
  - Reduce overfitting, achieve better generalization
  - Visualize in 2 dimensions
- PCA is a linear model. It's useful for understanding lots of other algorithms.
  - Autoencoders
  - Matrix factorizations (next week)
- PCA is linear-algebra-heavy. But we covered a lot of the main intuitions already when we framed multivariate Gaussians as a multivariate shift and "scale".

Recall: Multivariate Parameters

- Setup: given a iid dataset $\mathcal{D} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}\} \subset \mathbb{R}^D$.
- N instances/observations/examples

$$\mathbf{X} = \begin{bmatrix} [\mathbf{x}^{(1)}]^\top \\ [\mathbf{x}^{(2)}]^\top \\ \vdots \\ [\mathbf{x}^{(N)}]^\top \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_D^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdots & x_D^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{(N)} & x_2^{(N)} & \cdots & x_D^{(N)} \end{bmatrix}$$

- Mean

$$\mathbb{E}[\mathbf{x}^{(i)}] = \boldsymbol{\mu} = [\mu_1, \cdots, \mu_D]^T \in \mathbb{R}^D$$

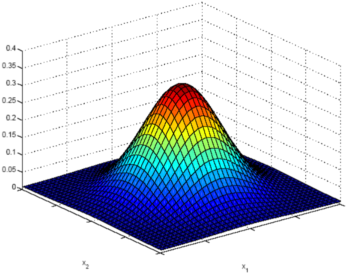
- Covariance

$$\boldsymbol{\Sigma} = \mathrm{Cov}(\mathbf{x}^{(i)}) = \mathbb{E}[(\mathbf{x}^{(i)} - \boldsymbol{\mu})(\mathbf{x}^{(i)} - \boldsymbol{\mu})^\top] = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1D} \\ \sigma_{12} & \sigma_2^2 & \cdots & \sigma_{2D} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{D1} & \sigma_{D2} & \cdots & \sigma_D^2 \end{bmatrix}$$

Multivariate Gaussian Model

- $\mathbf{x}^{(i)} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, a Gaussian (or normal) distribution defined as

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$



Mean and Covariance Estimators

- Observe data $\mathcal{D} = \{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(N)}\}$.
- Recall that the MLE estimators for the mean $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ under the multivariate Gaussian model is given by (previous lecture)

$$\text{Sample mean:} \quad \hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}^{(i)}$$

Sample covariance:

$$\begin{aligned} \hat{\boldsymbol{\Sigma}} &= \frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}})(\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}})^{\top} \\ &= \frac{1}{N} (\mathbf{X} - \mathbf{1}\boldsymbol{\mu}^{\top})^{\top} (\mathbf{X} - \mathbf{1}\boldsymbol{\mu}^{\top}) \end{aligned}$$

- $\hat{\boldsymbol{\mu}}$ quantifies where your data is located in space **(shift)**
- $\hat{\boldsymbol{\Sigma}}$ quantifies the shape of spread of your data points **(scale)**

Low dimensional representation

- In practice, even though data is very high dimensional, its important features can be accurately captured in a low dimensional subspace.

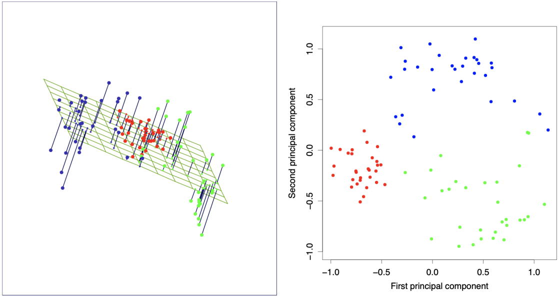


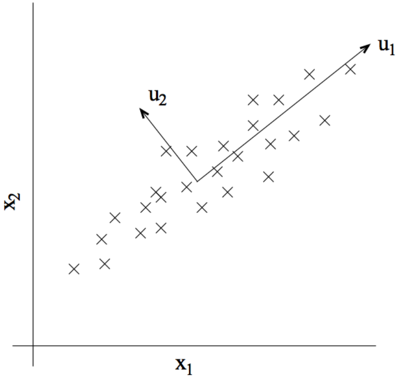
Image credit: Elements of Statistical Learning

- Find a low dimensional representation of your data.
  - Computational benefits
  - Interpretability, visualization
  - Generalization

Projection onto a subspace

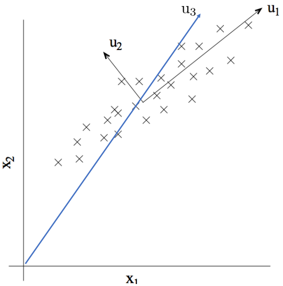
- Set-up: given a dataset $\mathcal{D} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\} \subset \mathbb{R}^D$
- Set $\hat{\boldsymbol{\mu}}$ to the sample mean of the data, $\hat{\boldsymbol{\mu}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}^{(i)}$
- Goal: find a K-dimensional subspace $\mathcal{S} \subset \mathbb{R}^D$ such that $\mathbf{x}^{(n)} - \hat{\boldsymbol{\mu}}$ is "well-represented" by its projection onto a K-dimensional $\mathcal{S}$
- Recall: The projection of a point $\mathbf{x}$ onto $\mathcal{S}$ is the point in $\mathcal{S}$ closest to $\mathbf{x}$. More on this coming soon.

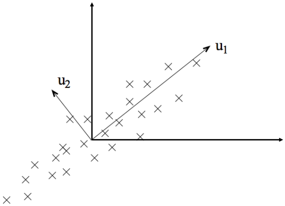
We are looking for directions



- For example, in a 2-dimensional problem, we are looking for the direction $\boldsymbol{u}_1$ along which the data is well represented: (?)
  - e.g. direction of higher variance
  - e.g. direction of minimum difference after projection
  - turns out they are the same!

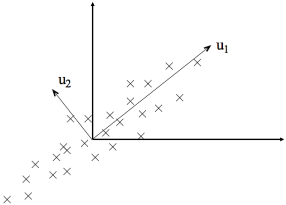
First step: Center data





- Directions we compute will pass through origin, and should represent the direction of highest variance.
- We need to center our data since we don't want location of data to influence our calculations. We are only interested in finding the direction of highest variance. This is independent from its mean.
- $\implies$ We are **not** interested in u_3, we are interested in u_1.

Second step: Project onto lower dimensional space $\mathcal{S}$



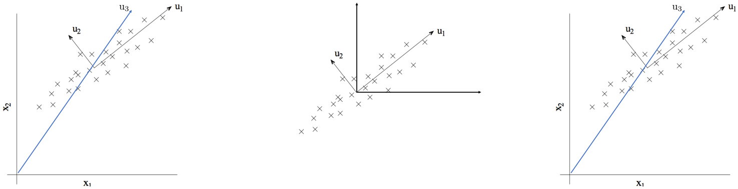
- A projection is just a multivariate "scale" by 0 in the pruned directions. You already know how to do this!
- Use positive semi-definite matrix:

$$\text{Proj}_{\mathbf{u}_1} = \mathbf{Q} \begin{pmatrix} 1 & 0 \\ 0 & {\color{red}0} \end{pmatrix} \mathbf{Q}^\top \quad \text{where} \quad \mathbf{Q} = \begin{pmatrix} | & | \\ \frac{\mathbf{u}_1}{\|\mathbf{u}_1\|} & \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} \\ | & | \end{pmatrix}$$

- This is the same as:

$$\text{Proj}_{\mathbf{u}_1} = \mathbf{Q} \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \mathbf{Q}^\top = \mathbf{U}\mathbf{U}^\top \quad \text{where} \quad \mathbf{U} = \left(\frac{\mathbf{u}_1}{\|\mathbf{u}_1\|} \right)$$

Third step: Add back mean



Summary for a given point $\mathbf{x}$:

1. Subtract mean: $\mathbf{x} - \hat{\boldsymbol{\mu}}$
2. Project on $\mathcal{S}$: $\mathbf{U}\mathbf{U}^\top(\mathbf{x} - \hat{\boldsymbol{\mu}})$, where columns of $\mathbf{U}$ are unit eigenvectors for largest K eigenvalues of $\hat{\boldsymbol{\Sigma}}$ (K directions of highest variance)
3. Add back mean: $\hat{\boldsymbol{\mu}} + \mathbf{U}\mathbf{U}^\top(\mathbf{x} - \hat{\boldsymbol{\mu}})$

The reconstruction is $\tilde{\mathbf{x}} = \hat{\boldsymbol{\mu}} + \mathbf{U}\mathbf{z} = \hat{\boldsymbol{\mu}} + \mathbf{U}\mathbf{U}^T(\mathbf{x} - \hat{\boldsymbol{\mu}})$

Here, $\mathbf{z} = \mathbf{U}^T(\mathbf{x} - \hat{\boldsymbol{\mu}})$ is a lower dimensional representation of $\mathbf{x}$.

And that's it! We've done Principal Components Analysis (PCA)!

- Let's now do this again in a bit more detail...