CPSC 540: Machine Learning
Density Estimation

Mark Schmidt

University of British Columbia

Winter 2019
Supervised Learning vs. Structured Prediction

- In 340 we focused a lot on “classic” supervised learning:
  - Model \( p(y \mid x) \) where \( y \) is a single discrete/continuous variable.

- In the next few classes we’ll focus on density estimation:
  - Model \( p(x) \) where \( x \) is a vector or general object.

- **Structured prediction** is the logical combination of these:
  - Model \( p(y \mid x) \) where \( y \) is a vector or general object.
  - Can be viewed as “conditional” density estimation.
3 Classes of Structured Prediction Methods

3 main approaches to structured prediction:

1. **Generative models** use $p(y \mid x) \propto p(y, x)$ as in naive Bayes.
   - Turns structured prediction into density estimation.
   - But we'll want to go beyond naive Bayes.
   - Examples: Gaussian discriminant analysis, mixtures and Markov models, VAEs.

2. **Discriminative models** directly fit $p(y \mid x)$ as in logistic regression.
   - View structured prediction as conditional density estimation.
   - Lets you use complicated features $x$ that make the task easier.
   - Examples: Conditional random fields, conditional RBMs, conditional neural fields.

3. **Discriminant functions** just try to map from $x$ to $y$ as in SVMs.
   - Now you don't even need to worry about calibrated probabilities.
   - Examples: Structured SVMs, fully-convolutional networks, RNNs.
The next topic we’ll focus on is **density estimation**:

$$X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

$$\tilde{X} = \begin{bmatrix} \_ & \_ & \_ & \_ \\ \_ & \_ & \_ & \_ \\ \_ & \_ & \_ & \_ \\ \_ & \_ & \_ & \_ \\ \_ & \_ & \_ & \_ \end{bmatrix}$$

- What is probability of $[1 \ 0 \ 1 \ 1]$?
- Want to estimate probability of feature vectors $x^i$.

For the training data this is easy:
- Set $p(x^i)$ to “number of times $x^i$ is in the training data” divided by $n$.

We’re interested in the probability of test data,
- What is probability of seeing feature vector $\tilde{x}^i$ for a new example $i$.
Density Estimation Applications

- Density estimation could be called a “master problem” in machine learning.
  - Solving this problem lets you solve a lot of other problems.

- If you have \( p(x^i) \) then:
  - **Outliers** could be cases where \( p(x^i) \) is small.
  - **Missing data** in \( x^i \) can be “filled in” based on \( p(x^i) \).
  - **Vector quantization** can be achieved by assigning shorter code to high \( p(x^i) \) values.
  - **Association rules** can be computed from conditionals \( p(x^i \mid x_k^i) \).

- We can also do density estimation on \((x^i, y^i)\) jointly:
  - **Supervised learning** can be done by conditioning to give \( p(y^i \mid x^i) \).
  - **Feature relevance** can be analyzed by looking at \( p(x^i \mid y^i) \).
Density estimation is an unsupervised learning method.

- We only have $x^i$ values, but no explicit target labels.
- You want to do “something” with them.

Some unsupervised learning tasks from CPSC 340 (depending on semester):

- Clustering: what types of $x^i$ are there?
- Association rules: which $x_j$ and $x_k$ occur together?
- Outlier detection: is this a “normal” $x^i$?
- Latent-factors: what “parts” are $x^i$ made from?
- Data visualization: what do the high-dimensional $x^i$ look like?
- Ranking: which are the most important $x^i$?

You can probably address all these if you can do density estimation.
Bernoulli Distribution on Binary Variables

Let’s start with the simplest case: $x^i \in \{0, 1\}$ (e.g., coin flips),

$$X = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

For IID data the only choice is the Bernoulli distribution:

$$p(x^i = 1 \mid \theta) = \theta, \quad p(x^i = 0 \mid \theta) = 1 - \theta.$$

We can write both cases

$$p(x^i \mid \theta) = \theta I[x^i=1](1 - \theta) I[x^i=0], \quad \text{where} \quad I[y] = \begin{cases} 1 & \text{if } y \text{ is true} \\ 0 & \text{if } y \text{ is false} \end{cases}.$$
Maximum Likelihood with Bernoulli Distribution

- MLE for Bernoulli likelihood is

\[
\arg\max_{0 \leq \theta \leq 1} p(X \mid \theta) = \arg\max_{0 \leq \theta \leq 1} \prod_{i=1}^{n} p(x^i \mid \theta)
\]

\[
= \arg\max_{0 \leq \theta \leq 1} \prod_{i=1}^{n} \theta^{I[x^i=1]}(1 - \theta)^{I[x^i=0]}
\]

\[
= \arg\max_{0 \leq \theta \leq 1} \theta^{\text{number of } x^i = 1}(1 - \theta)^{\text{number of } x^i = 0}
\]

\[
= \arg\max_{0 \leq \theta \leq 1} \theta^{n_1}(1 - \theta)^{n_0},
\]

where \(n_1\) is count of number of 1 values and \(n_0\) is the number of 0 values.

- If you equate the derivative of the log-likelihood with zero, you get \(\theta = \frac{n_1}{n_1 + n_0}\).

- So if you toss a coin 50 times and it lands heads 24 times, your MLE is 24/50.
Multinomial Distribution on Categorical Variables

- Consider the multi-category case: \( x^i \in \{1, 2, 3, \ldots, k\} \) (e.g., rolling di),

\[
X = \begin{bmatrix}
2 \\
1 \\
1 \\
3 \\
1 \\
2
\end{bmatrix}.
\]

- The categorical distribution is

\[
p(x^i = c \mid \theta_1, \theta_2, \ldots, \theta_k) = \theta_c,
\]

where \( \sum_{c=1}^{k} \theta_c = 1 \).

- We can write this for a generic \( x \) as

\[
p(x^i \mid \theta_1, \theta_2, \ldots, \theta_k) = \prod_{c=1}^{k} \theta^I[x^i = c].
\]
Using Lagrange multipliers (bonus) to handle constraints, the MLE is

$$\theta_c = \frac{n_c}{\sum_{c'} n_{c'}}.$$  
("fraction of times you rolled a 4")

- If we never see category 4 in the data, should we assume $\theta_4 = 0$?
  - If we assume $\theta_4 = 0$ and we have a 4 in test set, our test set likelihood is 0.

- To leave room for this possibility we often use “Laplace smoothing”,

$$\theta_c = \frac{n_c + 1}{\sum_{c'} (n_{c'} + 1)}.$$

- This is like adding a “fake” example to the training set for each class.
MAP Estimation with Bernoulli Distributions

- In the binary case, a generalization of Laplace smoothing is
  \[ \theta = \frac{n_1 + \alpha - 1}{(n_1 + \alpha - 1) + (n_0 + \beta - 1)}, \]

- We get the MLE when \( \alpha = \beta = 1 \), and Laplace smoothing with \( \alpha = \beta = 2 \).

- This is a MAP estimate under a \textit{beta} prior,
  \[ p(\theta \mid \alpha, \beta) = \frac{1}{B(\alpha, \beta)} \theta^{\alpha-1}(1 - \theta)^{\beta-1}, \]
  where the \text{beta function} \( B \) makes the probability integrate to one.

  We want \( \int \theta p(\theta \mid \alpha, \beta) \, d\theta = 1 \), so define \( B(\alpha, \beta) = \int \theta^{\alpha-1}(1 - \theta)^{\beta-1} \, d\theta \).

- Note that \( B(\alpha, \beta) \) is constant in terms of \( \theta \), it doesn’t affect MAP estimate.
MAP Estimation with Categorical Distributions

- In the categorical case, a generalization of Laplace smoothing is

\[ \theta_c = \frac{n_c + \alpha_c - 1}{\sum_{c'=1}^{k}(n_{c'} + \alpha_{c'} - 1)}, \]

which is a MAP estimate under a Dirichlet prior,

\[ p(\theta_1, \theta_2, \ldots, \theta_k \mid \alpha_1, \alpha_2, \ldots, \alpha_k) = \frac{1}{B(\alpha)} \prod_{c=1}^{k} \theta_c^{\alpha_c-1}, \]

where \( B(\alpha) \) makes the multivariate distribution integrate to 1 over \( \theta \),

\[ B(\alpha) = \int_{\theta_1} \int_{\theta_2} \cdots \int_{\theta_{k-1}} \int_{\theta_k} \prod_{c=1}^{k} \left[ \theta_c^{\alpha_c-1} \right] d\theta_k d\theta_{k-1} \cdots d\theta_2 d\theta_1. \]

- Because of MAP-regularization connection, Laplace smoothing is regularization.
Now consider the case where $x^i \in \{0, 1\}^d$ (e.g., words in e-mails):

$$X = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
1 & 0 & 1 & 1 
\end{bmatrix}.$$

Now there are $2^d$ possible values of vector $x^i$.

- Can’t afford to even store a $\theta$ for each possible vector $x^i$.
- With $n$ training examples we see at most $n$ unique $x^i$ values.
- But unless we have a small number of repeated $x^i$ values, we’ll hopelessly overfit.

With finite dataset, we’ll need to make assumptions...
Product of Independent Distributions

- A common assumption is that the variables are independent:

\[
p(x_1^i, x_2^i, \ldots, x_d^i \mid \Theta) = \prod_{j=1}^{d} p(x_j^i \mid \theta_j).
\]

- Now we just need to model each column of \( X \) as its own dataset:

\[
X = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
1 & 0 & 1 & 1
\end{bmatrix} \quad \rightarrow \quad X_1 = \begin{bmatrix}
1 \\
0 \\
0 \\
0 \\
1
\end{bmatrix}, \quad X_2 = \begin{bmatrix}
0 \\
1 \\
1 \\
0 \\
0
\end{bmatrix}, \quad X_3 = \begin{bmatrix}
0 \\
0 \\
1 \\
0 \\
1
\end{bmatrix}, \quad X_4 = \begin{bmatrix}
0 \\
0 \\
0 \\
1 \\
1
\end{bmatrix}.
\]

- A big assumption, but now you can fit Bernoulli for each variable.
  - We used a similar independence assumption in CPSC 340 for naive Bayes.
Density Estimation and Fundamental Trade-off

- **“Product of independent” distributions** (with $d$ parameters):
  - Easily estimate each $\theta_c$ but can’t model many distributions.

- **General discrete distribution** (with $2^d$ parameters):
  - Hard to estimate $2^d$ parameters but can model any distribution.

- An unsupervised version of the fundamental trade-off:
  - Simple models often don’t fit the data well but don’t overfit much.
  - Complex models fit the data well but often overfit.

- We’ll consider models that lie between these extremes:
  1. Mixture models.
  2. Markov models.
  3. Graphical models.
Outline

1. Density Estimation
2. Continuous Distributions
Univariate Gaussian

Consider the case of a continuous variable \( x \in \mathbb{R} \):

\[
X = \begin{bmatrix}
0.53 \\
1.83 \\
-2.26 \\
0.86
\end{bmatrix}.
\]

Even with 1 variable there are many possible distributions.

Most common is the Gaussian (or “normal”) distribution:

\[
p(x^i \mid \mu, \sigma^2) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left( -\frac{(x^i - \mu)^2}{2\sigma^2} \right) \quad \text{or} \quad x^i \sim \mathcal{N}(\mu, \sigma^2),
\]

for \( \mu \in \mathbb{R} \) and \( \sigma > 0 \).
Univariate Gaussian

- Mean parameter $\mu$ controls location of center of density.
- Variance parameter $\sigma^2$ controls how spread out density is.

https://en.wikipedia.org/wiki/Gaussian_function
Univariate Gaussian

- Why use the Gaussian distribution?
  - Data might actually follow Gaussian.
    - Good justification if true, but usually false.
  - Central limit theorem: mean estimators converge in distribution to a Gaussian.
    - Bad justification: doesn't imply data distribution converges to Gaussian.
  - Distribution with maximum entropy that fits mean and variance of data (bonus).
    - “Makes the least assumptions” while matching first two moments of data.
    - But for complicated problems, just matching mean and variance isn't enough.
  - Closed-form maximum likelihood estimate (MLE).
    - MLE for the mean is the mean of the data (“sample mean” or “empirical mean”).
    - MLE for the variance is the variance of the data (“sample variance”).
    - “Fast and simple”.

Density Estimation
Continuous Distributions
Univariate Gaussian (MLE for Mean)

- Gaussian likelihood for an example $x^i$ is

$$p(x^i \mid \mu, \sigma^2) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left( -\frac{(x^i - \mu)^2}{2\sigma^2} \right).$$

- So the negative log-likelihood for $n$ IID examples is

$$-\log p(X \mid \mu, \sigma^2) = -\sum_{i=1}^{n} \log p(x^i \mid \mu, \sigma^2) = \frac{1}{2\sigma^2} \sum_{i=1}^{n} (x^i - \mu)^2 + n \log(\sigma) + \text{const.}$$

- Setting derivative with respect to $\mu$ to 0 gives MLE of

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} x^i. \quad \text{(for any } \sigma > 0\text{)},$$

so the MLE is the mean of the samples.
Univariate Gaussian (MLE for Variance)

- Gaussian likelihood for an example $x^i$ is

$$p(x^i \mid \mu, \sigma^2) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left( -\frac{(x^i - \mu)^2}{2\sigma^2} \right).$$

- So the negative log-likelihood for $n$ IID examples is

$$-\log p(X \mid \mu, \sigma^2) = -\sum_{i=1}^{n} \log p(x^i \mid \mu, \sigma^2) = \frac{1}{2\sigma^2} \sum_{i=1}^{n}(x^i - \mu)^2 + n \log(\sigma) + \text{const.}$$

- Plugging in $\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} x^i$ and setting derivative with respect to $\sigma$ to zero gives

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^{n}(x^i - \hat{\mu})^2, \quad \text{(variance of the samples)}$$

unless all $x^i$ are equal (then NLL is not bounded below and MLE doesn’t exist).
Alternatives to Univariate Gaussian

Why not the Gaussian distribution?
- Negative log-likelihood is a quadratic function of $\mu$,

$$- \log p(X \mid \mu, \sigma^2) = \frac{1}{2\sigma^2} \sum_{i=1}^{n} (x^i - \mu)^2 + n \log(\sigma) + \text{const.}$$

so as with least squares the Gaussian is not robust to outliers.

This is a histogram of the $x^i$ values, and the red line is the estimated density.
- We say Gaussian is “Light-tailed”: assumes most data is close to mean.
Alternatives to Univariate Gaussian

- Robust: Laplace distribution or student’s t-distribution

- “Heavy-tailed”: has non-trivial probability that data is far from mean.
Alternatives to Univariate Gaussian

- Gaussian distribution is **unimodal**.

- Laplace and student t are also unimodal so don’t fix this issue.
  - Next time we’ll discuss “mixture models” that address this.
The generalization to multiple variables is the multivariate normal/Gaussian, 

\[ a^T x^i \] is a univariate Gaussian for any \( a \in \mathbb{R}^d \).
Multivariate Gaussian Distribution

- The probability density for the multivariate Gaussian is given by

\[ p(x^i | \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left( -\frac{1}{2} (x^i - \mu)^T \Sigma^{-1} (x^i - \mu) \right), \quad \text{or } x^i \sim \mathcal{N}(\mu, \Sigma), \]

where \( \mu \in \mathbb{R}^d \), \( \Sigma \in \mathbb{R}^{d \times d} \) and \( \Sigma \succ 0 \), and \( |\Sigma| \) is the determinant.

- Derived as an affine transformation of univariate standard normals (bonus).
  - Take \( z_j^i \sim \mathcal{N}(0, 1) \) and replace with \( x^i = Az^i + \mu \) (where \( \Sigma = AA^T \)).

- If \( |\Sigma| = 0 \) we say the Gaussian is degenerate (bonus).
  - PDF does not integrate to 1 over all \( x^i \).
Product of Independent Gaussians

- If we have $d$ variables, we could make each follow an independent Gaussian,
  \[ x_j^i \sim \mathcal{N}(\mu_j, \sigma_j^2), \]

- In this case the joint density over all $d$ variables is
  \[ \prod_{j=1}^{d} p(x_j^i \mid \mu_j, \sigma_j^2) \propto \prod_{j=1}^{d} \exp \left( -\frac{(x_j^i - \mu_j)^2}{2\sigma_j^2} \right) \]
  \[ = \exp \left( -\frac{1}{2} \sum_{j=1}^{d} \frac{1}{\sigma_j^2} (x_j^i - \mu_j)^2 \right) \]
  \[ = \exp \left( -\frac{1}{2} (x^i - \mu)^T \Sigma^{-1} (x^i - \mu) \right) \quad (\text{matrix notation}) \]

  where $\mu = (\mu_1, \mu_2, \ldots, \mu_d)$ and $\Sigma$ is a diagonal matrix with diagonal elements $\sigma_j^2$.

- This is a special case of a multivariate Gaussian with a diagonal covariance $\Sigma$. 
Product of Independent Gaussians

- The effect of a diagonal $\Sigma$ on the multivariate Gaussian:
  - If $\Sigma = \alpha I$ the level curves are circles: 1 parameter.
  - If $\Sigma = D$ (diagonal) then axis-aligned ellipses: $d$ parameters.
  - If $\Sigma$ is dense they do not need to be axis-aligned: $d(d + 1)/2$ parameters.
    (by symmetry, we only need upper-triangular part of $\Sigma$)

- Diagonal $\Sigma$ assumes features are independent, dense $\Sigma$ models dependencies.
Summary

- **Density estimation**: unsupervised modelling of probability of feature vectors.

- **Categorical distribution** for modeling discrete data.
  - Beta and Dirichlet priors as priors that give closed-form MAP ("Laplace smoothing").

- **Product of independent distributions** is simple/crude density estimation method.

- **Gaussian distribution** is a common distribution with many nice properties.
  - Closed-form MLE.
  - But unimodal and not robust.

- Next time: going beyond Gaussians.
Lagrangian Function for Optimization with Equality Constraints

- Consider minimizing a differentiable $f$ with linear equality constraints,

$$\arg\min_{Aw=b} f(w).$$

- The Lagrangian of this problem is defined by

$$L(w, v) = f(w) + v^T(Aw - b),$$

for a vector $v \in \mathbb{R}^m$ (with $A$ being $m$ by $d$).

- At a solution of the problem we must have

$$\nabla_w L(w, v) = \nabla f(w) + A^T v = 0 \quad \text{(gradient is orthogonal to constraints)}$$

$$\nabla_v L(w, v) = Aw - b = 0 \quad \text{(constraints are satisfied)}$$

- So solution is stationary point of Lagrangian.
Density Estimation Continuous Distributions

Lagrangian Function for Optimization with Equality Constraints

- Scans from Bertsekas discussing Lagrange multipliers (also see CPSC 406).

3.1 NECESSARY CONDITIONS FOR EQUALITY CONSTRAINTS

In this section we consider problems with equality constraints of the form

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h_i(x) = 0, \quad i = 1, \ldots, m. \\
\end{align*}
\]  

(ECP)

We assume that \( f: \mathbb{R}^n \to \mathbb{R}, \) \( h_i: \mathbb{R}^n \to \mathbb{R}, \) \( i = 1, \ldots, m, \) are continuously differentiable functions. All the necessary and sufficient conditions of this chapter relating to a local minimum can also be shown to hold if \( f \) and \( h_i \) are defined and are continuously differentiable within just an open set containing the local minimum. The proofs are essentially identical to those given here.

For notational convenience, we introduce the constraint function \( h: \mathbb{R}^n \to \mathbb{R}^m, \) where

\[ h = (h_1, \ldots, h_m). \]

We can then write the constraints in the more compact form

\[ h(x) = 0. \]  

(3.1)

Our basic Lagrange multiplier theorem states that for a given local minimum \( x^* \), there exist scalars \( \lambda_1, \ldots, \lambda_m \), called Lagrange multipliers, such that

\[ \nabla f(x^*) + \sum_{i=1}^m \lambda_i \nabla h_i(x^*) = 0. \]  

(3.2)

There are two ways to interpret this equation:

(a) The cost gradient \( \nabla f(x^*) \) belongs to the subspace spanned by the constraint gradients at \( x^* \). The example of Fig. 3.1.1 illustrates this interpretation.

(b) The cost gradient \( \nabla f(x^*) \) is orthogonal to the subspace of first order feasible variations

\[ V(x^*) = \{ \Delta x \mid \nabla h_i(x^*)^T \Delta x = 0, \quad i = 1, \ldots, m \}. \]

This is the subspace of variations \( \Delta x \) for which the vector \( x = x^* + \Delta x \) satisfies the constraint \( h(x) = 0 \) up to first order. Thus, according to the Lagrange multiplier condition of Eq. (3.2), at the local minimum \( x^* \), the first order cost variation \( \nabla f(x^*)^T \Delta x \) is zero for all variations

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad x_1^2 + x_2^2 = 2. \\
\end{align*}
\]

At the local minimum \( x^* = (-1, -1) \), the cost gradient \( \nabla f(x^*) \) is normal to the constraint surface and it therefore, coincides with the constraint gradient \( \nabla h_1(x^*) = (-2, -2) \). The Lagrange multiplier is \( \lambda = 1/2 \).

\[ \Delta x \] in this subspace. This statement is analogous to the “zero gradient condition” \( \nabla f(x^*) = 0 \) of unconstrained optimization.

Here is a formal statement of the main Lagrange multiplier theorem:

Proposition 3.1.1 (Lagrange Multiplier Theorem – Necessary Conditions) Let \( x^* \) be a local minimum of \( f \) subject to \( h(x) = 0 \), and assume that the constraint gradients \( \nabla h_1(x^*), \ldots, \nabla h_m(x^*) \) are linearly independent. Then there exists a unique vector \( \lambda^* = (\lambda_1, \ldots, \lambda_m) \), called a Lagrange multiplier vector, such that

\[ \nabla f(x^*) + \sum_{i=1}^m \lambda_i \nabla h_i(x^*) = 0. \]  

(3.3)

If in addition \( f \) and \( \lambda \) are twice continuously differentiable, we have
Lagrangian Function for Optimization with Equality Constraints

- We can use these optimality conditions,
  \[
  \nabla_w L(w, v) = \nabla f(w) + A^T v = 0 \quad \text{(gradient is orthogonal to constraints)}
  \]
  \[
  \nabla_v L(w, v) = Aw - b = 0 \quad \text{(constraints are satisfied)}
  \]

  to solve some constrained optimization problems.

- A typical approach might be:
  1. Solve for \( w \) in the equation \( \nabla_w L(w, v) = 0 \) to get \( w = g(v) \) for some function \( g \).
  2. Plug this \( w = g(v) \) into the equation \( \nabla_v L(w, v) = 0 \) to solve for \( v \).
  3. Use this \( v \) in \( g(v) \) to get the optimal \( w \).

- But note that these are necessary conditions (may need to check it’s a min).
MAP for Univariate Gaussian Mean

- Assume $x^i \sim \mathcal{N}(\mu, \sigma^2)$ and assume $\mu \sim \mathcal{N}(\mu_0, 1)$.

- The MAP estimate of $\mu$ under these assumptions can be written as

$$\hat{\mu} = \frac{n}{n + \sigma^2} \bar{x} + \frac{\sigma^2}{n + \sigma^2} \mu_0,$$

where $\bar{x}$ is the sample mean, $\frac{1}{n} \sum_{i=1}^{n} x^i$ (which is the MLE).

- The MAP estimate is a convex combination of the MLE and prior mean $\mu_0$.
  - Regularizer moves us in a straight line away from MLE towards $\mu_0$. 

Maximum Entropy and Gaussian

Consider trying to find the PDF $p(x)$ that

1. **Agrees with the sample mean and sample covariance of the data.**
2. **Maximizes entropy** subject to these constraints,

$$\max_p \left\{ - \int_{-\infty}^{\infty} p(x) \log p(x) \, dx \right\}, \text{ subject to } \mathbb{E}[x] = \mu, \mathbb{E}[(x-\mu)^2] = \sigma^2.$$

- **Solution is the Gaussian with mean $\mu$ and variance $\sigma^2$.**
  - Beyond fitting mean/variance, Gaussian makes **fewest assumptions** about the data.

- This is proved using the **convex conjugate** (see duality lecture).
  - Convex conjugate of Gaussian negative log-likelihood is entropy.
  - Same result holds in higher dimensions for multivariate Gaussian.
If $|\Sigma| = 0$, we say the Gaussian is degenerate.

In this case the PDF only integrates to 1 along a subspace of the original space.

With $d = 2$ degenerate Gaussians only have non-zero probability along a line (or just one point).
Multivariate Gaussian from Univariate Gaussians

- Consider a joint distribution that is the product univariate standard normals:

\[
p(z^i) = \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi}} \exp \left( -\frac{1}{2} (z_j^i)^2 \right)
= \frac{1}{(2\pi)^{d/2}} \exp \left( \frac{1}{2} \langle z^i, z^i \rangle \right).
\]

- Now define \( x^i = A z^i + \mu \) for some (non-singular) matrix \( A \) and vector \( \mu \).

- The change of variables formula for multivariate probabilities is

\[
p(x^i) = p(z^i) \left| \frac{\partial z^i}{\partial x^i} \right|.
\]

- Plug in \( z^i = A^{-1}(x^i - \mu) \) and \( \frac{\partial z^i}{\partial x^i} = A^{-1} \ldots \).
Multivariate Gaussian from Univariate Gaussians

This gives

\[ p(x^i | \mu, A) = \frac{1}{(2\pi)^{d/2}} \exp \left( \frac{1}{2} \langle A^{-1}(x^i - \mu), A^{-1}(x^i \mu) \rangle \right) | \det(A^{-1})| \]

\[ = \frac{1}{(2\pi)^{d/2} | \det(A)|} \exp \left( \frac{1}{2} (x^i - \mu) A^{-\top} A^{-1} (x^i - \mu) \right) . \]

Define \( \Sigma = A A^\top \) (so \( \Sigma^{-1} = A^{-\top} A^{-1} \) and \( \det \Sigma = (\det A)^2 \)) to get

\[ p(x^i | \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} | \Sigma |^{1/2}} \exp \left( -\frac{1}{2} (x^i - \mu)^\top \Sigma^{-1} (x^i - \mu) \right) \]

So multivariate Gaussian is an affine transformation of independent Gaussians.