Assorted Topics (2)

CS772A: Probabilistic Machine Learning

Piyush Rai

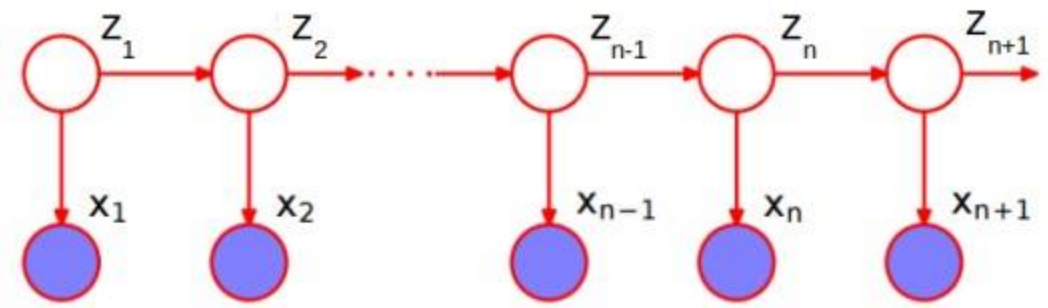
Plan today

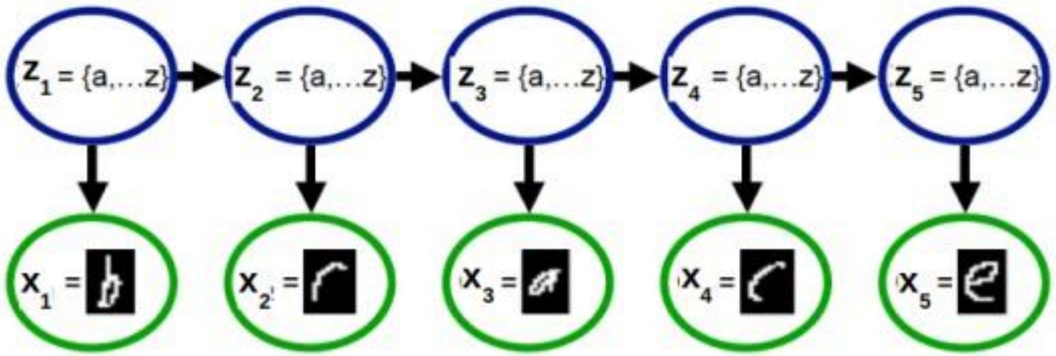
- Some classical probabilistic models for sequential data
  - Hidden Markov Models (HMM) and State-Space Models (SSM)
- Another non-Bayesian way to get uncertainty estimates:
  - Conformal Prediction
- Simulation based inference

Probabilistic Models for Sequential Data

Latent Variable Models for Sequential Data

- Task: Given a sequence of observations, infer the latent state of each observation





Observation model

State-transition model

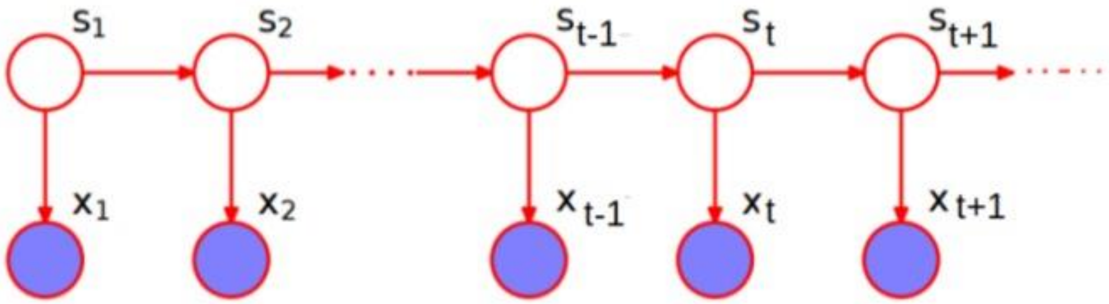
$$\boldsymbol{x}_n|\boldsymbol{z}_n \sim p(\boldsymbol{x}_n|\boldsymbol{z}_n) \quad \text{(i.i.d. draws of } \boldsymbol{x}_n \text{ given } \boldsymbol{z}_n\text{)}$$

$$\boldsymbol{z}_n|\boldsymbol{z}_{n-1} \sim p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) \quad \text{(first-order dependence b/w } \boldsymbol{z}_n\text{'s)}$$

- If z_n's are discrete, we have a hidden Markov model (HMM) $p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}=\ell) = \text{multinoulli}(\boldsymbol{\pi}_\ell)$
- If z_n's are real-valued, we have a state-space model (SSM) $p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) = \mathcal{N}(\mathbf{A}\boldsymbol{z}_{n-1}, \mathbf{I}_K)$

State-Space Models

- In the most general form, the state-transition and observation models of an SSM



Using ‘**s**’ instead of ‘**z**’ to refer to states

Using ‘t’ to denote the ‘time-step’

HMM is similar to SSM except the state-transition model is a discrete distribution

g_t, h_t can be linear or nonlinear functions

$$\boldsymbol{s}_t|\boldsymbol{s}_{t-1} = g_t(\boldsymbol{s}_{t-1}) + \epsilon_t \quad \text{(must be a cont. dist. over } \boldsymbol{s}_t\text{)}$$

$$\boldsymbol{x}_t|\boldsymbol{s}_t = h_t(\boldsymbol{s}_t) + \delta_t \quad \text{(can be any dist. over } \boldsymbol{x}_t\text{)}$$

- Assuming Gaussian noise in the state-transition and observation models

This is a Gaussian SSM

$$\boldsymbol{s}_t|\boldsymbol{s}_{t-1} \sim \mathcal{N}(\boldsymbol{s}_t|g_t(\boldsymbol{s}_{t-1}), \mathbf{Q}_t)$$

$$\boldsymbol{x}_t|\boldsymbol{s}_t \sim \mathcal{N}(\boldsymbol{x}_t|h_t(\boldsymbol{s}_t), \mathbf{R}_t)$$

If g_t, h_t, Q_t, R_t are independent of t then it is called a stationary model

g_t, h_t, Q_t, R_t may be known or can be learned

State-Space Models: A Simple Example

- Consider the linear Gaussian SSM

$$\boldsymbol{s}_t | \boldsymbol{s}_{t-1} = \mathbf{A}_t \boldsymbol{s}_{t-1} + \epsilon_t$$

$$\boldsymbol{x}_t | \boldsymbol{s}_t = \mathbf{B}_t \boldsymbol{s}_t + \delta_t$$

- Suppose $\boldsymbol{x}_t \in \mathbb{R}^2$ denotes the (noisy) observed 2D location of an object
- Suppose $\boldsymbol{s}_t \in \mathbb{R}^6$ denotes the "state" vector

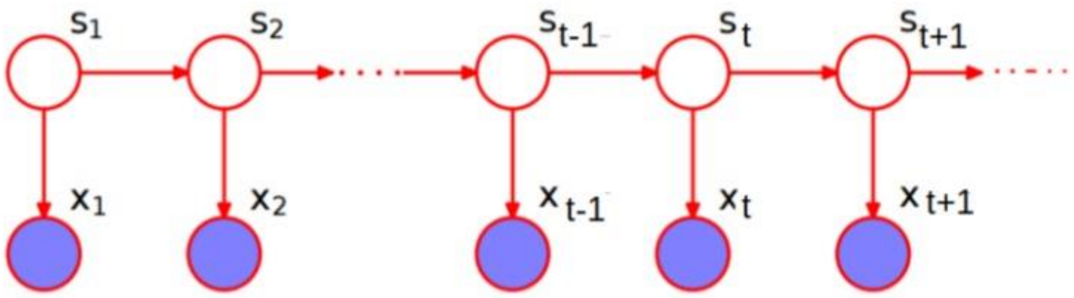
$\boldsymbol{s}_t$ = [pos1, vel1, accel1, pos2, vel2, accel2]

- Here is an example SSM for this problem with pre-defined $\mathbf{A_t}$ and $\mathbf{B_t}$ matrices

$$\mathbf{s}_t = \underbrace{\begin{bmatrix} 1 & \Delta t & \frac{1}{2}(\Delta t)^2 & 0 & 0 & 0 \\ 0 & 1 & \Delta t & 0 & 0 & 0 \\ 0 & 0 & e^{-\alpha \Delta t} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t & \frac{1}{2}(\Delta t)^2 \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & e^{-\alpha \Delta t} \end{bmatrix}}_{\mathbf{A}_t} \mathbf{s}_{t-1} + \epsilon_t \qquad \mathbf{x}_t = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}}_{\mathbf{B}_t} \mathbf{s}_t + \delta_t$$

Typical Inference Task for Gaussian SSM

- One of the key tasks: Given sequence $x_1, x_2, \ldots, x_T$, infer latent $s_1, s_2, \ldots, s_T$



- Usually two ways of inferring the latent states
  - Infer $p(s_t|x_1, x_2, \ldots, x_t)$: Called the "filtering" problem

$$p(\mathbf{s}_t|\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_t) \propto \underbrace{p(\mathbf{x}_t|\mathbf{s}_t)}_{\mathcal{N}(\mathbf{x}_t|\mathbf{B}\mathbf{s}_t, \mathbf{R})} \int \underbrace{p(\mathbf{s}_t|\mathbf{s}_{t-1})}_{\mathcal{N}(\mathbf{s}_t|\mathbf{A}\mathbf{s}_{t-1}, \mathbf{Q})} p(\mathbf{s}_{t-1}|\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{t-1}) d\mathbf{s}_{t-1}$$

Turns out to be another Gaussian

A Gaussian

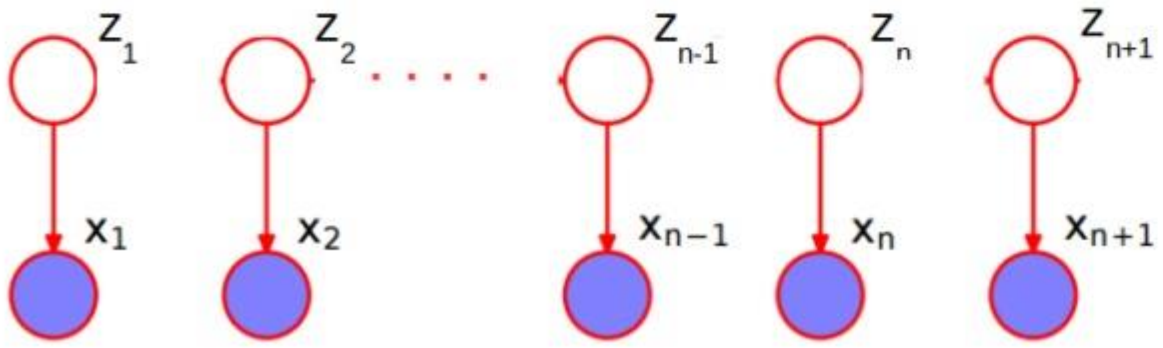
Kalman Filtering is a popular algorithm for a linear Gaussian SSM

  - Infer $p(s_t|x_1, x_2, \ldots, x_t, \ldots, x_T)$: Called the "smoothing" problem

- Some other tasks one can solve for using an SSM
  - Predicting future states $p(s_{t+h}|x_1, x_2, \ldots, x_t)$ for $h \geq 1$, given observations thus far
  - Predicting future observations $p(x_{t+h}|x_1, x_2, \ldots, x_t)$ for $h \geq 1$, given observations thus far

A Special Case

- What if we have i.i.d. latent states, i.e.,. $p(z_n|z_{n-1}) = p(z_n)$?



- Discrete case (HMM) becomes a simple mixture model $p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1} = \ell) = p(\boldsymbol{z}_n) = \text{multinoulli}(\boldsymbol{\pi})$
- Real-valued case (SSM) becomes a PPCA model $p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) = p(\boldsymbol{z}_n) = \mathcal{N}(\mathbf{0}, \mathbf{I}_K)$ or $\mathcal{N}(\boldsymbol{\mu}, \Psi)$
- Inference algos for HMM/SSM are thus very similar to that of mixture models/PPCA
  - Only main difference is how the latent variables z_n's are inferred since they aren't i.i.d.
  - E.g., if using EM, only E step needs to change (Bishop Chap 13 has EM for HMM and SSM)

Conformal Prediction

Conformal Prediction

- A simple technique to easily obtain confidence intervals
  - In classification, such an interval may refer to the set of highly likely classes for a test input

  - For more difficult test inputs, the set would typically be larger
- In a way, conformal prediction gives predictive uncertainty
  - However, unlike Bayesian ML, we don't get model uncertainty
  - Only one model is learned in the standard way and we construct the set of likely classes
  - It's like a black-box method; no change to training procedure for the model

Conformal Prediction

Assume it's a classification model which produces softmax scores

Conformal prediction can be used for regression problems too*

- Assume we already have a trained model $\hat{f}$ using some labelled data
- Suppose we get a test input X_{test} whose true (unknown) label is Y_{test}
- Use $\hat{f}$ and a calibration set of n examples to generate a prediction set $\mathcal{C}(X_{test})$ s.t.

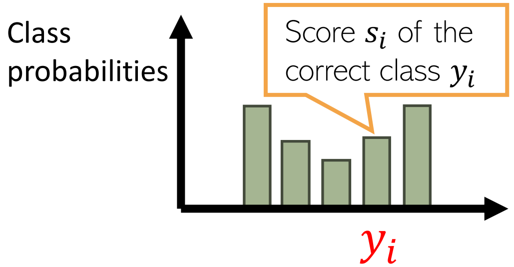
α is a user chosen error rate

$$1 - \alpha \leq p\big(Y_{test} \in \mathcal{C}(X_{test})\big) \leq 1 - \alpha + \frac{1}{n+1}$$

- To construct the set, we first compute, for each example in the calibration set

Probability/score of the correct class y_i of the input x_i

$$s_i = \hat{f}(x_i)_{y_i}$$



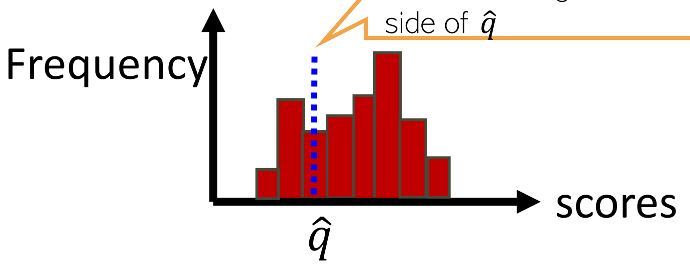
- Use the calibration set scores $s_1, s_2, \ldots, s_n$ to compute their α quantile

*A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification (Angelopoulos and Bates, 2022)

Conformal Prediction

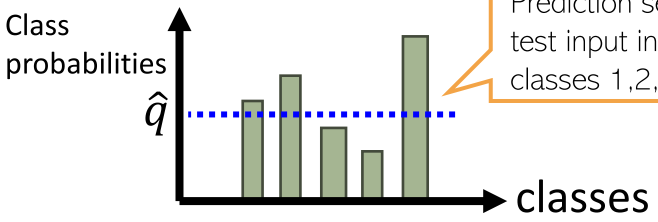
- Assume the α (say 0.1) quantile of the calibration set scores is equal to $\hat{q}$

Pick $\hat{q}$ such that α fraction of the mass of histogram is on the left side of $\hat{q}$



Probability of true class

- Assuming n is very large, roughly $(1-\alpha)$ fraction of inputs will have score higher than $\hat{q}$
- Given a test input X_{test}, whose label is is unknown, we compute the class probabilities



Prediction set for this test input includes classes 1,2,5

- Define the prediction set for X_{test} as

$$\mathcal{C}(X_{test}) = \{y: \hat{f}(X_{test})_y \geq \hat{q}\}$$

Report all the classes whose probability is large enough (the "large enough" value is given by the α quantile $\hat{q}$)

Conformal Prediction

- A generic black-box method
- Can be easily applied to any already trained classifier
- Predicted set has some nice guarantees

$$1 - \alpha \leq p\big(Y_{test} \in \mathcal{C}(X_{test})\big) \leq 1 - \alpha + \frac{1}{n+1}$$

- Does not make any assumptions on the distribution of the data
  - Thus considered a "distribution-free" approach to uncertainty quantification
- Can also be applied to regression problems*

*A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification (Angelopoulos and Bates, 2022)

Simulation-based Inference

Simulation-based Inference

- Suppose we wish to compute the posterior $p(\theta|D)$
- However, suppose we can't compute the likelihood $p(D|\theta)$
  - Evaluation too expensive, or don't have explicit likelihood
- Simulation-based Inference (SBI) approximates $p(\theta|D)$ as follows
  - For $i = 1,2, \dots S$
    - Draw a random $\theta^{(i)}$ the prior $p(\theta)$. Simulate a dataset $D^{(i)}$ from some simulator using $\theta^{(i)}$
    - Check how "similar" $D^{(i)}$ is to D. Define a suitable distance to measure this, e.g.,

$$d_i = \left\| s\left(D^{(i)}\right) - s(D) \right\|$$

    - Define the weight of $\theta^{(i)}$ as inversely proportional to d_i, e.g., $w_i \propto \exp(-d_i)$
  - The approximate posterior is $\{w_i, \theta^{(i)}\}_{i=1}^{S}$
- The vanilla SBI/ABC can be inefficient in practice (most $\theta^{(i)}$'s may have low weights)
  - More efficient versions proposed in recent research, e.g, neural conditional density estimators
  - Check out this package for code and links to other methods: https://github.com/sbi-dev/sbi

SBI is also known as "Approximate Bayesian Computation" (ABC)

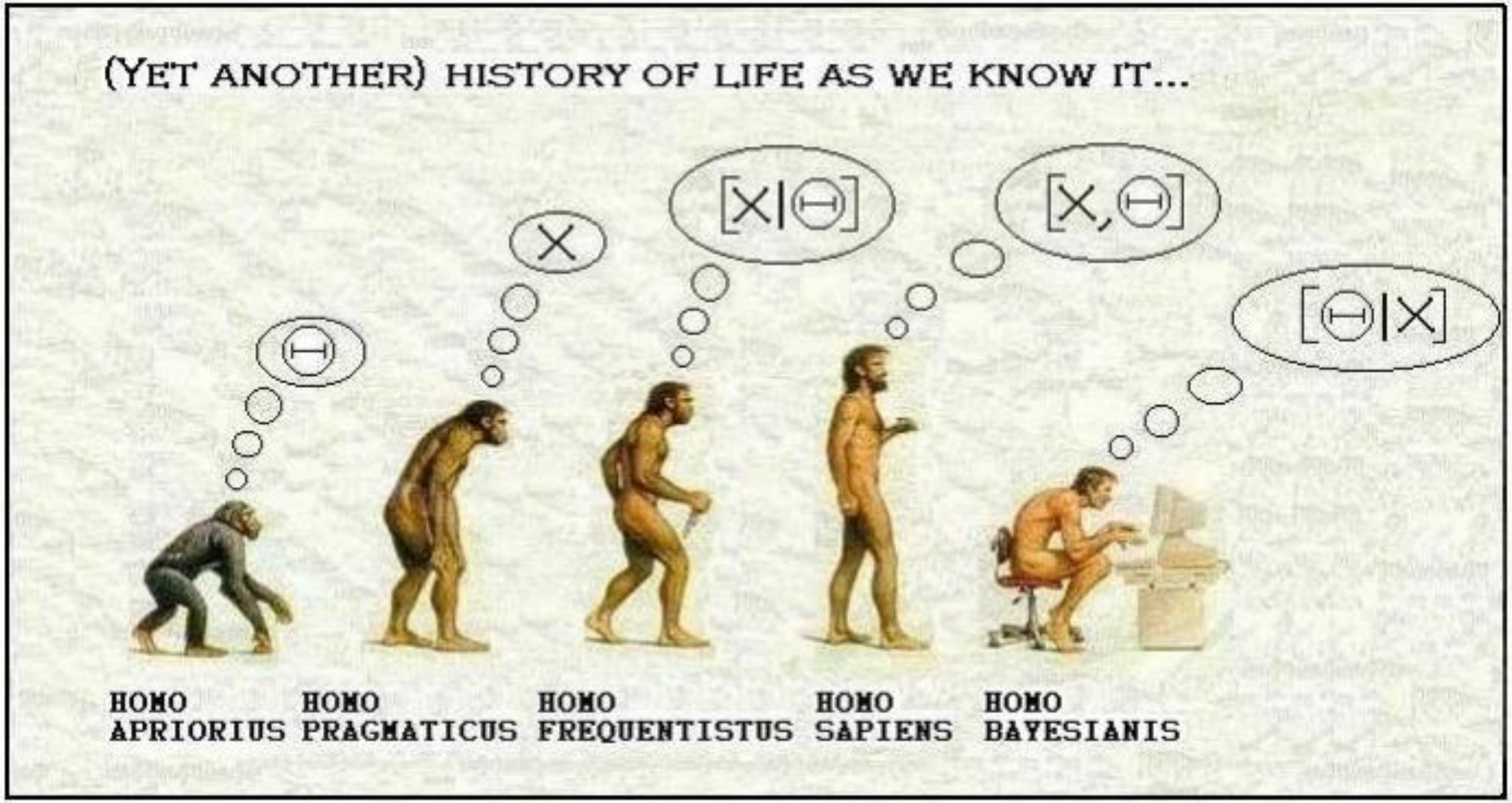
This simulator may be some domain-specific model of the data generation process (e.g., a physics engine, robotics/control simulator, etc)

Here $s(.)$ denotes a "summary statistics" which provides a summary of the dataset (e.g., its mean and covariance) which makes the comparison easier

Conclusion

- Probabilistic modeling provides a natural way to think about models of data
- Many benefits as compared to non-probabilistic approaches
  - Easier to model and leverage uncertainty in data/parameters
  - Principle of marginalization while making prediction
  - Easier to encode prior knowledge about the problem (via prior/likelihood distributions)
  - Easier to handle missing data (by marginalizing it out if possible, or by treating as latent variable)
  - Easier to build complex models can be neatly combining/extending simpler probabilistic models
  - Easier to learn the "right model" (hyperparameter estimation, nonparametric Bayesian models)
- Bayesian approaches as well as single model based uncertainty
- Uncertainty is important but proper calibration of uncertainty is also important
- Fast-moving field, lots of recent advances on new models and inference methods

Thank You!


(YET ANOTHER) HISTORY OF LIFE AS WE KNOW IT...
[Θ]
[X]
[X|Θ]
[X,Θ]
[Θ|X]
HOMO APRIORIUS
HOMO PRAGMATICUS
HOMO FREQUENTISTUS
HOMO SAPIENS
HOMO BAYESIANIS