Approximate Inference in Practice
Microsoft’s Xbox TrueSkill™

Daniel Hernández-Lobato

Universidad Autónoma de Madrid

2014
Outline

1. Introduction
2. The Probabilistic Model
3. Approximate Inference
4. Results
Outline

1 Introduction

2 The Probabilistic Model

3 Approximate Inference

4 Results
Introduction

- **Competition is a key aspect of humans.**
  - Innate in most persons since a young age.
  - Used as a principle in most sports: soccer, basketball, etc.
- **Ratings used in games for fair competition.**
  - ELO system: Estimates the skill level of a chess player.
  - ATP system: Estimates the skill level of a tennis player.
  - Used for matchmaking in tournaments.
  - Generate a players ranking.
- **Online gaming poses additional challenges.**
  - Infer from a few match outcomes player skills.
  - Consider the possibility of teams with different number of players.
Questions that Arise in Online Gaming Skill Rating

- **Observed data:** Match outcomes of $k$ teams with $n_1, \ldots, n_k$ players each, in the form of a ranking with potential ties between teams.

- **Information we would like to obtain:**
  - Skills of each player $s_1, \ldots, s_k$.
    - If $s_i > s_j$ player $i$ is expected to beat player $j$.
  - Global ranking among players.
  - Fair matches between players and teams of players.

Successfully achieved by Microsoft’s Xbox TrueSkill™

Outline

1. Introduction
2. The Probabilistic Model
3. Approximate Inference
4. Results
TrueSkill™ Observed and Latent Variables

Latent Variables:
- **Skill** $s_i$ for player $i$: $p(s_i) = \mathcal{N}(s_i|\mu_i, \sigma_i^2)$.
- **Performance** $p_i$ of player $i$: $p(p_i|s_i) = \mathcal{N}(p_i|s_i, \beta^2)$.
- **Performance** $t_j$ of team $j$: $p(t_j|\{p_i: i \in A_j\}) = \delta(t_j - \sum_{i \in A_j} p_i)$.

Observed Variables:
- **Match outcome involving** $k$ teams: Rank $r_j$ for each team $j$.

$$p(r_1, \ldots, r_k|t_1, \ldots, t_k) = \mathbb{I}(t_{r_1} > t_{r_2} > \cdots > t_{r_k})$$

Thus, $r_j < r_{j+1}$ implies $t_j > t_{j+1}$.

The parameters of the model are $\beta^2$, $\mu_i$ and $\sigma_i^2$. 
We consider a game with 3 teams $A_1 = \{1\}$, $A_2 = \{2, 3\}$, $A_3 = \{4\}$.

**Results:** $r = (1, 2, 3)$. Team $A_1$ wins followed by $A_2$ and $A_3$.

Define $d_1 = t_1 - t_2$ and $d_2 = t_2 - t_3$:

$$p(d_1|t_1, t_2) = \delta(d_1 - t_1 + t_2), \quad p(d_2|t_2, t_3) = \delta(d_2 - t_2 + t_3).$$

The result implies $d_1 > 0$ and $d_2 > 0$. **Use transitivity!**

Thus, $r = (1, 2, 3)$ is equivalent to observing $b_1 = 1$ and $b_2 = 1$ where:

\[
p(b_1|d_1) = \begin{cases} 
1 & \text{if } d_1 > 0, \\
0 & \text{if } d_1 \leq 0.
\end{cases} \quad p(b_2|d_2) = \begin{cases} 
1 & \text{if } d_2 > 0, \\
0 & \text{if } d_2 \leq 0.
\end{cases}
\]
Note that we observe $b_1 = 1$ and $b_2 = 1$. 
Inference involves computing a posterior given a match outcome:

\[ p(s, p, t, d, | b) = \frac{p(s, b, t, p, d)}{p(b)} = \frac{p(b|d)p(d|t)p(t|p)p(p|s)p(s)}{p(b)} \]

We marginalize out \( t \) and \( p \) and \( d \) to get the marginal posterior of \( s \).

Bayesian Online Learning

The posterior after one match is used as the prior for the next match.

\[ \{\mu_i(0), \sigma_i^2(0)\} \xrightarrow{\text{Match #1}} \{\mu_i(1), \sigma_i^2(1)\} \xrightarrow{\text{Match #2}} \{\mu_i(2), \sigma_i^2(2)\} \]

Marginal Posterior is not Gaussian!
Outline

1 Introduction

2 The Probabilistic Model

3 Approximate Inference

4 Results
Message Passing Algorithm

We pass messages in the Bethe cluster graph until convergence to compute the posterior marginals of $s_1, \ldots, s_4$.

$$
\delta_{i \rightarrow j}(s_{i,j}) = \left\{ \int \psi_i(c_i) \left[ \prod_{k \in \text{Nb}_i} \delta_{k \rightarrow i}(s_{k,i}) \right] d(c_i \setminus s_{i,j}) \right\} / \delta_{j \rightarrow i}(s_{i,j}).
$$

where $s_{i,j}$ are the variables in the sepset of edge $i \rightarrow j$. This is the same rule as the one used in Belief Propagation.

The posterior marginals for $s_1, \ldots, s_4$ are given:

$$
p(s_i|b) \propto \psi_i(s_i) \prod_{k \in \text{Nb}_i} \delta_{k \rightarrow i}(s_i) = \mathcal{N}(s_i|\mu_i, \sigma_i^2) \delta_{k \rightarrow i}(s_i).
$$

All messages are Gaussian except for the two bottom messages!
Bethe Cluster Graph: Message Passing

\[ N(s_1|\mu_1, \sigma_1^2) \]

\[ s_1 \]

\[ N(p_1|s_1, \beta^2) \]

\[ p_1 \]

\[ \delta(t_1 - p_1) \]

\[ t_1 \]

\[ \delta(d_1 - t_1 + t_2) \]

\[ d_1 \]

\[ \mathbb{I}(d_1 > 0) \]

\[ N(s_2|\mu_2, \sigma_2^2) \]

\[ s_2 \]

\[ N(p_2|s_2, \beta^2) \]

\[ p_2 \]

\[ \delta(t_2 - p_2 - p_3) \]

\[ t_2 \]

\[ \delta(d_2 - t_2 + t_3) \]

\[ d_2 \]

\[ \mathbb{I}(d_2 > 0) \]

\[ N(s_3|\mu_3, \sigma_3^2) \]

\[ s_3 \]

\[ N(p_3|s_3, \beta^2) \]

\[ p_3 \]

\[ \delta(t_3 - p_4) \]

\[ t_3 \]

\[ N(s_4|\mu_4, \sigma_4^2) \]

\[ s_4 \]

\[ N(p_4|s_4, \beta^2) \]

\[ p_4 \]

\[ \delta(t_4) \]
Bethe Cluster Graph: Message Passing
Approximate Messages: Projection Step

We consider the case of the first message. The first step is to approximate the marginal by projecting into the Gaussian family:

$$\psi_i(d_1)\delta_{j \rightarrow i}(d_1) = \mathbb{I}(d_1 > 0) \mathcal{N}(d_1 | \hat{m}, \hat{\nu}).$$

For this, we compute the log of the normalization constant:

$$\log Z = \log \int \mathbb{I}(d_1 > 0) \mathcal{N}(d_1 | \hat{m}, \hat{\nu}) dd_1 = \log \Phi \left( \frac{\hat{m}}{\sqrt{\hat{\nu}}} \right).$$

We can obtain the mean and the variance of $\mathbb{I}(d_1 > 0) \mathcal{N}(d_1 | \hat{m}, \hat{\nu})$ by computing the derivatives with respect to $\hat{m}$ and $\hat{\nu}$!

Assume $m$ and $\nu$ are the mean and the variance. The approximate message is then:

$$\delta_{i \rightarrow j}(d_1) \propto \frac{\mathcal{N}(d_1 | m, \nu)}{\mathcal{N}(d_1 | \hat{m}, \hat{\nu})}.$$

The computation of the other approximate message is equivalent.
Convergence Speed

![Convergence Speed Graph](image)

- **Level** vs **Number of Games**
- **char (TrueSkill™)**
- **SQLWildman (TrueSkill™)**
- **char (Halo 2 rank)**
- **SQLWildman (Halo 2 rank)**

Legend:
- Red solid line: char (TrueSkill™)
- Blue solid line: SQLWildman (TrueSkill™)
- Red dashed line: char (Halo 2 rank)
- Blue dashed line: SQLWildman (Halo 2 rank)
Applications to Online Gaming

- **Leader-board of players:**
  - Provides a global ranking of all players.
  - The rank is conservative: $\mu_i - 3\sigma_i$.

- **Matchmaking:**
  - For players: Most uncertain outcome is better.
  - For inference: Most uncertain outcome is most informative.
  - Good for both players and the model.
Matchmaking: Probability of winning or loosing

We assume player $i$ wants to play against player $j$. What is the probability of winning?

$$p(p_i > p_j) = \int \mathbb{I}(p_i - p_j > 0) p(p_j | s_j) p(s_j) p(p_i | s_i) p(s_i) dp_i dp_j ds_i ds_j$$

$$= \int \mathbb{I}(p_i - p_j > 0) \mathcal{N}(p_j | s_j, \beta^2) \mathcal{N}(p_i | s_i, \beta^2) \cdot \mathcal{N}(s_j | \mu_j, \sigma_j^2) \mathcal{N}(s_i | \mu_i, \sigma_i^2) dp_i dp_j ds_i ds_j$$

$$= \int \mathbb{I}(p_i - p_j > 0) \mathcal{N}(p_j | \mu_j, \sigma_j^2 + \beta^2) \mathcal{N}(p_i | \mu_i, \sigma_i^2 + \beta^2) dp_i dp_j$$

$$= \Phi \left( \frac{\mu_i - \mu_j}{\sqrt{\sigma_i^2 + \sigma_j^2 + 2\beta^2}} \right).$$
Game Over!