Applications of Gaussian Distribution in Machine Learning

Girish Jayakumar

November 16, 2023

Gaussian Distribution

The intuition behind the derivation of the Gaussian distribution for a univariate case starts with the examination of the exponential function:

$$\begin{aligned} e^{x} &\quad \text{– Exponential growth} \\ e^{-x} &\quad \text{– Exponential decay} \\ e^{-|x|} &\quad \text{– Modulus causes symmetrical decay} \\ e^{-cx^2} &\quad \text{– where } c \text{ is a scaling factor} \end{aligned}$$

Normalization

To ensure that the Gaussian function represents a probability distribution, we need to normalize it so that its total area is equal to 1. This is achieved by scaling the function by a factor of $\frac{1}{\sigma\sqrt{2\pi}}$:

$$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{1}$$

Multivariate Gaussian Distribution

The Multivariate Gaussian Distribution is defined as:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right) \tag{2}$$

where:

- $\mathbf{x}$ is an n-dimensional vector (finite)
- $\boldsymbol{\mu}$ is an n-dimensional vector where each entry represents the mean of the Gaussian univariate distribution for a particular random variable X.
- $\boldsymbol{\Sigma}$ is the covariance matrix; each entry represents strength or correlation between 2 random variables ($n \times n$ matrix)

Temperature Prediction Problem

Let's say you've written down the temperature at three different times today, like this:

- At 1 pm: 15 degrees Celsius
- At 2 pm: 19 degrees Celsius
- At 3 pm: 14 degrees Celsius

Our goal is to find the temperature at time t = 1.5 (1:30 pm), and we will use a gaussian process to do so, involving the use of a kernel and covariance matrix (explained below).

Definitions

RBF Kernel (A Magic Measure)

This is a way of measuring how closely related two different times are, based on their temperatures. It uses a special dial called the "length scale" to adjust the measurement. The length scale specifies how strong the correlation drops from observation to observation (this allows us, for instance, to say 1 o'clock and 2 o'clock temperatures are more related than 1 o'clock and 3 o'clock temperatures).

Covariance Matrix (A Relationship Grid)

Think of this as a grid that helps us understand how each time we noted is connected to every other time, showing us a pattern in temperature changes.

How We Guess (The Prediction)

Using the relationship grid, we mix the temperatures we know and use them to make a guess for the temperature at 1:30. This is the nature of a Gaussian Process, which is simply a tool to predict values for data we do not have. The machine learning aspect comes into play when the Gaussian Process uses newly acquired data points from its old predictions to make better, more accurate predictions.

Visualizing the Guesswork (3D Picture)

If we were to draw this out, we'd have a 3D chart where the width shows the time, the depth shows the temperature, and the height shows how sure we are about our guess.

The Nitty-Gritty Details

For the math behind this problem, here it is in full:

RBF Kernel

By convention, the RBF kernel is used the most in a Gaussian Process when calculating the relationship between temperatures at different times. The "magic measure" we talked about is calculated like this:

$$k(t_i, t_j) = \exp\left(-\frac{(t_i - t_j)^2}{2l^2}\right) \tag{3}$$

Covariance Matrix

Our relationship grid looks something like this, after we've filled it in using our measure:

$$K = \begin{bmatrix} 1 & e^{-0.5} & e^{-2} \\ e^{-0.5} & 1 & e^{-0.5} \\ e^{-2} & e^{-0.5} & 1 \end{bmatrix} \tag{4}$$

Guessing the Temperature

To make our guess, we calculate the new relationship values for 1:30 like so:

$$k^T = \begin{bmatrix} e^{-0.125} & e^{-0.125} & e^{-1.125} \end{bmatrix} \tag{5}$$

And we update our grid to include this new time:

$$\Sigma = \begin{bmatrix} 1 & e^{-0.5} & e^{-2} & e^{-0.125} \\ e^{-0.5} & 1 & e^{-0.5} & e^{-0.125} \\ e^{-2} & e^{-0.5} & 1 & e^{-1.125} \\ e^{-0.125} & e^{-0.125} & e^{-1.125} & 1 \end{bmatrix} \tag{6}$$

To explain this in words, let's take the entry in row 1 column 2, which is $e^{-0.5}$. This represents the correlation between how similar temperatures are at 1 pm and 2 pm. If we were to take row 1 column 4, which is $e^{-0.125}$, this represents the correlation between how similar temperatures are at 1 pm and 1:30 pm. These values in the covariance matrix help with determining the mean and variance of temperatures at various times (in our case, predicting the temperature at time 1:30). If we wanted to predict the temperature at time t = 5 (5 p.m.), we would plug in 5 into the RBF kernel, obtain the relationship values, and update our grid to include this new time accordingly. This is how the covariance matrix expands to include relationships between temperatures at observed times and temperatures at predicted times.

Mean Temperature at t = 1.5

To find out what the temperature was at 1:30 pm using Gaussian Processes, we use the temperatures we recorded at three different times. Assuming we don't have any other information, we start with a mean function that is zero. However, if we know something about the temperature trend, we could start with a non-zero mean.

In our case, it doesn't make sense to start at mean 0 at every time t since the temperature cannot simply be assumed to be centered around 0 degrees. As such, by convention, the predicted mean temperature at a new time t can be calculated using the formula:

$$\mu(t) = K_*^T K^{-1}\mathbf{y}$$

Here's what each term means:

- K_* is a vector that tells us how the new time point $t = 1{:}30$ is related to the times we already know.
- K is a matrix that tells us how all the times we know are related to each other.
- $\mathbf{y}$ is a vector of the temperatures we've observed.

Using the temperatures:

- At $t = 1$, $y_1 = 15°C$,
- At $t = 2$, $y_2 = 19°C$,
- At $t = 3$, $y_3 = 14°C$,

we can calculate K_* using the Radial Basis Function (RBF) kernel with each observed time point.

The next step is to invert the matrix K (which we have from earlier calculations) and then multiply it by the vector K_* and our observed temperature vector $\mathbf{y}$. This will give us our predicted mean temperature at $t = 1.5$ or 1:30 pm. The specific values for K_* and K will depend on the length scale l used in the RBF kernel. It turns out that assuming l = 1, the mean we get is around 18.11 degrees. This makes sense as it lies in between the temperatures at time t = 1 and time t = 2.

Variance at t = 1.5

The prediction variance for temperature at time t = 1.5 is:

$$\sigma^2 = K_{**} - K_\theta (K_{\theta\theta})^{-1} K_\theta^T \tag{7}$$

Definitions for the Prediction Variance Equation

The equation you're looking at is all about figuring out how much we can trust our temperature prediction for a time we didn't measure. Here's what each symbol means:

- σ^2: This represents the variance of our prediction. It's a number that tells us how certain or uncertain our guess at the temperature is. The bigger this number, the less sure we are.
- K_{**}: This is the variance we expect at the new time point we're interested in. It comes from our magic measure when we compare the time to itself.
- K_θ: This represents how the time we want to predict (1:30 pm) is related to the times we've already measured.
- $K_{\theta\theta}$: This is our relationship grid for the times we've actually measured. It's like a mirror that shows us how they all relate to each other.

- $(K_{\theta\theta})^{-1}$: This is the inverse of $K_{\theta\theta}$
- K_{θ}^{T}: This is the transposed version of K_{θ}.

And when we put it all together in the equation, it gives us a number that says, "Based on the temperatures we know, this is how sure we are about our guess at 1:30 pm." It turns out the variance for our specific scenario is equivalent to 0.018.

3D Visualization

If we wanted to show this visually, here's a figure showing a simulation of our prediction. At time t = 1.5 = 1:30 pm, we calculated that the mean = 18.11 degrees and that the variance = 0.018. Therefore, at time t = 1.5, there will be a univariate gaussian distribution centering around 18.11 degrees and variance 0.018. This process can be repeated for other times as well, resulting in a collection of univariate gaussian distributions. This is what causes the 3-D nature of the multivariate gaussian distribution.

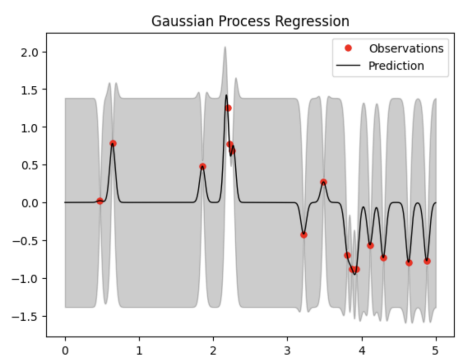

Figure 1: Example Python Matplotlib Simulation of Gaussian Process using Time and Temperature as inputs.

Inherent Flaw (The Catch)

You might wonder how we can predict temperatures at times we didn't measure. Well, we do this by making a series of educated guesses at certain times and then connecting the dots to see the bigger picture.

Application in Real Life (Practical Use)

In real-life projects, like making sure self-driving cars know which way to go, this math helps us predict things like traffic conditions.

Conclusion (The Takeaway)

This whole process is about making smart guesses for things we don't know for sure, like temperature at different times, and it's super useful for things like planning the paths for self-driving cars.

Problems for Practice

Problem 1: Basic Probability Calculation

Given a univariate Gaussian distribution with mean $\mu = 0$ and standard deviation $\sigma = 2$, calculate the probability of the random variable X falling between -1 and 1.

Problem 2: Covariance Matrix Analysis

Consider a bivariate Gaussian distribution with the following covariance matrix:

$$\Sigma = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 2 \end{bmatrix}$$

So far, we have looked at how to construct the covariance matrix. Now, let's see if we can find the strength of the relationship. See if you can find the correlation between the two variables and comment on the strength of their relationship.

Problem 3: Gaussian Process Prediction

Using the Radial Basis Function (RBF) kernel and the covariance matrix provided, calculate the Gaussian Process prediction and the confidence interval for a temperature at $t = 4$ given the following observed temperatures:

- $t = 1$: $15°C$
- $t = 2$: $18°C$
- $t = 3$: $20°C$

Assume a length scale $l = 1$ for the RBF kernel.

Problem 4: Designing a Kernel Function

Design your own kernel function that could potentially capture periodic patterns in temperature data, such as daily temperature cycles. Describe its form and explain the choice of parameters.

Answers to Practice Problems

Answer to Problem 1

The probability of a univariate Gaussian random variable X falling between -1 and 1 is found by calculating the integral of the probability density function from -1 to 1:

$$P(-1 \leq X \leq 1) = \int_{-1}^{1} \frac{1}{2\sqrt{2\pi}} e^{-\frac{x^2}{8}} dx$$

This integral can be evaluated using standard Gaussian tables or a computational tool.

Answer to Problem 2

The correlation coefficient ρ between two random variables in a bivariate Gaussian distribution is the off-diagonal element of the covariance matrix divided by the product of the standard deviations of the two variables:

$$\rho = \frac{\Sigma_{12}}{\sqrt{\Sigma_{11}\Sigma_{22}}} = \frac{0.5}{\sqrt{1 \cdot 2}} = \frac{0.5}{\sqrt{2}} \approx 0.354$$

A correlation of 0.354 indicates a moderate positive linear relationship between the two variables.

Answer to Problem 3

The Gaussian Process prediction for a temperature at $t = 4$ is calculated using the kernel function and the covariance matrix. Assuming the mean of the predictions is 0 (common assumption in GP), the prediction is the product of the inverse of the covariance matrix K and the vector k. The confidence interval can be calculated from the predictive variance, which is the diagonal of the matrix $K_{**} - K(X^*, X)(K(X, X))^{-1}K(X^*, X)^T$.

Answer to Problem 4

Answers may vary: A possible kernel function capturing periodic patterns could be the Periodic Kernel defined as:

$$k(t_i, t_j) = \exp\left(-\frac{2\sin^2(\pi|t_i - t_j|/p)}{l^2}\right)$$