Tutorial on Estimation and Multivariate Gaussians

STAT 27725/CMSC 25400: Machine Learning

Shubhendu Trivedi - shubhendu@uchicago.edu

Toyota Technological Institute

October 2015
Things we will look at today

- Maximum Likelihood Estimation
- ML for Bernoulli Random Variables
- Maximizing a Multinomial Likelihood: Lagrange Multipliers
- Multivariate Gaussians
- Properties of Multivariate Gaussians
- Maximum Likelihood for Multivariate Gaussians
- (Time permitting) Mixture Models
The Principle of Maximum Likelihood

- Suppose we have \( N \) data points \( X = \{x_1, x_2, \ldots, x_N\} \) (or \( \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}\))
- Suppose we know the probability distribution function that describes the data \( p(x; \theta) \) (or \( p(y|x; \theta)\))
- Suppose we want to determine the parameter(s) \( \theta \)
- Pick \( \theta \) so as to explain your data best
- What does this mean?
- Suppose we had two parameter values (or vectors) \( \theta_1 \) and \( \theta_2 \).
- Now suppose you were to pretend that \( \theta_1 \) was really the true value parameterizing \( p \). What would be the probability that you would get the dataset that you have? Call this \( P_1 \)
- If \( P_1 \) is very small, it means that such a dataset is very unlikely to occur, thus perhaps \( \theta_1 \) was not a good guess
The Principle of Maximum Likelihood

- We want to pick $\theta_{ML}$ i.e. the best value of $\theta$ that explains the data you have.
- The plausibility of given data is measured by the "likelihood function" $p(x; \theta)$.
- Maximum Likelihood principle thus suggests we pick $\theta$ that maximizes the likelihood function.
- The procedure:
  - Write the log likelihood function: $\log p(x; \theta)$ (we’ll see later why log).
  - Want to maximize - So differentiate $\log p(x; \theta)$ w.r.t $\theta$ and set to zero.
  - Solve for $\theta$ that satisfies the equation. This is $\theta_{ML}$.
The Principle of Maximum Likelihood

- As an aside: Sometimes we have an initial guess for $\theta$ BEFORE seeing the data
- We then use the data to refine our guess of $\theta$ using Bayes Theorem
- This is called MAP (Maximum a posteriori) estimation (we’ll see an example)

Advantages of ML Estimation:
- Cookbook, ”turn the crank” method
- ”Optimal” for large data sizes

Disadvantages of ML Estimation
- Not optimal for small sample sizes
- Can be computationally challenging (numerical methods)
A Gentle Introduction: Coin Tossing
Problem: estimating bias in coin toss

- A single coin toss produces $H$ or $T$.
- A sequence of $n$ coin tosses produces a sequence of values; $n = 4$
  
  $T,H,T,H$
  $H,H,T,T$
  $T,T,T,H$

- A probabilistic model allows us to model the uncertainty inherent in the process (randomness in tossing a coin), as well as our uncertainty about the properties of the source (fairness of the coin).
Probabilistic model

• First, for convenience, convert $H \rightarrow 1$, $T \rightarrow 0$.
  • We have a random variable $X$ taking values in $\{0, 1\}$
• Bernoulli distribution with parameter $\mu$:
  $$\Pr(X = 1; \mu) = \mu.$$

• We will write for simplicity $p(x)$ or $p(x; \mu)$ instead of $\Pr(X = x; \mu)$
• The parameter $\mu \in [0, 1]$ specifies the bias of the coin
  • Coin is fair if $\mu = \frac{1}{2}$
Reminder: probability distributions

- Discrete random variable $X$ taking values in set $\mathcal{X} = \{x_1, x_2, \ldots\}$
- Probability mass function $p : \mathcal{X} \rightarrow [0, 1]$ satisfies the law of total probability:

\[ \sum_{x \in \mathcal{X}} p(X = x) = 1 \]

- Hence, for Bernoulli distribution we know

\[ p(0) = 1 - p(1; \mu) = 1 - \mu. \]
Sequence probability

Now consider two tosses of the same coin, $\langle X_1, X_2 \rangle$.

We can consider a number of probability distributions:

- Joint distribution $p(X_1, X_2)$
- Conditional distributions $p(X_1 | X_2), p(X_2 | X_1)$
- Marginal distributions $p(X_1), p(X_2)$

We already know the marginal distributions:

$p(X_1 = 1; \mu) \equiv p(X_2 = 1; \mu) = \mu$

What about the conditional?
We will assume the sequence is i.i.d. - independently identically distributed.

Independence, by definition, means

\[ p(X_1 | X_2) = p(X_1), \quad p(X_2 | X_1) = p(X_2) \]

i.e., the conditional is the same as marginal - knowing that \( X_2 \) was \( H \) does not tell us anything about \( X_1 \).

Finally, we can compute the joint distribution, using chain rule of probability:

\[ p(X_1, X_2) = p(X_1)p(X_2 | X_1) = p(X_1)p(X_2) \]
Sequence probability (contd)

\[
p(X_1, X_2) = p(X_1)p(X_2 | X_1) = p(X_1)p(X_2)
\]

- More generally, for i.i.d. sequence of \( n \) tosses,

\[
p(x_1, \ldots, x_n; \mu) = \prod_{i=1}^{n} p(x_i; \mu).
\]

- Example: \( \mu = \frac{1}{3} \). Then,

\[
p(H, T, H; \mu) = p(H; \mu)^2 p(T; \mu) = \left( \frac{1}{3} \right)^2 \cdot \frac{2}{3} = \frac{2}{27}.
\]

Note: the order of outcomes does not matter, only the number of \( H \)s and \( T \)s.
The parameter estimation problem

- Given a sequence of $n$ coin tosses $x_1, \ldots, x_n \in \{0, 1\}^n$, we want to estimate the bias $\mu$.
- Consider two coins, each tossed 6 times:
  - coin 1: $H, H, T, H, H, H$
  - coin 2: $T, H, T, T, H, H$
- What do you believe about $\mu_1$ vs. $\mu_2$?
- Need to convert this intuition into a precise procedure.
Maximum Likelihood estimator

- We have considered $p(x; \mu)$ as a function of $x$, parametrized by $\mu$.
- We can also view it as a function of $\mu$. This is called the likelihood function.
- Idea for estimator: choose a value of $\mu$ that maximizes the likelihood given the observed data.
ML for Bernoulli

- Likelihood of an i.i.d. sequence \( \mathbf{X} = [x_1, \ldots, x_n] \):

\[
L(\mu) = p(\mathbf{X}; \mu) = \prod_{i=1}^{n} p(x_i; \mu) = \prod_{i=1}^{n} \mu^{x_i} (1 - \mu)^{1-x_i}
\]

- log-likelihood:

\[
l(\mu) = \log p(\mathbf{X}; \mu) = \sum_{i=1}^{n} [x_i \log \mu + (1 - x_i) \log(1 - \mu)]
\]

- Due to monotonicity of \( \log \), we have

\[
\arg\max_{\mu} p(\mathbf{X}; \mu) = \arg\max_{\mu} \log p(\mathbf{X}; \mu)
\]

- We will usually work with log-likelihood (why?)
ML for Bernoulli (contd)

- ML estimate is
  \[ \hat{\mu}_{ML} = \arg\max_{\mu} \left\{ \sum_{i=1}^{n} [x_i \log \mu + (1 - x_i) \log (1 - \mu)] \right\} \]
  - To find it, set the derivative to zero:
    \[
    \frac{\partial}{\partial \mu} \log p(X; \mu) = \frac{1}{\mu} \sum_{i=1}^{n} x_i - \frac{1}{1 - \mu} \sum_{j=1}^{n} (1 - x_j) = 0
    \]
    \[
    \frac{1 - \mu}{\mu} = \frac{\sum_{j=1}^{n} (1 - x_j)}{\sum_{i=1}^{n} x_i}
    \]
    \[ \hat{\mu}_{ML} = \frac{1}{n} \sum_{i=1}^{n} x_i \]

- ML estimate is simply the fraction of times that \( H \) came up.
Are we done?

\[ \hat{\mu}_{ML} = \frac{1}{n} \sum_{i=1}^{n} x_i \]

- Example: \( H,T,H,T \rightarrow \hat{\mu}_{ML} = \frac{1}{2} \)
- How about: \( H \ H \ H \ H \)? \( \rightarrow \hat{\mu}_{ML} = 1 \)

Does this make sense?

- Suppose we record a very large number of 4-toss sequences for a coin with true \( \mu = \frac{1}{2} \).
  We can expect to see \( H,H,H,H \) about 1/16 of all sequences!

- A more extreme case: consider a single toss. \( \hat{\mu}_{ML} \) will be either 0 or 1.
Bayes rule

To proceed, we will need to use Bayes rule.

We can write the joint probability of two RV in two ways, using chain rule:

\[ p(X, Y) = p(X)p(Y|X) = p(Y)p(X|Y). \]

From here we get the Bayes rule:

\[ p(X|Y) = \frac{p(X)p(Y|X)}{p(Y)}. \]
Bayes rule and estimation

- Now consider $\mu$ to be a RV. We have

$$p(\mu | X) = \frac{p(X | \mu)p(\mu)}{p(X)}$$

- Bayes rule converts *prior* probability $p(\mu)$ (our belief about $\mu$ prior to seeing any data) to *posterior* $p(\mu | X)$, using the likelihood $p(X | \mu)$. 
MAP estimation

\[ p(\mu | \mathbf{X}) = \frac{p(\mathbf{X} | \mu)p(\mu)}{p(\mathbf{X})} \]

- The maximum a-posteriori (MAP) estimate is defined as

\[ \hat{\mu}_{MAP} = \arg\max_{\mu} p(\mu | \mathbf{X}) \]

- Note: \( p(\mathbf{X}) \) does not depend on \( \mu \), so if we only care about finding the MAP estimate, we can write

\[ p(\mu | \mathbf{X}) \propto p(\mathbf{X} | \mu)p(\mu) \]

- What’s \( p(\mu) \)?
Bayesian approach: try to reflect our belief about $\mu$

Utilitarian approach: choose a prior which is computationally convenient

- Later in class: regularization - choose a prior that leads to better prediction performance

One possibility: uniform $p(\mu) \equiv 1$ for all $\mu \in [0, 1]$. “Uninformative” prior: MAP is the same as ML estimate
Constrained Optimization: A Multinomial Likelihood
Problem: estimating biases in Dice

- A dice is rolled $n$ times: A single roll produces one of 
  \{1, 2, 3, 4, 5, 6\}
- Let $n_1, n_2, \ldots n_6$ count the outcomes for each value
- This is a multinomial distribution with parameters 
  $\theta_1, \theta_2, \ldots, \theta_6$
- The joint distribution for $n_1, n_2, \ldots, n_6$ is given by

\[
p(n_1, n_2, \ldots, n_6; n, \theta_1, \theta_2, \ldots, \theta_6) = \left( \frac{n!}{n_1!n_2!n_3!n_4!n_5!n_6!} \right) \prod_{i=1}^{6} \theta_i^{n_i}
\]

Subject to $\sum_i \theta_i = 1$ and $\sum_i n_i = n$
The likelihood is

\[ L(\theta_1, \theta_2, \ldots, \theta_6) = \left( \frac{n!}{n_1!n_2!n_3!n_4!n_5!n_6!} \right)^6 \prod_{i=1}^{6} \theta_{n_i}^i \]

The Log-Likelihood is

\[ l(\theta_1, \theta_2, \ldots, \theta_6) = \left( \log \frac{n!}{n_1!n_2!n_3!n_4!n_5!n_6!} \right) + \sum_{i=1}^{6} n_i \log \theta_i \]

Optimize by taking derivative and setting to zero:

\[ \frac{\partial l}{\partial \theta_1} = \frac{n_1}{\theta_1} = 0 \]

Therefore: \( \theta_1 = \infty \)

What went wrong?
A Possible Solution

- We forgot that \( \sum_{i=1}^{6} \theta_i = 1 \)
- We could use this constraint to eliminate one of the variables:

\[
\theta_6 = 1 - \sum_{i=1}^{5} \theta_i
\]

- and then solve the equations

\[
\frac{\partial l}{\partial \theta_i} = \frac{n_1}{\theta_i} - \frac{n_6}{1 - \sum_{i=1}^{5} \theta_i} = 0
\]

- Gets messy
A More Elegant Solution: Lagrange Multipliers

- General constrained optimization problem:
  \[
  \max_{\theta} f(\theta) \text{ subject to } g(\theta) - c = 0
  \]

- We can then define the Lagrangian
  \[
  \mathcal{L}(\theta, \lambda) = f(\theta) - \lambda(g(\theta) - c)
  \]

- Is equal to \( f \) when the constraint is satisfied
- Now do unconstrained optimization over \( \theta \) and \( \lambda \):
- Optimizing the Lagrange multiplier \( \lambda \) enforces constraint
- More constraints, more multipliers
Recall

\[ l(\theta_1, \theta_2, \ldots, \theta_6) = \left( \log \frac{n!}{n_1!n_2!n_3!n_4!n_5!n_6!} \right) + \sum_{i=1}^{6} n_i \log \theta_i \]

The Lagrangian may be defined as:

\[ \mathcal{L} = \log \frac{n!}{\prod_i n_i!} + \sum_{i=1}^{6} n_i \log \theta_i - \lambda \left( \sum_{i=1}^{6} \theta_i - 1 \right) \]
Back to Rolling Dice

- Taking derivative with respect to $\theta_i$ and setting to 0
  \[
  \frac{\partial L}{\partial \theta_i} = 0
  \]

- Let optimal $\theta_i = \theta_i^*$
  \[
  \frac{n_i}{\theta_i^*} - \lambda^* = 0 \implies \frac{n_i}{\lambda^*} = \theta_i^*
  \]

  \[
  \sum_{i=1}^{6} \frac{n_i}{\lambda^*} = \sum_{i=1}^{6} \theta_i^* = 1
  \]

  \[
  \lambda^* = \sum_{i=1}^{6} n_i \implies \theta_i^* = \frac{n_i}{\sum_{i=1}^{6} n_i}
  \]
Multivariate Gaussians
Quick Review: Discrete/Continuous Random Variables

- A Random Variable is a function $X : \Omega \mapsto \mathbb{R}$
- The set of all possible values a random variable $X$ can take is called its range
- Discrete random variables can only take isolated values (probability of a random variable taking a particular value reduces to counting)
- Discrete Example: Sum of two fair dice

![Diagram of discrete random variable]

- Continuous Example: Speed of a car
Discrete Distributions

- Assume $X$ is a discrete random variable. We would like to specify probabilities of events $\{X = x\}$
- If we can specify the probabilities involving $X$, we can say that we have specified the probability distribution of $X$
- For a countable set of values $x_1, x_2, \ldots, x_n$, we have $\mathbb{P}(X = x_i) > 0, i = 1, 2, \ldots, n$ and $\sum_i \mathbb{P}(X = x_i) = 1$
- We can then define the **probability mass function** $f$ of $X$ by $f(X) = \mathbb{P}(X = x)$
- Sometimes write as $f_X$
Example: Toss a die and let $X$ be its face value. $X$ is discrete with range $\{1, 2, 3, 4, 5, 6\}$. The pmf is

\[
\begin{array}{cccccc|c}
 x & 1 & 2 & 3 & 4 & 5 & 6 \\
 f(x) & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\
 \hline
 & 1 & 1 & 1 & 1 & 1 & 1 \\
\end{array}
\]

Another example: Toss two dice and let $X$ be the largest face value. The pmf is

\[
\begin{array}{cccccc|c}
 x & 1 & 2 & 3 & 4 & 5 & 6 \\
 f(x) & \frac{1}{36} & \frac{3}{36} & \frac{5}{36} & \frac{7}{36} & \frac{9}{36} & \frac{11}{36} \\
 \hline
 & \frac{1}{36} & \frac{3}{36} & \frac{5}{36} & \frac{7}{36} & \frac{9}{36} & \frac{11}{36} \\
\end{array}
\]
A random variable $X$ taking values in set $\mathcal{X}$ is said to have a continuous distribution if $\mathbb{P}(X = x) = 0$ for all $x \in \mathcal{X}$.

The probability density function of a continuous random variable $X$ satisfies

- $f(x) \geq \forall x$
- $\int_{-\infty}^{\infty} f(x)dx = 1$
- $\mathbb{P}(a \leq X \leq b) = \int_{a}^{b} f(x)dx \forall a, b$

Probabilities correspond to areas under the curve $f(x)$.

Reminder: No longer need to have $\mathbb{P}(a \leq X \leq b) = \int_{a}^{b} f(x)dx \leq 1$ but must have $\int_{-\infty}^{\infty} f(x)dx = 1$.
Why Gaussians?

Gaussian distributions are widely used in machine learning:

- Central Limit Theorem!

\[ \bar{X}_n = X_1 + X_2 + \cdots + X_n \]
\[ \sqrt{n} \bar{X}_n \xrightarrow{d} \mathcal{N} (x; \mu, \sigma^2) \]

- Actually, there are a set of "Central Limit Theorems" (e.g. corresponding to \( p \)-Stable Distributions)
Why Gaussians?
Why Gaussians?

- Gaussian distributions are widely used in machine learning:
  - Central Limit Theorem!
  - Gaussians are convenient computationally;
  - Mixtures of Gaussians (just covered in class) are sufficient to approximate a wide range of distributions;
  - Closely related to squared loss (have seen earlier in class), an important error measure in statistics.
Reminder: univariate Gaussian distribution

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

- mean \( \mu \) determines location
- variance \( \sigma^2 \);
  - standard deviation \( \sqrt{\sigma^2} \)
  - determines the spread around \( \mu \)
Moments

- Reminder: expectation of a RV $x$ is $E[x] \triangleq \int x p(x) dx$, so

$$E[x] = \int_{-\infty}^{\infty} x N(x; \mu, \sigma^2) dx = \mu$$

- Variance of $x$ is $\text{var } x \triangleq E[(x - E[x])^2]$, and

$$\text{var } x = \int_{-\infty}^{\infty} (x - \mu)^2 N(x; \mu, \sigma^2) dx = \sigma^2$$
Multivariate Gaussian

- Gaussian distribution of a random vector $\mathbf{x}$ in $\mathbb{R}^d$:

$\mathcal{N}(\mathbf{x}; \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left( -\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right)$

- The $\frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}}$ factor ensures it’s a pdf (integrates to one).
Matrix notation

\[ \mathcal{N}(\mathbf{x}; \mu, \Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp \left( -\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1}(\mathbf{x} - \mu) \right) \]

- **Boldfaced lowercase vectors** \( \mathbf{x} \), **uppercase matrices** \( \Sigma \).
- **Determinant** \( |\Sigma| \)
- **Matrix inverse** \( \Sigma^{-1} \)
- **Transpose** \( \mathbf{x}^T, \Sigma^T \)
Mean of the Gaussian

- By definition,

\[
E [\mathbf{x}] = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \mathbf{x} \mathcal{N}(\mathbf{x}; \mu, \Sigma) d\mathbf{x}_1 \cdots d\mathbf{x}_d
\]

- Solving this we indeed get

\[
E [\mathbf{x}] = \mu
\]
Covariance

- Variance of a RV $x$ with mean $\mu$: $\sigma^2_x = E[(x - \mu)^2]$
- Generalization to two variables: covariance

$$\text{Cov}_{x_1,x_2} \triangleq E[(x_1 - \mu_1)(x_2 - \mu_2)]$$

- Measures how the two variables deviate together from their means ("co-vary").
- Note: $\text{Cov}_{x,x} \equiv \text{var}(x) = \sigma^2_x$
Correlation vs. covariance

- Correlation:

\[ \text{cor}(a, b) \triangleq \frac{\text{Cov}_{a,b}}{\sigma_a \sigma_b}. \]

- \( \text{cor}(a, b) \approx 1 \) measures the linear relationship between \( a \) and \( b \).
- \( -1 \leq \text{cor}(a, b) \leq +1 \); \( +1 \) or \( -1 \) means \( a \) is a linear function of \( b \).
Covariance matrix

For a random vector \( \mathbf{x} = [x_1, \ldots, x_d]^T \) with mean \( \mu \),

\[
\mathbf{\text{Cov}}_{\mathbf{x}} \triangleq \begin{bmatrix}
\sigma^2_{x_1} & \text{Cov}_{x_1, x_2} & \cdots & \text{Cov}_{x_1, x_d} \\
\text{Cov}_{x_2, x_1} & \sigma^2_{x_2} & \cdots & \text{Cov}_{x_2, x_d} \\
\vdots & \vdots & \ddots & \vdots \\
\text{Cov}_{x_d, x_1} & \text{Cov}_{x_d, x_2} & \cdots & \sigma^2_{x_d}
\end{bmatrix}.
\]

- Square, symmetric, non-negative main diagonal—why?
  - variances \( \geq 0 \), and \( \text{Cov}(x, y) = \text{Cov}(y, x) \) by definition
- One can show (directly from definition):
  \[
  \mathbf{\text{Cov}}_{\mathbf{x}} = E \left[ (\mathbf{x} - \mu)(\mathbf{x} - \mu)^T \right]
  \]
  - i.e. expectation of the outer product of \( \mathbf{x} - E[\mathbf{x}] \) with itself.
- Note: so far nothing Gaussian-specific!
Covariance of the Gaussian

- We need to calculate $E \left[ (x - \mu)(x - \mu)^T \right]$
- With a bit of algebra, we get

$$E \left[ xx^T \right] = \mu\mu^T + \Sigma$$

Now, we already have $E [x] = \mu$, and

$$E \left[ (x - \mu)(x - \mu)^T \right] = E \left[ xx^T - \mu x^T - x \mu^T + \mu \mu^T \right]$$
$$= E \left[ xx^T \right] - \left\{ \mu (E [x])^T + E [x] \mu^T - \mu \mu^T \right\}$$
$$= E \left[ xx^T \right] - \mu \mu^T = \Sigma$$
Properties of the covariance

- Consider the eigenvector equation: \( \Sigma u = \lambda u \)
- As a covariance matrix, \( \Sigma \) is symmetric \( d \times d \) matrix. Therefore, we have \( d \) solutions \( \{\lambda_i, u_i\}_{i=1}^{d} \) where the eigenvalues \( \lambda_i \) are real, and the eigenvectors \( u_i \) are orthonormal, i.e., inner product

\[
    u_j^T u_i = \begin{cases} 
    0 & \text{if } i \neq j, \\
    1 & \text{if } i = j.
\end{cases}
\]

- The covariance matrix \( \Sigma \) then may be written as:

\[
    \Sigma = \sum_i \lambda_i u_i u_i^T
\]

- Thus, the inverse covariance may be written as:

\[
    \Sigma^{-1} = \sum_i \frac{1}{\lambda_i} u_i u_i^T
\]
The quadratic form \((x - \mu)^T \Sigma^{-1} (x - \mu)\) becomes:

\[
\sum_i \frac{y_i^2}{\lambda_i}
\]

where \(y_i = u_i^T (x - \mu)\)

\(\{y_i\}\) may be interpreted as a new coordinate system defined by the orthonormal vectors \(u_i\) that are shifted and rotated with respect to the original coordinate system.

Stack the \(d\) transposed orthonormal eigenvectors of \(\Sigma\) into

\[
U = \begin{bmatrix} u_1^T & \cdots & u_d^T \end{bmatrix}
\]

Then, \(y = U(x - \mu)\) defines rotation (and possibly reflection) of \(x\), shifted so that \(\mu\) becomes origin.
Geometry of the Gaussian

- $\sqrt{\lambda_i}$ gives scaling along $u_i$
- Example in 2D:
The determinant of the covariance matrix may be written as the product of its eigenvalues i.e.
\[ |\Sigma|^{\frac{1}{2}} = \prod_j \lambda_j^{\frac{1}{2}} \]

Thus, in the \( y_i \) coordinate system, the Gaussian distribution takes the form:

\[
p(y) = \prod_j \frac{1}{(2\pi\lambda_j)^{\frac{1}{2}}} \exp \left(-\frac{y_j^2}{2\lambda_j}\right)
\]

which is the product of \( d \) independent univariate Gaussians.

The eigenvectors thus define a new set of shifted and rotated coordinates w.r.t which the joint probability distribution factorizes into a product of independent distributions.
Density contours

- What are the constant density contours?

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

\[
(x - \mu)^T \Sigma^{-1} (x - \mu) = \text{const}
\]

- This is a quadratic form, whose solution is an ellipsoid (in 2D, simply an ellipse)
Density Contours are Ellipsoids

- We saw that: \((x - \mu)^T \Sigma^{-1} (x - \mu) = \text{const}^2\)
- Recall that \(\Sigma^{-1} = \sum \frac{1}{\lambda_i} u_i u_i^T\)
- Thus we have:

\[
\sum \frac{y_i^2}{\lambda_i} = \text{const}^2
\]

where \(y_i = u_i^T (x - \mu)\)

- Recall the expression for an ellipse in 2D: \(\left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 = 1\)
Intuition so far

\[ \mathcal{N}(\mathbf{x}; \mu, \Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp \left( -\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1}(\mathbf{x} - \mu) \right) \]

- Falls off exponentially as a function of (squared) Euclidean distance to the mean \( \| \mathbf{x} - \mu \|^2 \);
- the covariance matrix \( \Sigma \) determines the shape of the density;
- Determinant \( |\Sigma| \) measures the “spread” (analogous to \( \sigma^2 \)).
- \( \mathcal{N} \) is the joint density of coordinates \( x_1, \ldots, x_d \).
Linear functions of a Gaussian RV

- For any RV $x$, and for any $A$ and $b$,

$$E[Ax + b] = AE[x] + b,$$  \[\text{Cov}(Ax+b) = A \text{Cov}(x)A^T.\]

- Let $x \sim \mathcal{N}(\cdot; \mu, \Sigma)$; then $p(z) = \mathcal{N}(z; A\mu + b, A\Sigma A^T)$.

- Consider a row vector $a^T$ that “selects” a single component from $x$, i.e., $a_k = 1$ and $a_j = 0$ if $j \neq k$. Then, $z = a^T x$ is simply the coordinate $x_k$.

- We have: $E[z] = a^T \mu = \mu_k$, and $\text{Cov}(z) = \text{var}(z) = \Sigma_{k,k}$. i.e., marginal of a Gaussian is also a Gaussian.
Conditional and marginal

- Marginal ("projection" of the Gaussian on a subset of coordinates) is Gaussian
- Conditional ("slice" through Gaussian at fixed values for a subset of coordinates) is Gaussian
Log-likelihood

\[ \mathcal{N}(x; \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left( -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right) \]

- Take the log, for a single example \( x \):

\[ \log \mathcal{N}(x; \mu, \Sigma) = -\frac{d}{2} \log 2\pi - \frac{1}{2} \log |\Sigma| - \frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \]

- Can ignore terms independent of parameters:

\[ \log \mathcal{N}(x; \mu, \Sigma) = -\frac{1}{2} \log |\Sigma| - \frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) + \text{const} \]
Log-likelihood (contd)

\[
\log \mathcal{N}(x; \mu, \Sigma) = -\frac{1}{2} \log |\Sigma| - \frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu) + \text{const}
\]

- Given a set \(X\) of \(n\) i.i.d. vectors, we have

\[
\log \mathcal{N}(X; \mu, \Sigma) = -\frac{n}{2} \log |\Sigma| - \frac{1}{2} \sum_{i=1}^{n} (x_i - \mu)^T \Sigma^{-1}(x_i - \mu) + \text{const}
\]

- We are now ready to compute ML estimates for \(\mu\) and \(\Sigma\).
ML for parameters

\[
\log \mathcal{N}(X; \mu, \Sigma) = -\frac{n}{2} \log |\Sigma| - \frac{1}{2} \sum_{i=1}^{n} (x_i - \mu)^T \Sigma^{-1} (x_i - \mu) + \text{const}
\]

- To find ML estimate, we use the rule
  \[
  \frac{\partial}{\partial a} a^T b = \frac{\partial}{\partial a} b^T a = b,
  \]
  and set derivative w.r.t. \( \mu \) to zero:
  \[
  \frac{\partial}{\partial \mu} \log \mathcal{N}(X; \mu, \Sigma) = \sum_{i=1}^{n} \Sigma^{-1} (x_i - \mu) = 0,
  \]
  which yields \( \hat{\mu}_{ML} = \frac{1}{n} \sum_{i=1}^{n} x_i \).
A somewhat lengthier derivation produces ML estimate for the covariance:

\[ \hat{\Sigma}_{ML} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)(x_i - \mu)^T. \]

Note: the \( \mu \) above is the ML estimate \( \hat{\mu}_{ML} \).

Thus ML estimates for the mean is the *sample mean* of the data, and ML estimate for the covariance is the *sample covariance* of the data.
Mixture Models and Expected Log Likelihood
Mixture Models

- **Assumptions:**
  - $k$ underlying types (clusters/components)
  - $y_i$ is the identity of the component "responsible" for $x_i$
  - $y_i$ is a *hidden* (latent) variable: never observed

- **A mixture model:**

\[
p(x; \pi) = \sum_{c=1}^{k} p(y = c) p(x | y = c)
\]

- $\pi_c$ are called mixing probabilities
- The component densities $p(x | y = c)$ needs to be parameterized

Next few slides adapted from TTIC 31020 by Gregory Shakhnarovich
Parametric Mixtures

- Suppose the parameters of the $c$-th component are $\theta_c$. Then we can denote $\theta = [\theta_1, \ldots, \theta_k]$ and write

$$p(x; \theta, \pi) = \sum_{c=1}^{k} \pi_c p(x, \theta_c)$$

- Any valid setting of $\theta$ and $\pi$, such that $\sum_{c=1}^{k} \pi_c = 1$ produces a valid pdf

- Example: Mixture of Gaussians
Generative Model for a Mixture

- The generative process with a $k$-component mixture:
  - The parameters $\theta_c$ for each component are fixed
  - Draw $y_i \sim [\pi_1, \ldots, \pi_k]$
  - Given $y_i$, draw $x_i \sim p(x|y_i; \theta_{y_i})$
- The entire generative model for $x$ and $y$

$$p(x, y; \theta, \pi) = p(y; \pi)p(x|y; \theta_y)$$

- What does this mean? Any data point $x_i$ could have been generated in $k$ ways
- If the $c$-th component is Gaussian i.e.

$$p(x|y = c) = \mathcal{N}(x; \mu_c, \Sigma_c)$$

$$p(x; \theta, \pi) = \sum_{c=1}^{k} \pi_c \mathcal{N}(x; \mu_c, \Sigma_c)$$

where $\theta = [\mu_1, \ldots, \mu_k, \Sigma_1, \ldots, \Sigma_k]$
Likelihood of a Mixture Model

- Usual Idea: Estimate set of parameters that maximize likelihood given observed data
- The log-likelihood of $\pi, \theta$ for $X = \{x_1, \ldots, x_N\}$:

$$
\log p(X; \pi, \theta) = \sum_{i=1}^{N} \log \left( \sum_{c=1}^{k} \pi_c \mathcal{N}(x_i; \mu_c, \Sigma_c) \right)
$$

- No closed form solution because of sum inside log
- How will we estimate parameters?
Scenario 1: Known Labels. Mixture Density Estimation

- Suppose that we do observe \( y_i \in \{1, \ldots, k\} \) for each \( i = 1, \ldots, N \).
- Let us introduce a set of binary indicator variables \( z_i = [z_{i1}, \ldots, z_{ik}] \), where:

\[
z_{ic} = \begin{cases} 
1 & \text{if } y_i = c \\
0 & \text{otherwise}
\end{cases}
\]

- The count of examples from \( c \)-th component

\[
N_c = \sum_{i=1}^{N} z_{ic}
\]
Scenario 1: Known Labels. Mixture Density Estimation

If we know $z_i$, the ML estimates of the Gaussian components are simply (as we have seen earlier)

$$\hat{\pi}_c = \frac{N_c}{N}$$

$$\hat{\mu}_c = \frac{1}{N_c} \sum_{i=1}^{N} z_{ic} x_i,$$

$$\hat{\Sigma}_c = \frac{1}{N_c} \sum_{i=1}^{N} z_{ic}(x_i - \hat{\mu}_c)(x_i - \hat{\mu}_c)^T$$
Scenario 2: Credit Assignment

- When we *don’t know* $y$, we face a credit assignment problem: Which component is responsible for $x_i$?
- Suppose for a moment that we do know the component parameters $\theta = [\mu_1, \ldots, \mu_k, \Sigma_1, \ldots, \Sigma_k]$ and mixing probabilities $\pi = [\pi_1, \ldots, \pi_k]$.
- Then, we can compute the posterior of each label using Bayes’ theorem:

  $$\gamma_{ic} = \hat{p}(y = c|x; \theta, \pi) = \frac{\pi_c p(x; \mu_c, \Sigma_c)}{\sum_{l=1}^{k} \pi_l p(x; \mu_l, \Sigma_l)}$$

- We call $\gamma_{ic}$ the *responsibility* of the $c$-th component for $x$.
Expected Likelihood

- The "complete data" likelihood (when \( z \) are known):

\[
p(X, Z; \pi, \theta) = \propto \prod_{i=1}^{N} \prod_{c=1}^{k} (\pi_c N(x_i; \mu_c, \Sigma_c))^{z_{ic}}
\]

and the log

\[
p(X, Z; \pi, \theta) = \text{const} + \sum_{i=1}^{N} \sum_{c=1}^{k} z_{ic} (\log \pi_c + \log N(x_i; \mu_c, \Sigma_c))
\]

- We can't compute it (why?), but can take the expectation w.r.t the posterior of \( z \), which is just \( \gamma_{ic} \) i.e. \( \mathbb{E}[z_{ic}] = \gamma_{ic} \)

- The expected likelihood of the data:

\[
\mathbb{E}[\log p(X, Z; \pi, \theta)] = \text{const} + \sum_{i=1}^{N} \sum_{c=1}^{k} \gamma_{ic} (\log \pi_c + \log N(x_i; \mu_c, \Sigma_c))
\]
Expectation Maximization

- The expected likelihood of the data:

\[
\mathbb{E}[\log p(X, Z; \pi, \theta)] = \text{const} + \sum_{i=1}^{N} \sum_{c=1}^{k} \gamma_{ic} (\log \pi_c + \log \mathcal{N}(x_i; \mu_c, \Sigma_c))
\]

- We can find \(\pi, \theta\) that maximizes this expected likelihood - by setting derivatives to zero and for \(\pi\), using Lagrange Multipliers to enforce \(\sum_c \pi_c = 1\)
Expectation Maximization

- If we know the parameters and indicators (assignments) we are done
- If we know the indicators but not the parameters, we can do ML estimation of the parameters - and we are done
- If we know the parameters but not the indicators, we can compute the posteriors of the indicators. With known posteriors, we can estimate parameters that maximize the expected likelihood - and then we are done
- In reality, we know neither the parameters nor the indicators
Expectation Maximization for Mixture Models

- General Mixture Models: 
  \[ p(x) = \sum_{c=1}^{k} \pi_c p(x; \theta_c) \]

- Initialize \( \pi, \theta^{old} \), and iterate until convergence:
  - E-Step: Compute responsibilities:
    \[ \gamma_{ic} = \frac{\pi_c^{old} p(x_i; \theta_c^{old})}{\sum_{l=1}^{k} \pi_l^{old} p(x_i; \theta_l^{old})} \]
  - M-Step: Re-estimate mixture parameters:
    \[ \pi^{old}, \theta^{new} = \arg \max_{\theta, \pi} \sum_{i=1}^{N} \sum_{c=1}^{k} \gamma_{ic} (\log \pi_c + \log p(x_i; \theta_c)) \]