

Integration

Cheng Soon Ong

Data61, CSIRO

`chengsoon.ong@anu.edu.au`

 @ChengSoonOng

Marc Peter Deisenroth

University College London

`m.deisenroth@ucl.ac.uk`

 @mpd37

December 2020

Integration problems

I Moment computation

$$M_k(x) = \int x^k p(x) dx$$

Integration problems

I Moment computation

$$M_k(x) = \int x^k p(x) dx$$

I Evidence (marginal likelihood)

$$p(\mathbf{X}) = \int p(\mathbf{X}^j) p(\theta) d\theta$$

Integration problems

I Moment computation

$$M_k(\mathbf{x}) = \int \mathbf{x}^k p(\mathbf{x}) d\mathbf{x}$$

I Evidence (marginal likelihood)

$$p(\mathbf{X}) = \int p(\mathbf{X}^j) p(\cdot) d\cdot$$

I Relative entropy (KL divergence)

$$\text{KL}(p||q) = \int \log \frac{p(\mathbf{x})}{q(\mathbf{x})} p(\mathbf{x}) d\mathbf{x}$$

Integration problems

I Moment computation

$$M_k(\mathbf{x}) = \int \mathbf{x}^k p(\mathbf{x}) d\mathbf{x}$$

I Evidence (marginal likelihood)

$$p(\mathbf{X}) = \int p(\mathbf{X} | \mathbf{j}) p(\mathbf{j}) d\mathbf{j}$$

I Relative entropy (KL divergence)

$$\text{KL}(p || q) = \int \log \frac{p(\mathbf{x})}{q(\mathbf{x})} p(\mathbf{x}) d\mathbf{x}$$

I Prediction in time-series models

$$p(\mathbf{x}_{t+1}) = \int p(\mathbf{x}_{t+1} | \mathbf{x}_t) p(\mathbf{x}_t) d\mathbf{x}_t$$

Integration problems

I Moment computation

$$M_k(\mathbf{x}) = \int \mathbf{x}^k p(\mathbf{x}) d\mathbf{x}$$

I Evidence (marginal likelihood)

$$p(\mathbf{X}) = \int p(\mathbf{X} | \mathbf{y}) p(\boldsymbol{\theta}) d\boldsymbol{\theta}$$

I Relative entropy (KL divergence)

$$\text{KL}(p || q) = \int \log \frac{p(\mathbf{x})}{q(\mathbf{x})} p(\mathbf{x}) d\mathbf{x}$$

I Prediction in time-series models

$$p(\mathbf{x}_{t+1}) = \int p(\mathbf{x}_{t+1} | \mathbf{x}_t) p(\mathbf{x}_t) d\mathbf{x}_t$$

I Experimental design

$$\mathbb{E} [U(\mathbf{x})] = \int U(\mathbf{x}; \boldsymbol{\theta}) p(\boldsymbol{\theta}) d\boldsymbol{\theta}$$

Integration problems

I Moment computation

$$M_k(\mathbf{x}) = \int \mathbf{x}^k p(\mathbf{x}) d\mathbf{x}$$

I Evidence (marginal likelihood)

$$p(\mathbf{X}) = \int p(\mathbf{X} | \mathbf{j}) p(\mathbf{j}) d\mathbf{j}$$

I Relative entropy (KL divergence)

$$\text{KL}(p || q) = \int \log \frac{p(\mathbf{x})}{q(\mathbf{x})} p(\mathbf{x}) d\mathbf{x}$$

I Prediction in time-series models

$$p(\mathbf{x}_{t+1}) = \int p(\mathbf{x}_{t+1} | \mathbf{j}, \mathbf{x}_t) p(\mathbf{j}, \mathbf{x}_t) d\mathbf{j}, \mathbf{x}_t$$

I Experimental design

$$\mathbb{E}[U(\mathbf{x})] = \int U(\mathbf{x}; \mathbf{j}) p(\mathbf{j}) d\mathbf{j}$$

I Planning

$$J(\mathbf{x}(0)) = \int_0^T r(\mathbf{x}(t); \mathbf{u}(t)) p(\mathbf{x}(t) | \mathbf{x}(0)) dt$$

Exact integration

- I Compute integrals analytically, if possible (Gradshteyn & Ryzhik, 2007)

Exact integration

- I Compute integrals analytically, if possible (Gradshteyn & Ryzhik, 2007)

Approximate integration

Topic	Useful reference	Video
Numerical integration	Stoer & Bulirsch (2002)	Chapter 1
Bayesian quadrature	Rasmussen & Ghahramani (2003), Gunter et al. (2014)	Chapter 1
Monte-Carlo integration	MacKay (2003), Murray (2015)	Chapter 2
Normalizing flows	Weng (2018), Papamakarios et al. (2019), Kobyzev et al. (2020)	Chapter 3
Inference in time series	Julier & Uhlmann (2004), Särkkä (2013)	Chapter 4

References

- Gradshteyn, I. S. and Ryzhik, I. M. (2007). *Table of Integrals, Series, and Products*. Academic Press, 7th edition.
- Gunter, T., Osborne, M. A., Garnett, R., Hennig, P., and Roberts, S. J. (2014). Sampling for Inference in Probabilistic Models with Fast Bayesian Quadrature. In *Advances in Neural Information Processing Systems*.
- Julier, S. J. and Uhlmann, J. K. (2004). Unscented Filtering and Nonlinear Estimation. *Proceedings of the IEEE*, 92(3):401–422.
- Kobyzev, I., Prince, S., and Brubaker, M. (2020). Normalizing Flows: An Introduction and Review of Current Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- MacKay, D. J. C. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.
- Murray, I. (2015). Monte Carlo Inference Methods. *NeurIPS Tutorial*.

References (cont.)

- Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. (2019). Normalizing Flows for Probabilistic Modeling and Inference. *arXiv:1912.02762*.
- Rasmussen, C. E. and Ghahramani, Z. (2003). Bayesian Monte Carlo. In *Advances in Neural Information Processing Systems*.
- Särkkä, S., Solin, A., and Hartikainen, J. (2013). Spatiotemporal Learning via Infinite-Dimensional Bayesian Filtering and Smoothing: A Look at Gaussian Process Regression Through Kalman Filtering,. *IEEE Signal Processing Magazine*, 30(4):51–61.
- Stoer, J. and Bulirsch, R. (2002). *Introduction to Numerical Analysis*. Texts in Applied Mathematics. Springer-Verlag, 3rd edition.
- Weng, L. (2018). Flow-based Deep Generative Models. *lilianweng.github.io/lil-log*.