Statistical Techniques in Robotics (16-831, F10) Lecture#18 (Thursday October 21)

Bayesian Linear Regression Part 2 & Gaussian Properties

Lecturer: Drew Bagnell *Scribe: Heather Justice* [1]

1 Bayesian Linear Regression

1.1 Problem Formulation

Recall from the previous lecture our formulation of the Bayesian Linear Regression problem, with the corresponding graphical model shown in Figure 1. The variables $x_0, ..., x_t$ are our input feature vectors. The weight vector is $\theta \sim \mathcal{N}(\mu_\theta, \Sigma_\theta)$. Our prediction is based on:

$$y_t \sim \mathcal{N}(\theta^T x_t, \sigma^2) = \theta^T x_t + \epsilon, \text{ where } \epsilon \sim \mathcal{N}(0, \sigma^2)$$

We need to compute $P(\theta|D)$, where D is all data $x_0, ..., x_t$ and $y_0, ..., y_t$, so that we can make a prediction for y_{t+1} by conditioning on θ as follows:

$$P(y_{t+1}|D) = \int_\theta P(y_{t+1}|x_{t+1}, \theta) P(\theta|D) \delta\theta$$

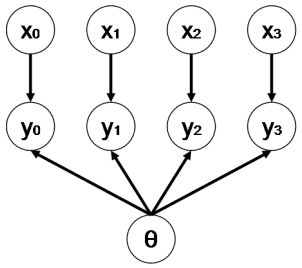

Figure 1: Bayesian Linear Regression Model

1.2 Update Step

Recall from the previous lecture that we can write the weight vector in the natural parameterization:

$$\theta \sim \tilde{\mathcal{N}}(J_\theta, P_\theta) = \frac{1}{z} \exp\left(-\frac{1}{2}\theta^T P \theta + J^T \theta\right)$$

We do the update step as follows (as derived in the previous lecture):

$$\begin{aligned} J_{t+1} &= J_t + \frac{1}{\sigma^2} y_t x_t \\ P_{t+1} &= P_t + \frac{1}{\sigma^2} x_t x_t^T \end{aligned}$$

[1] Some content adapted from previous scribes: Laura Lindzey, Hans Pirnay, Siddharth Mehrotra

Notice that, due to the properties of Gaussians, the precision is always increasing for each new data point (in other words, the variance never increases). Also notice that the update for P has no dependence on y_t (the variance depends only on inputs and not outputs); this model does not account for surprises in the data! Gaussians don't really have a good model for outliers. In this model, if an outlier is observed, that will essentially shift the mean, and the algorithm will become more certain about that new mean. In practice, you may want to have some sort of wrapper function to detect and discard outliers from the data first (imagine if the data contained an outlier 10σ from the mean!).

1.3 Transfer to Moment Parameterization

Given the transfer rules to the moment parameterization

$$\begin{aligned} \Sigma &= P^{-1} \\ \mu &= P^{-1}J \end{aligned}$$

the moment parameterization after N timesteps is then

$$\begin{aligned} \Sigma_{\theta|D} &= \left[\Sigma_0^{-1} + \sum_{i=1}^{N} \frac{x_i x_i^T}{\sigma^2}\right]^{-1} \\ \mu_\theta &= \Sigma_{\theta|D} \cdot \left[\sum_{t=1}^{N} \frac{y_t x_t}{\sigma^2}\right] \\ &= \left[\Sigma_0^{-1} + \sum_{i=1}^{N} \frac{x_i x_i^T}{\sigma^2}\right]^{-1} \cdot \left[\sum_{t=1}^{N} \frac{y_t x_t}{\sigma^2}\right] \end{aligned}$$

Watch out for the different uses of the sigma symbol (variance versus summation)! Also note that this assumes the initial $\mu_0 = 0$; if μ_0 is nonzero, then the μ_θ update will be more complicated.

The complexity of this this computation is essentially cubed in the number of features. More precisely, $\mathcal{O}(F^3 + TF^2)$, where F is the number of features and T is the number of data points.

1.4 Making Predictions

Given all data D up to timestep t and x_{t+1}, the probability of an observation $\tilde{y}_{t+1}$ is

$$\begin{aligned} p(\tilde{y}_{t+1}|x_{t+1}, D) &= \int p(\tilde{y}_{t+1}|x_{t+1}, D, \theta) \cdot p(\theta|D) d\theta \quad (1) \\ &= \int p(\tilde{y}_{t+1}|x_{t+1}, \theta) \cdot p(\theta, D) d\theta \quad (2) \end{aligned}$$

We might expect that $E[y_{t+1}|D] = \mu_\theta^T x_{t+1}$ since we know that $E[\theta] = \mu_\theta$ and we want θ such that

$y_{t+1} = \theta^T x_{t+1}$. More formally, we can compute:

$$\begin{aligned} E_{P(\theta|D)}[y_{t+1}|x_{t+1}] &= E[\theta^T x_{t+1} + \epsilon] \\ &= E[\theta^T x_{t+1}] + E[\epsilon] \\ &= E[\theta]^T x_{t+1} + 0 \text{ (since that } \epsilon \text{ is independent of } \theta) \\ &= \mu_\theta^T x_{t+1} \end{aligned}$$

Notice that inference is exact for this model!

2 Properties of Gaussian Parameterizations

There are two common parameterizations for Gaussians, the moment parameterization and the natural parameterization. It is often most practical to switch back and forth between representations, depending on which calculations are needed. The moment parameterization is more convenient for visualization (simply draw a Gaussian centered around the mean with width determined by the variance), calculating expected value, and computing marginals. The natural parameterization is more convenient for multiplying Gaussians and for conditioning on known variables.

2.1 Moment Parameterization

Recall that the moment parameterization of a Gaussian is:

$$\mathcal{N}(\mu, \Sigma) = p(\theta) = \frac{1}{z} \exp\left(-\frac{1}{2}(\theta - \mu)^T \Sigma^{-1} (\theta - \mu)\right) \tag{3}$$

Given:

$$\mathcal{N}\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$$

Marginal: computing $p(x_2)$

$$\begin{aligned} \mu_2^{\text{marg}} &= \mu_2 \\ \Sigma_2^{\text{marg}} &= \Sigma_{22} \end{aligned}$$

Conditional: computing $p(x_1|x_2)$

$$\begin{aligned} \mu_{1|2} &= \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2) \\ \Sigma_{1|2} &= \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} \end{aligned}$$

2.2 Natural Parameterization

Recall that the natural parameterization of a Gaussian is:

$$\tilde{\mathcal{N}}(J, P) = \tilde{p}(\theta) = \frac{1}{z} \exp\left(J^T\theta - \frac{1}{2}\theta^T P \theta\right) \tag{4}$$

Given:

$$\mathcal{N}\left(\begin{bmatrix} J_1 \\ J_2 \end{bmatrix}, \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}\right)$$

Marginal: computing $p(x_2)$

$$\begin{aligned} J_2^{\text{marg}} &= J_2 - P_{21}P_{11}^{-1}J_1 \\ P_2^{\text{marg}} &= P_{22} - P_{21}P_{11}^{-1}P_{12} \end{aligned}$$

Conditional: computing $p(x_1|x_2)$

$$\begin{aligned} J_{1|2} &= J_1 - P_{12}x_2 \\ P_{1|2} &= P_{11} \end{aligned}$$

3 Introduction to the Gauss-Markov Model

Consider $X_1, X_2, ..., X_t, X_{t+1}$ to be the state variables and $Y_1, Y_2, ..., Y_t, Y_{t+1}$ to be the sequence of corresponding observations. As in Hidden Markov models, conditional independencies (see Figure 2) dictate that past and future states are decorrelated given the current state, X_t at time t. For example, if we know what X_2 is, then no information about X_1 can possibly help us to reason about what X_3 should be.

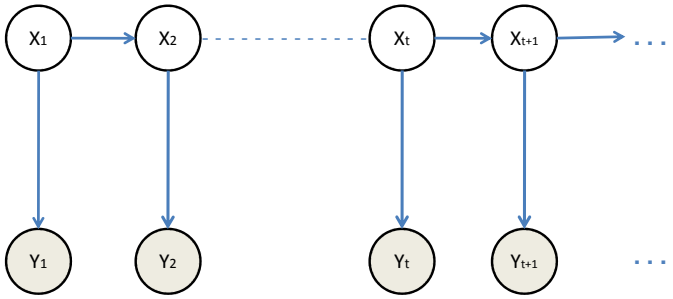

Figure 2: The Independence Diagram of a Gauss-Markov Model

Note that we want to be able to preserve Gaussian properties in this model, so our motion and observation models must be linear systems.

Prior:

$$x_1 \sim \mathcal{N}(\mu_0, \Sigma_0)$$

Motion Model:

$$x_{t+1} = Ax_t + \epsilon, \text{ where } \epsilon \sim \mathcal{N}(0, Q)$$

(Note that A could be indexed by time, but for simplicity we will not consider that case here.)

Observation Model:

$$y_{t+1} = Cx_{t+1} + \delta, \text{ where } \delta \sim \mathcal{N}(0, R)$$

Ultimately we want to be able to compute $P(x_t|y_1, ..., y_t)$. For example, we might want to compute the robot's location (x_t) at some time t given the robot's observations ($y_1, ..., y_t$).

Introduction to the Lazy Gauss-Markov Filter

At time step t, given that we have already computed $P(x_t|D) \sim \mathcal{N}(\mu_t, \Sigma_t)$, the prediction (or motion update) is easiest to do in the moment form of the Gaussian.

The form of this update is $P(x_{t+1}|D) \sim \mathcal{N}(A\mu_t, ???)$. Note that we will be deriving the variance "???" in a future homework. We can easily prove the expected value as follows:

$$E[X_{t+1}] = E[AX_t + \epsilon]$$

$$\Rightarrow E[X_{t+1}] = E[AX_t] + E[\epsilon]$$

since variance of ϵ is 0,

$$\Rightarrow E[X_{t+1}] = AE[X_t] = A\mu_t$$

Further details about the Lazy Gauss-Markov Filter will be explored in the next lecture!