Bayesian Learning for Computer graphics
Aaron Hertzmann
University of Toronto

Computers are really fast
• If you can create it, you can render it

How do you create it?
Digital Michaelangelo Project

Two problems
• How do we get the data into the computer?
• How do we manipulate it?

Data fitting

Key questions
• How do you fit a model to data?
  - How do you choose weights and thresholds?
  - How do you incorporate prior knowledge?
  - How do you merge multiple sources of information?
  - How do you model uncertainty?

Bayesian reasoning provides a solution
Talk outline

• Bayesian reasoning
• Facial modeling example
• Non-rigid modeling from video

What is reasoning?

• How do people reason?
• How should computers do it?

Aristotelian Logic

• If A is true, then B is true
• A is true
• Therefore, B is true

A: My car was stolen
B: My car isn’t where I left it

Real-world is uncertain

Problems with pure logic
• Don’t have perfect information
• Don’t really know the model
• Model is non-deterministic

Beliefs

• Let B(X) = “belief in X”,
• B(¬X) = “belief in not X”

1. An ordering of beliefs exists
2. B(X) = f(B(¬X))
3. B(X) = g(B(X|Y),B(Y))

Cox axioms


\[
p(\text{TRUE}) = 1 \\
p(A) = \frac{f_B}{p_B} p(A; B) \\
p(A; B) = p(A|B)p(B)
\]
“Probability theory is nothing more than common sense reduced to calculation.”

- Pierre-Simon Laplace, 1814

Bayesian vs. Frequentist

- Frequentist (“Orthodox”):
  - Probability = percentage of events in infinite trials
- Medicine, biology: Frequentist
- Astronomy, geology, EE, computer vision: largely Bayesian

Learning in a nutshell

- Create a mathematical model
- Get data
- Solve for unknowns

Face modeling


Generative model

- Faces come from a Gaussian
  \[
  p(S | \theta; \beta) = \frac{1}{(2\pi \beta^2)^{n/2}} e^{-\frac{1}{2\beta^2} (S-S^\theta)^T (S-S^\theta)}
  \]
  \[
  p(S | g; \beta) = \sim \mathcal{N}(S | g, \beta)
  \]

Bayes Rule

\[
P(A; B) = p(A|B)p(B) = p(B|A)p(A)
\]

Often:

\[
p(\text{model | data}) / p(\text{data | model})p(\text{model})
\]
Learning a Gaussian

\[
\arg\max_{\theta, \beta} p(\theta | S, \beta) = \arg\max_{\theta, \beta} p(f(S, \theta) | \beta)
\]

\[
= \arg\max_{\theta, \beta} \exp\left\{ \frac{1}{2} \| S - \beta \|^2 \right\} p(\beta) p(\theta | S, \beta)
\]

\[
\theta \sim \mathcal{N}(0, I)
\]

\[
\beta \sim \mathcal{N}(0, I)
\]

Maximization trick

\[\text{Maximize} \quad p(x) \quad \iff \quad \text{minimize} \quad \ln p(x)\]

Fitting a face to an image

Generative model

\[
p(S | \beta) = \frac{1}{2\pi} \exp\left\{ -\frac{1}{2} \| S - \beta \|^2 \right\}
\]

\[
I = \text{Render}(S; \hat{\beta}) + n
\]

\[
\ln p(I | S; \hat{\beta}; \beta) = \frac{1}{2} \| S - \beta \|^2 + \ln \text{Render}(S; \hat{\beta}) + \ln \beta + \mathcal{N}(\beta, 2\pi^{-\frac{1}{2}})
\]

Why does it work?

\[
p(f(x | \hat{g}; \beta) \mid \hat{g} | \hat{\beta}) = \hat{p}(x | \hat{g}; \beta)
\]

\[
\text{s.t.} \quad \hat{p}(x | \hat{g}; \beta) = 1
\]

General features

- Models uncertainty
- Applies to any generative model
- Merge multiple sources of information
- Learn all the parameters
Caveats
- Still need to understand the model
- Not necessarily tractable
- Potentially more involved than ad hoc methods

Applications in graphics
- Shape and motion capture
- Learning styles and generating new data

Learning Non-Rigid 3D Shape from 2D Motion
Joint work with Lorenzo Torresani and Chris Bregler (Stanford, NYU)

Camera geometry
Orthographic projection
\[ p_{\hat{\alpha}} = R_{\hat{\alpha}}S_{\hat{\alpha}} + T_{\hat{\alpha}} \]

Non-rigid reconstruction
Input: 2D point tracks
Output: 3D nonrigid motion
\[ p_t = R_tS_t + T_t \]

Totally ambiguous!

Shape reconstruction
Least-squares version
\[ p_t = R_tS_t + T_t \]
minimize
\[ \sum_j (\hat{p}_t \hat{R}_t \hat{S}_t \hat{T}_t - p_j)^2 + \lambda \sum_j (\hat{S}_t - S_\beta)^2 \]
Bayesian formulation

\[ p(S_j; \theta) = \frac{1}{(2\pi)^{\frac{1}{2}}} \exp(-\frac{1}{2}(S_j - \mu)^T \Sigma^{-1} (S_j - \mu)) \]

\[ p(S_{\theta} | \theta) = R_S + T + n \quad n \sim N(0; \Sigma) \]

\[ \text{maximize} \quad p(R; S; T; S_{\theta} | \theta) \]

\[ \text{Minimize} \quad \frac{1}{2} \sum_{i} \ln p(P R; S; T) \]

How does it work?

Maximize

\[ p(R; S; T; S_{\theta} | \theta) / p(P R; S; T) p(S_{\theta} | \theta) \]

Minimize

\[ \ln p(P R; S; T) \]

Results

Input
Least-squares, no reg.
Least-squares, reg.
Gaussian
LDS

Conclusions

- Bayesian methods provide a unified framework
- Build a model, and reason about it
- The future of data-driven graphics