



Bookmarks

- ▶ [Introduction](#)
- ▶ [Part 1: Probability and Inference](#)
- ▶ [Part 2: Inference in Graphical Models](#)
- ▶ [Part 3: Learning Probabilistic Models](#)
- ▶ [Final Project](#)
- ▼ [Epilogue](#)

[What's Next?](#)[MITx Exit Survey](#)

Epilogue &gt; What's Next? &gt; What's Next?

## What's Next?

[Bookmark this page](#)

### WHAT'S NEXT?

In 6.008.1x, we only scratched the surface of what's out there for probability and inference. We took a computational approach, where almost every topic we encountered had Python code to go with it, where initially we supplied much of the code and then by parts 2 and 3 of the course, you were providing most of the code in the form of projects. We also sketched out how to derive the different algorithms we encountered (thus providing some justification for why they are correct), and how to analyze their running times. Of course, we also saw how these algorithms or their variants can be used to solve a variety of real-world problems (well...and problems involving wizards, aliens, and other standbys).

There is an enormous amount of material that builds off what we've covered in 6.008.1x. Below are just a few examples. This listing is hardly exhaustive!

**Part 1: Probability and Inference.** We mainly focused on finite probability spaces and finite random variables and briefly mentioned discrete probability spaces and discrete random variables. Of course, there are also continuous random variables (such as the normal/Gaussian distribution informally called the "bell curve", where the sample space is the entire real number line), and more general random

variables (for example, a random variable where with some probability it takes on the value of a discrete random variable and with some probability it takes on the value of a continuous random variable).

As for inference, we only provided a cursory introduction with the help of Bayes' theorem, skirting discussion on the two major schools of thought of frequentist and Bayesian inference, where the former views the quantity we are inferring as fixed but unknown, and the latter views the quantity we are inferring as random. These two schools of thought lead to very different interpretations of inference procedures!

**Part 2: Inference in Graphical Models.** Undirected graphical models can have loops, which introduces a variety of complications that in turn can be addressed by a variety of algorithms. Here, exact inference algorithms can be extremely computationally expensive, demanding that we settle for approximate inference algorithms, such as algorithms based on random sampling. A famous example of this is Markov chain Monte Carlo.

Also, we can have graphical models for continuous random variables. When the random variables are all Gaussian (and specifically what are called jointly-distributed Gaussian random variables), the Sum-Product algorithm can be written in a simple way to produce an algorithm called "Gaussian belief propagation", which specialized to Gaussian hidden Markov models leads to the Kalman filter (which just does the forward pass of the forward-backward algorithm).

**Part 3: Learning Probabilistic Models.** There are many complications to learning probabilistic models. For example, we have always assumed that every training data point has an observed value for every random variable in the graphical model of interest. What if certain random variables are never observed? For example, for HMM's, the latent variables are generally never observed, not even in the training data. The algorithm we presented for learning parameters of a graphical model would not be able to handle this case. What we can do is "fill-in" missing values. A general algorithm for doing this

procedure is the expectation-maximization (EM) algorithm, which finds approximate maximum likelihood estimates when certain random variables aren't observed. Of course, there are many other possible complications, such as different variables being missing across different training data points, the training data points not being i.i.d., or even us having control over which training data to collect (i.e., we get to query for what the next training data point should be that someone then manually labels, and we repeatedly do this to build up a training dataset and to learn a model).

Separately, there is a question of what we can say about guarantees on learning procedures. For example, how much training data should we collect before we can achieve some tolerated probability of error? One of the major trends in analysis of inference algorithms is to trade off running time, quality of inference (for example, in a classification task, the quality of inference could be measured by misclassification rate), and amount of training data. For example, in a classification task, if we don't have much training data, maybe we are okay with having the running time of an inference procedure be very large to achieve a small misclassification rate. There are a variety of tools for providing theoretical performance guarantees on inference algorithms, much of which draws on ideas from information theory. A subset of the results here are referred to as "asymptotic" guarantees on inference procedures in that certain quantities (such as the number of training data) are taken to go to infinity.

**Courses.** At MIT, the residential version of 6.008 covers not only the material in 6.008.1x but also basics of asymptotic analysis of inference procedures, random sampling for approximate inference, and an introduction to inference with continuous random variables (which is largely a re-hash of everything else in the class now with a slight twist as we use continuous random variables instead of discrete random variables).

A few other courses at MIT that build on 6.008 and that are available online:

- 6.041 Probabilistic Systems Analysis (available on edX and OCW): a much more thorough introduction to part 1 of 6.008.1x although without the coding
- 6.436 Fundamentals of Probability (available on OCW): a graduate-level measure-theoretic introduction to probability that more rigorously/formally explains why probability theory works; less emphasis than 6.041 on solving engineering/word problems and more emphasis on proofs/theoretical analysis
- 6.438 Algorithms for Inference (available on OCW): a graduate-level introduction to probabilistic graphical models (heavily builds on parts 2 and 3 of 6.008.1x)
- 6.867 Machine Learning (available on OCW): a graduate-level introduction to machine learning (covers a variety of topics, only partly overlapping with 6.008.1x)
- 9.520 Statistical Learning Theory and Applications (available on OCW): a graduate-level introduction to statistical learning theory (largely about theoretical guarantees on learning procedures)

A few textbooks that may be of interest (again, not exhaustive):

- Dimitri P. Bertsekas and John N. Tsitsiklis. *Introduction to Probability*. 2nd edition. Athena Scientific 2008.
- Geoffrey R. Grimmett and David R. Stirzaker. *Probability and Random Processes*. 3rd edition. Oxford University Press 2001.
- Christopher Bishop. *Pattern Recognition and Machine Learning*. Springer 2007.
- Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press 2009.

- David MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press 2003. (Freely available off the official textbook website.)
- Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning with Applications in R*. Springer-Verlag 2013. (Freely available off the official textbook website.)
- Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag 2009. (Freely available off the official textbook website.)

Of course, MIT isn't the only place providing related courses and resources. Please keep the discussion forum going (at least until December 31, 2016, 5pm Eastern Time when the forum closes) and share what resources you have found useful!

Cheers,  
George Chen (on behalf of the 6.008.1x staff)

### Discussion

**Topic:** Epilogue / What's Next?

[Show Discussion](#)

© All Rights Reserved



© 2016 edX Inc. All rights reserved except where noted. EdX, Open edX and the edX and Open EdX logos are registered trademarks or trademarks of edX Inc.

