Bits-Back Coding With Latent Variable Models

Last video:  
- overview of probability theory & random vars.  
- modeling error: KL-Divergence  
  → need to model correlations!  
  Problem 4.2 & 6.2  
  \[ H_p(x) + H_p(y) \geq H_p((x,y)) \]  
- autoregressive models

This video:  
- latent variable models  
- Bayesian inference  
- Bits-back coding

- can model long-ish range correlations  
- compact to store

- struggle with very long correlations  
- not parallelizable
Consider these hypothetical news headlines.

- Parliament Votes on New Labor Bill.
- Labor Union Votes to Extend Strikes.
- Soccer Player Scores First Goal Since Joining New Team.
- Guest Team is Leading by One Goal.

**Observation:** words within a headline appear to be correlated:

- Consider two positions \( i \neq j \); claim: \( X_i \) & \( X_j \) are not stat. indep.
- Words at these positions

\[
P(X_i = \text{"Goal"}, X_j = \text{"Team"}) > P(X_i = \text{"Goal"}) P(X_j = \text{"Team"})
\]

**Model of the "Generative Process"**

These pictures denote a joint prob. dist. that factorizes as follows:

\[
P(X, Z) = P(Z) P(X|Z) = P(Z) \prod_{i=1}^{k} P(X_i; | Z)
\]

("Topic Model", e.g. LDA: Blei & Ng 2003, blei quotes Pitchford et al. 2000)

\[\Rightarrow \text{marginal dist. of the message } X:
\]

\[
P(X) = \sum_Z P(X, Z) = \sum_Z \left( P(Z; z) \prod_{i=1}^{k} P(X_i; | z) \right)
\]

**Claim:** this model can capture correlations like

\[
P(X_i = \text{"Goal"}, X_j = \text{"Team"}) > P(X_i = \text{"Goal"}) P(X_j = \text{"Team"})
\]

**Proof:** exercise.
**Data Compression With Latent Variable Models**

\[ P(X, Z) = P(Z) P(X | Z) \]

Ideally we would like to compress \( X \) with this model.

**Problem:** we don’t know value of \( Z \)

**Problem Set 5:** implement & compare 3 compression methods for \( Z \): 1.7

- **Problem 5.2:** treat \( X_i \) as independent
  - \( H_p \{ X \} = \sum \frac{1}{i} H_p \{ X_i \} \geq H_p \{ X \} \)

- **Problem 5.3:** MAP estimate
  - \( H_p \{ X \} \choose \{ Z \} = \sum \log P(Z | \hat{X}) + H_p \{ X | Z = \hat{X} \} \)

- **Problem 5.4:** belief back coding
  - \( H_p \{ X \} \choose \{ Z \} = H_p \{ X \} \)

**Naive approach:** MAP estimate

\[ P(X, Z) = P(Z) P(X | Z) \]

- idea: encode some value \( \hat{Z} \) for \( Z \) using \( P(Z) \) & transmit
- then encode \( \hat{X} \) using \( P(X | Z = \hat{Z}) \)

\( = \prod_{i=1}^{k} P(X_i | Z = \hat{Z}) \)

\( \Rightarrow \) decoder can: decode \( \hat{Z} \) using \( P(Z) \)
- decode \( \hat{X} \) using \( P(X | Z = \hat{Z}) \)

**Bit rate:**

\[ R^{(1)}(x) = -\log P(Z = \hat{Z}) - \log P(X = \hat{X} | Z = \hat{Z}) \]

\[ = -\log P(X = \hat{X}, Z = \hat{Z}) \]

Chose \( \hat{Z} = \arg \min_{z} R^{(1)}(x) = \arg \max_{\hat{Z}} P(X = \hat{X}, Z = \hat{Z}) \)

\( \uparrow \)

\( \text{“minimum a-posteriori” (MAP) estimate of } Z \)

Overhead over theoret. bound:

\[ R^{(2)}(x) - (-\log P(X = \hat{X})) = -\log P(X = \hat{X}, Z = \hat{Z}) + \log P(X = \hat{X}) \]

\[ = -\log P( Z = \hat{Z} | X = \hat{X} ) \sim \text{posterior distribution} \]
Bayesian Inference

- model: $P(X, Z) = P(Z) P(X|Z)$

  => know $X$, don't know $Z$ (=>$\text{MAP estimate method has an overhead}$)

  => But: knowing $X$ typically reveals some information about $Z$

  Parliament Votes on New Labor Bill.
  Labor Union Votes to Extend Strikes.
  Soccer Player Scores First Goal Since Joining New Team.
  Guest Team is Leading by One Goal.

  However: there can still be some ambiguity about $Z$ (even after you know $X$)

  Parliament Votes on Aid for Community Sports Teams.

  => can only make prob. statements about $Z$

  $P(Z|X=x) = \frac{P(Z) P(X=x|Z)}{P(X=x)}$

  Remarks: in principle posterior distribution is known once you know $P(X, Z)$ & $X$

  in practice, however, calculating the posterior is often prohibitively expensive

  (=>$\text{Lecture 7: approximate Bayesian inference}$)

Understanding the overhead of MAP-est. method

- we could encode $(X)$ in two different ways

  1. $Z=\text{"politics"}$, then we use $P(X|Z=\text{"politics"})$
  2. $Z=\text{"sports"}$, then we use $P(X|Z=\text{"sports"})$

  remember: overhead = $-\log P(Z=Z^*|X=X)$
Bits-Back Coding


Idea: "piggyback" some additional message into the choice of a setup: communicate multiple messages (e.g., multiple image patches) over a single channel

- Usually:

- Bits-back: (operates as a stack, i.e. "last in first out")

Algorithm: "Bits-Back Coding"

- Subroutine encode (x, compressed, P):
  - begin
  - \( z \leftarrow \text{decode from compressed using } P(2 | X = z) \)
  - \( \text{encode } x \text{ using } P(X | 2 = z) \) into compressed
  - \( \text{encode } z \text{ using } P(2) \) onto compressed
  - return compressed

- Subroutine decode (compressed, P):
  - \( z \leftarrow \text{decode from compressed using } P(2) \)
  - \( x \leftarrow \text{decode from compressed using } P(X | 2 = z) \)
  - \( \text{encode } z \text{ onto compressed using } P(2 | X = z) \)
  - return \((x, \text{ compressed})\)
not bit rate of bits-back coding:

\[ R_{\text{net}}(x) = -\log \frac{P(x=x, z=z)}{P(z=z)} - \log P(z=z) - \log P(x=x) \]

\[ = -\log \frac{P(x=x, z=z) P(z=z)}{P(x=x)} = -\log P(x=x) \]

\[ \Rightarrow \text{bits-back coding is optimal (net).} \]

Next steps:

• How do we encode/decode fractional numbers of bits with stack semantics?

• What if we don’t know the exact posterior?
  \[ \Rightarrow \text{Lecture 7: approximate Bayesian inference} \]

• How can we efficiently train deep latent variable models?
  \[ \Rightarrow \text{Lecture 7 & subsequent: variational expectation maximizing deep generative models} \]