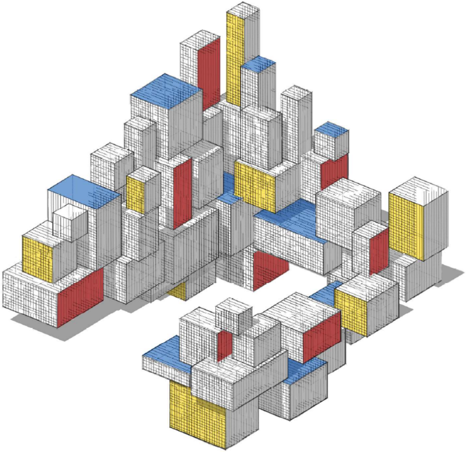
EM Algorithm and Density Estimation



Purpose

In this lecture we discuss:

- The expectation–maximization algorithm
- Density estimation

Expectation–Maximization Algorithm

The Expectation–Maximization algorithm (EM) is a general algorithm for maximization of complicated (log-)likelihood functions, through the introduction of auxiliary variables.

> To simplify the notation, we use a Bayesian notation system, where the same symbol is used for different (conditional) probability densities.

Given independent observations $\tau = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ from some unknown pdf f, the objective is to find the best approximation to f in a function class $\mathcal{G} = \{g(\cdot \,|\, \boldsymbol{\theta}), \boldsymbol{\theta} \in \Theta\}$ by solving the maximum likelihood problem:

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta} \in \Theta}{\operatorname{argmax}}\, g(\tau \,|\, \boldsymbol{\theta}),$$

where $g(\tau \,|\, \boldsymbol{\theta}) := g(\boldsymbol{x}_1 \,|\, \boldsymbol{\theta}) \cdots g(\boldsymbol{x}_n \,|\, \boldsymbol{\theta})$.

Latent Variables

The key element of the EM algorithm is the augmentation of the data τ with a suitable vector of latent variables, z, such that

$$g(\tau \,|\, \boldsymbol{\theta}) = \int g(\tau, z \,|\, \boldsymbol{\theta}) \, \mathrm{d}z.$$

The function $\boldsymbol{\theta} \mapsto g(\tau, z \,|\, \boldsymbol{\theta})$ is usually referred to as the complete-data likelihood function.

The choice of the latent variables is guided by the desire to make the maximization of $g(\tau, z \,|\, \boldsymbol{\theta})$ much easier than that of $g(\tau \,|\, \boldsymbol{\theta})$.

KL Divergence

Suppose p denotes an arbitrary density of the latent variables z. Then, we can write:

$$
\begin{aligned}
\ln g(\tau \,|\, \boldsymbol{\theta}) &= \int p(z) \ln g(\tau \,|\, \boldsymbol{\theta}) \, \mathrm{d}z \\
&= \int p(z) \ln \left(\frac{g(\tau, z \,|\, \boldsymbol{\theta}) / p(z)}{g(z \,|\, \tau, \boldsymbol{\theta}) / p(z)} \right) \mathrm{d}z \\
&= \int p(z) \ln \left(\frac{g(\tau, z \,|\, \boldsymbol{\theta})}{p(z)} \right) \mathrm{d}z - \int p(z) \ln \left(\frac{g(z \,|\, \tau, \boldsymbol{\theta})}{p(z)} \right) \mathrm{d}z \\
&= \int p(z) \ln \left(\frac{g(\tau, z \,|\, \boldsymbol{\theta})}{p(z)} \right) \mathrm{d}z + \mathcal{D}(p, g(\cdot \,|\, \tau, \boldsymbol{\theta})),
\end{aligned}
$$

where $\mathcal{D}(p, g(\cdot \,|\, \tau, \boldsymbol{\theta}))$ is the Kullback–Leibler divergence from the density p to $g(\cdot \,|\, \tau, \boldsymbol{\theta})$.

Lower Bound

Since $\mathcal{D} \geqslant 0$, it follows that

$$\ln g(\tau \,|\, \boldsymbol{\theta}) \geqslant \int p(\boldsymbol{z}) \ln \left(\frac{g(\tau, \boldsymbol{z} \,|\, \boldsymbol{\theta})}{p(\boldsymbol{z})} \right) \mathrm{d}\boldsymbol{z} =: \mathcal{L}(p, \boldsymbol{\theta})$$

for all $\boldsymbol{\theta}$ and any density p of the latent variables.

In other words, $\mathcal{L}(p, \boldsymbol{\theta})$ is a lower bound on the log-likelihood that involves the complete-data likelihood.

The EM algorithm then aims to increase this lower bound as much as possible by starting with an initial guess $\boldsymbol{\theta}^{(0)}$ and then, for $t = 1, 2, \ldots$, solving the following two steps:

1. $p^{(t)} = \operatorname{argmax}_p \mathcal{L}(p, \boldsymbol{\theta}^{(t-1)})$,
2. $\boldsymbol{\theta}^{(t)} = \operatorname{argmax}_{\boldsymbol{\theta} \in \Theta} \mathcal{L}(p^{(t)}, \boldsymbol{\theta})$.

Two Steps

The first optimization problem can be solved explicitly. Namely,

$$p^{(t)} = \underset{p}{\operatorname{argmin}}\, \mathcal{D}(p, g(\cdot \,|\, \tau, \boldsymbol{\theta}^{(t-1)})) = g(\cdot \,|\, \tau, \boldsymbol{\theta}^{(t-1)}).$$

That is, the optimal density is the conditional density of the latent variables given the data τ and the parameter $\boldsymbol{\theta}^{(t-1)}$.

The second optimization problem can be simplified by writing $\mathcal{L}(p^{(t)}, \boldsymbol{\theta}) = Q^{(t)}(\boldsymbol{\theta}) - \mathbb{E}_{p^{(t)}} \ln p^{(t)}(\boldsymbol{Z})$, where

$$Q^{(t)}(\boldsymbol{\theta}) := \mathbb{E}_{p^{(t)}} \ln g(\tau, \boldsymbol{Z} \,|\, \boldsymbol{\theta})$$

is the expected complete-data log-likelihood under $\boldsymbol{Z} \sim p^{(t)}$. Hence, maximization of $\mathcal{L}(p^{(t)}, \boldsymbol{\theta})$ means finding

$$\boldsymbol{\theta}^{(t)} = \underset{\boldsymbol{\theta} \in \Theta}{\operatorname{argmax}}\, Q^{(t)}(\boldsymbol{\theta}).$$

Algorithm 1: Generic EM Algorithm

input: Data τ, initial guess $\boldsymbol{\theta}^{(0)}$.

output: Approximation of the maximum likelihood estimate.

$t \leftarrow 1$

while a stopping criterion is not met **do**

  Expectation Step: Find $p^{(t)}(z) := g(z \mid \tau, \boldsymbol{\theta}^{(t-1)})$ and compute the expectation

$$Q^{(t)}(\boldsymbol{\theta}) := \mathbb{E}_{p^{(t)}} \ln g(\tau, \boldsymbol{Z} \mid \boldsymbol{\theta}). \tag{1}$$

  Maximization Step: Let $\boldsymbol{\theta}^{(t)} \leftarrow \operatorname{argmax}_{\boldsymbol{\theta} \in \Theta} Q^{(t)}(\boldsymbol{\theta})$.

  $t \leftarrow t + 1$

return $\boldsymbol{\theta}^{(t)}$

Properties of the EM Algorithm

The likelihood $g(\tau \mid \boldsymbol{\theta}^{(t)})$ does not decrease with every iteration of the algorithm.

The convergence of the sequence $\{\boldsymbol{\theta}^{(t)}\}$ to a global maximum (if it exists) is highly dependent on the initial value $\boldsymbol{\theta}^{(0)}$ and, in many cases, an appropriate choice of $\boldsymbol{\theta}^{(0)}$ may not be clear.

Typically, practitioners run the algorithm from different random starting points over Θ, to ascertain empirically that a suitable optimum is achieved.

Example: Censored Data

The lifetime of a certain type of machine is modeled via a $\mathcal{N}(\mu, \sigma^2)$ distribution.

To estimate μ and σ^2, the lifetimes of n (independent) machines are recorded up to c years.

Denote these censored lifetimes by $x_1, \ldots, x_n$. The $\{x_i\}$ are thus realizations of iid random variables $\{X_i\}$, distributed as $\min\{Y, c\}$, where $Y \sim \mathcal{N}(\mu, \sigma^2)$.

The marginal pdf of each X can thus be written as:

$$g(x \mid \mu, \sigma^2) = \underbrace{\Phi((c-\mu)/\sigma)}_{\mathbb{P}[Y<c]} \, \frac{\varphi_{\sigma^2}(x-\mu)}{\Phi((c-\mu)/\sigma)} \mathbb{I}\{x < c\} + \underbrace{\overline{\Phi}((c-\mu)/\sigma)}_{\mathbb{P}[Y \geqslant c]} \, \mathbb{I}\{x = c\},$$

where $\varphi_{\sigma^2}(\cdot)$ is the pdf of the $\mathcal{N}(0, \sigma^2)$ distribution, Φ is the cdf of the standard normal distribution, and $\overline{\Phi} := 1 - \Phi$.

Likelihood of Censored Data

It follows that the likelihood of the data $\tau = \{x_1, \ldots, x_n\}$ as a function of the parameter $\boldsymbol{\theta} := [\mu, \sigma^2]^\top$ is:

$$g(\tau \,|\, \boldsymbol{\theta}) = \prod_{i:x_i<c} \frac{\exp\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)}{\sqrt{2\pi\sigma^2}} \times \prod_{i:x_i=c} \overline{\Phi}((c-\mu)/\sigma).$$

Let n_c be the total number of x_i such that $x_i = c$. Using n_c latent variables $\boldsymbol{z} = [z_1, \ldots, z_{n_c}]^\top$, we can write the joint pdf:

$$g(\tau, \boldsymbol{z} \,|\, \boldsymbol{\theta}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{\sum_{i:x_i<c}(x_i-\mu)^2}{2\sigma^2} - \frac{\sum_{i=1}^{n_c}(z_i-\mu)^2}{2\sigma^2}\right) \mathbb{I}\left\{\min_i z_i \geqslant c\right\},$$

so that $\int g(\tau, \boldsymbol{z} \,|\, \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{z} = g(\tau \,|\, \boldsymbol{\theta})$. We can thus apply the EM algorithm to maximize the likelihood, as follows.

E and M Steps

For the E(xpectation)-step, we have for a fixed $\boldsymbol{\theta}$:

$$g(\boldsymbol{z} \mid \tau, \boldsymbol{\theta}) = \prod_{i=1}^{n_c} g(z_i \mid \tau, \boldsymbol{\theta}),$$

where $g(z \mid \tau, \boldsymbol{\theta}) = \mathbb{I}\{z \geqslant c\} \, \varphi_{\sigma^2}(z - \mu)/\overline{\Phi}((c - \mu)/\sigma)$ is simply the pdf of the $\mathcal{N}(\mu, \sigma^2)$ distribution, truncated to $[c, \infty)$.

For the M(aximization)-step, we compute the expectation of the complete log-likelihood with respect to a fixed $g(\boldsymbol{z} \mid \tau, \boldsymbol{\theta})$ and use the fact that $Z_1, \ldots, Z_{n_c}$ are iid:

$$\mathbb{E} \ln g(\tau, \boldsymbol{Z} \mid \boldsymbol{\theta}) = -\frac{\sum_{i: x_i < c} (x_i - \mu)^2}{2\sigma^2} - \frac{n_c \mathbb{E}(Z - \mu)^2}{2\sigma^2} - \frac{n}{2} \ln \sigma^2 - \frac{n}{2} \ln(2\pi),$$

where Z has a $\mathcal{N}(\mu, \sigma^2)$ distribution, truncated to $[c, \infty)$.

M-Step

To maximize the last expression with respect to μ we set the derivative with respect to μ to zero, and obtain:

$$\mu = \frac{n_c \mathbb{E} Z + \sum_{i:x_i<c} x_i}{n}.$$

Similarly, setting the derivative with respect to σ^2 to zero gives:

$$\sigma^2 = \frac{n_c \mathbb{E}(Z-\mu)^2 + \sum_{i:x_i<c} (x_i-\mu)^2}{n}.$$

EM Steps

In summary, the EM iterates for $t = 1, 2, \ldots$ are as follows.

- E-step. Given the current estimate $\boldsymbol{\theta}_t := [\mu_t, \sigma_t^2]^\top$, compute the expectations $\nu_t := \mathbb{E}Z$ and $\zeta_t^2 := \mathbb{E}(Z - \mu_t)^2$, where $Z \sim \mathcal{N}(\mu_t, \sigma_t^2)$, conditional on $Z \geqslant c$; that is,

$$\nu_t := \mu_t + \sigma_t^2 \frac{\varphi_{\sigma_t^2}(c - \mu_t)}{\overline{\Phi}((c - \mu_t)/\sigma_t)}$$

$$\zeta_t^2 := \sigma_t^2 \left(1 + (c - \mu_t) \frac{\varphi_{\sigma_t^2}(c - \mu_t)}{\overline{\Phi}((c - \mu_t)/\sigma_t)}\right).$$

- M-step. Update the estimate to $\boldsymbol{\theta}_{t+1} := [\mu_{t+1}, \sigma_{t+1}^2]^\top$ via:

$$\mu_{t+1} = \frac{n_c \nu_t + \sum_{i:x_i<c} x_i}{n}$$

$$\sigma_{t+1}^2 = \frac{n_c \zeta_t^2 + \sum_{i:x_i<c}(x_i - \mu_{t+1})^2}{n}.$$

Empirical Distribution and Density Estimation

Suppose we have an iid training set $\tau = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ from an unknown pdf f.

A random vector $\boldsymbol{X}$ that is distributed according to the empirical distribution of τ has discrete pdf $\mathbb{P}[\boldsymbol{X} = \boldsymbol{x}_i] = 1/n, i = 1, \ldots, n$.

For continuous data it makes sense to also consider a kernel density estimate (KDE).

Definition: Gaussian KDE

A Gaussian kernel density estimate of f is a mixture of normal pdfs, of the form

$$g_{\tau_n}(\boldsymbol{x} \mid \sigma) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{(2\pi)^{d/2}\sigma^d} \, \mathrm{e}^{-\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}{2\sigma^2}}, \quad \boldsymbol{x} \in \mathbb{R}^d, \qquad (2)$$

where $\sigma > 0$ is called the bandwidth.

Gaussian KDE

Thus g_{τ_n} in (2) is the average of a collection of n normal pdfs, where each normal distribution is centered at the data point $\boldsymbol{x}_i$ and has covariance matrix $\sigma^2 \mathbf{I}_d$.

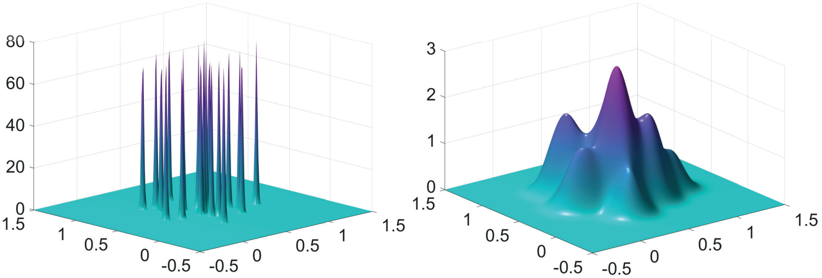


Figure: Two two-dimensional Gaussian KDEs, with $\sigma = 0.01$ (left) and $\sigma = 0.1$ (right).

Different Kernels

Write the Gaussian KDE in (2) as

$$g_{\tau_n}(\boldsymbol{x} \mid \sigma) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\sigma^d} \phi\left(\frac{\boldsymbol{x} - \boldsymbol{x}_i}{\sigma}\right),$$

where

$$\phi(\boldsymbol{z}) = \frac{1}{(2\pi)^{d/2}}\, \mathrm{e}^{-\frac{\|\boldsymbol{z}\|^2}{2}}, \quad \boldsymbol{z} \in \mathbb{R}^d$$

is the pdf of the d-dimensional standard normal distribution.

By choosing a different pdf ϕ, with$\phi(\boldsymbol{x}) = \phi(-\boldsymbol{x})$, we can obtain a wide variety of KDEs.

A simple pdf ϕ is, for example, the uniform pdf on $[-1, 1]^d$:

$$\phi(\boldsymbol{z}) = \begin{cases} 2^{-d}, & \text{if } \boldsymbol{z} \in [-1, 1]^d, \\ 0, & \text{otherwise.} \end{cases}$$

Uniform KDE

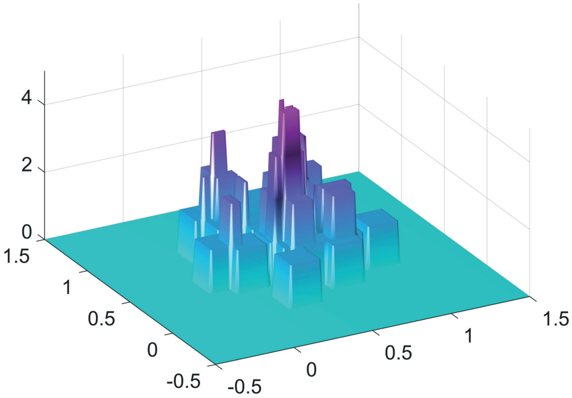


Figure: A two-dimensional uniform KDE, with bandwidth $\sigma = 0.1$.

Qualitatively similar behavior for the Gaussian and uniform KDEs.
The choice of ϕ is less important than the choice of σ.

Bandwidth Selection

Bandwidth selection has been extensively studied for one-dimensional data $\tau = \{x_1, \ldots, x_n\}$ from unkown pdf f.

First, we define the loss function as

$$\text{Loss}(f(x), g(x)) = \frac{(f(x) - g(x))^2}{f(x)}. \tag{3}$$

The risk to minimize is thus $\int (f(x) - g(x))^2 \, \mathrm{d}x$.

We choose the learner g_τ of the form by (2) for a fixed σ.

The objective is now to find a σ that minimizes the generalization risk $\ell(g_\tau(\cdot \,|\, \sigma))$ or the expected generalization risk $\mathbb{E}\ell(g_{\mathcal{T}}(\cdot \,|\, \sigma))$. The generalization risk is in this case

$$\int (f(x) - g_\tau(x \,|\, \sigma))^2 \, \mathrm{d}x = \int f^2(x) \, \mathrm{d}x - 2 \int f(x) g_\tau(x \,|\, \sigma) \, \mathrm{d}x + \int g_\tau^2(x \,|\, \sigma) \, \mathrm{d}x.$$

Minimization of the Generalization Risk

Minimizing this generalization risk with respect to σ is equivalent to minimizing the last two terms, which can be written as

$$-2\,\mathbb{E}_f\, g_\tau(X \,|\, \sigma) + \int \left(\frac{1}{n} \sum_{i=1}^{n} \frac{1}{\sigma} \phi\left(\frac{x - x_i}{\sigma}\right)\right)^2 \mathrm{d}x.$$

This expression in turn can be estimated by using a test sample $\{x'_1 \ldots, x'_{n'}\}$ from f, yielding the following minimization problem:

$$\min_{\sigma} -\frac{2}{n'} \sum_{i=1}^{n'} g_\tau(x'_i \,|\, \sigma) + \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \int \frac{1}{\sigma^2} \phi\left(\frac{x - x_i}{\sigma}\right) \phi\left(\frac{x - x_j}{\sigma}\right) \mathrm{d}x,$$

where $\int \frac{1}{\sigma^2} \phi\left(\frac{x-x_i}{\sigma}\right) \phi\left(\frac{x-x_j}{\sigma}\right) \mathrm{d}x = \frac{1}{\sqrt{2}\sigma} \phi\left(\frac{x_i - x_j}{\sqrt{2}\sigma}\right)$ in the case of the Gaussian kernel with $d = 1$.

MISE

To estimate σ requires a test sample or an application of cross-validation. Another approach is to minimize the expected generalization risk, called the mean integrated squared error (MISE):

$$\mathbb{E} \int (f(x) - g_{\mathcal{T}}(x \,|\, \sigma))^2 \, \mathrm{d}x.$$

Decompose into an integrated squared bias and integrated variance:

$$\int (f(x) - \mathbb{E} g_{\mathcal{T}}(x \,|\, \sigma))^2 \, \mathrm{d}x + \int \mathbb{V}\mathrm{ar}(g_{\mathcal{T}}(x \,|\, \sigma)) \, \mathrm{d}x.$$

It can be show that for $\sigma \to 0$ and $n\sigma \to \infty$, the asymptotic approximation to the MISE of the Gaussian kernel density estimator (for $d = 1$) is given by

$$\frac{1}{4} \sigma^4 \, \|f''\|^2 + \frac{1}{2n\sqrt{\pi\sigma^2}}, \tag{4}$$

where $\|f''\|^2 := \int (f''(x))^2 \, \mathrm{d}x$.

Optimal Bandwidth

The asymptotically optimal value of σ is the minimizer

$$\sigma^* := \left(\frac{1}{2n\sqrt{\pi}\,\|f''\|^2} \right)^{1/5}. \qquad (5)$$

To compute the optimal σ^* in (5), one needs to estimate the functional $\|f''\|^2$.

The Gaussian rule of thumb is to assume that f is the density of the $\mathcal{N}(\overline{x}, s^2)$ distribution, where $\overline{x}$ and s^2 are the sample mean and variance of the data, respectively .

In this case $\|f''\|^2 = s^{-5}\pi^{-1/2}3/8$ and the Gaussian rule of thumb becomes:

$$\sigma_{\text{rot}} = \left(\frac{4\,s^5}{3\,n} \right)^{1/5} \approx 1.06\; s\,n^{-1/5}.$$

Theta KDE

We recommend, however, the fast and reliable theta KDE, which chooses the bandwidth in an optimal way via a fixed-point procedure. The theta KDE source code is available as `kde.py` on the book's GitHub site.

This alleviates problems with traditional KDEs:

- For distributions on a bounded domain, such as the uniform distribution on $[0,1]^2$, the KDE assigns positive probability mass outside this domain.
- At the boundary of the support the density is not well estimated.

The following Python program draws an iid sample from the $\mathsf{Exp}(1)$ distribution and constructs a Gaussian kernel density estimate.

gausthetakde.py

```python
import matplotlib.pyplot as plt
import numpy as np
from kde import *

sig = 0.1; sig2 = sig**2; c = 1/np.sqrt(2*np.pi)/sig #Constants
phi = lambda x,x0: np.exp(-(x-x0)**2/(2*sig2)) #Unscaled Kernel
f = lambda x: np.exp(-x)*(x >= 0) # True PDF
n = 10**4 # Sample Size
x = -np.log(np.random.uniform(size=n))# Generate Data via IT method
xx = np.arange(-0.5,6,0.01, dtype = "d")# Plot Range
phis = np.zeros(len(xx))
for i in range(0,n):
    phis = phis + phi(xx,x[i])
phis = c*phis/n
plt.plot(xx,phis,'r')# Plot Gaussian KDE
[bandwidth,density,xmesh,cdf] = kde(x,2**12,0,max(x))
idx = (xmesh <= 6)
plt.plot(xmesh[idx],density[idx])# Plot Theta KDE
plt.plot(xx,f(xx))# Plot True PDF
```

No Boundary Effects

We see that with an appropriate choice of the bandwidth a good fit to the true pdf can be achieved, except at the boundary $x = 0$. The theta KDE does not exhibit this boundary effect. Moreover, it chooses the bandwidth automatically, to achieve a superior fit.

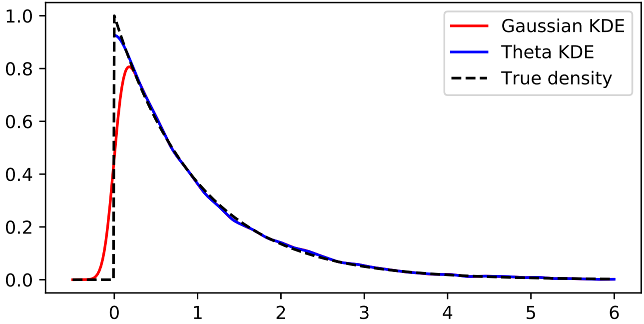


Figure: Kernel density estimates for $\mathsf{Exp}(1)$-distributed data.