
Heteroscedastic TrueSkill: Modeling Match Noise and Player Consistency

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 TrueSkill is a widely used Bayesian rating system for inferring latent player skill
2 from game outcomes. However, its standard formulation assumes a fixed global
3 performance-noise parameter, making it sensitive to atypical outcomes and unable
4 to distinguish skill from player-specific consistency. We propose two heteroscedas-
5 tic extensions: a match-specific precision model that yields a robust heavy-tailed
6 comparison likelihood, and a player-specific precision model that captures differ-
7 ences in performance consistency among players. Both extensions use Gamma-
8 distributed precision variables and retain a modular factor-graph representation.
9 Because the introduced Gaussian-Gamma factors do not admit the same closed-
10 form expectation-propagation updates as vanilla TrueSkill, we derive an efficient
11 hybrid message-passing scheme that combines expectation-propagation updates
12 for outcome truncation factors with variational updates for latent precision factors.
13 Experiments on synthetic and real match data show that the proposed models
14 improve robustness to anomalous outcomes and provide interpretable estimates of
15 player consistency.

16 1 Introduction

17 Skill-rating systems are widely used to summarize competitive performance, rank players, and predict
18 future outcomes. Among these, TrueSkill [Herbrich et al., 2007] has become a standard Bayesian
19 framework because it combines interpretable latent skill variables with efficient approximate inference
20 in a factor graph. In its basic form, however, TrueSkill assumes that all players and all matches share
21 the same performance variance.

22 First, real competitive data frequently contain atypical or noisy matches. Such outcomes may arise
23 from injury, fatigue, unfavorable conditions, or simply unusually variable performance. Under a
24 fixed global noise parameter, these matches can exert disproportionate influence on the inferred skills.
25 Second, players are not equally consistent: some perform close to their underlying skill from match
26 to match, whereas others exhibit substantial volatility. A single global performance variance cannot
27 represent this heterogeneity.

28 In this paper, we propose two extensions of TrueSkill that address these limitations. The first
29 introduces a *match-specific* latent precision variable, allowing the model to treat some matches as
30 intrinsically noisier than others. This yields a robust ranking model in which anomalous outcomes
31 are partially absorbed by match-level uncertainty rather than forcing large skill updates. The second
32 introduces a *player-specific* latent precision variable, allowing each player to have an individual
33 degree of performance consistency. This extension captures persistent cross-player differences in
34 volatility while preserving the probabilistic semantics of the original model.

35 Both extensions use latent relative-precision multipliers centered at one. In the robust model, a
36 match-specific multiplier yields a Student- t comparison-noise model after marginalization, providing

a heavy-tailed alternative to the Gaussian performance-difference noise of vanilla TrueSkill. In the consistency model, a player-specific multiplier rescales the player’s performance variance relative to the global baseline, allowing persistent differences in performance variability. Exact posterior inference is intractable, so we develop a hybrid inference procedure that combines expectation propagation (EP)[Minka, 2013], as used in vanilla TrueSkill, with variational inference (VI) message updates for the introduced Gaussian–Gamma scale-mixture factors (for VI, see [Zhang et al., 2018]).

Besides being employed for ranking and automatic matchmaking at scale in numerous online games such as the *Halo*, *Forza Motorsport* or *Gears of War* series, TrueSkill is used in a variety of use cases such as ranking machine translation systems from pairwise human judgments, converting human preferences of street view images into safety scores, predicting clicks of online advertisement, and assessing LLM-agent abilities [Naik et al., 2014, Sakaguchi et al., 2014, Graepel et al., 2010, Chen et al., 2024]. As these settings are based on human judgement or competitions involving luck (e.g. Poker), we stipulate they would benefit from match robustness or player consistency.

Our main contributions are as follows:

- We introduce a match-level precision factor for TrueSkill that yields a Student-t comparison likelihood and downweights anomalous outcomes through posterior match reliability.
- We introduce a player-level precision factor for TrueSkill that estimates persistent player-specific performance consistency independently of the global skill scale.
- We derive a modular hybrid EP–VI message-passing scheme for Gaussian–Gamma precision factors, enabling these heteroscedastic factors to be inserted into standard TrueSkill-style factor graphs. Evaluations for synthetic and large-scale real-world chess data (with 10K players and millions of games and variables) demonstrate both effectiveness and applicability.

For clarity, we present the static two-player setting throughout most of the paper. The same modeling ideas extend naturally to team games, multiplayer settings, and temporal TrueSkill variants.

2 Related Work

Classical paired-comparison models. The problem of estimating latent skill from competitive outcomes has a long history in statistics, psychometrics, and rating systems. Early formulations include Thurstone’s law of comparative judgment and the Bradley–Terry model, which infer latent strengths from paired comparisons and convert strength differences into win probabilities [David, 1988]. Building on the Bradley–Terry model, Elo-type systems popularized simple sequential updates for chess and related games [Elo, 1978] and Glicko 1 & 2 introduced Bayesian modeling of (historical) player skill [Glickman, 1999].

Bayesian rating systems. TrueSkill casts rating as inference in a Gaussian latent-variable model, extending beyond two-player matches to teams, multiplayer outcomes, and draws [Herbrich et al., 2007]. *TrueSkill Through Time* extends TrueSkill from online filtering to smoothing over a time series, estimating historical skill trajectories of chess players [Dangauthier et al., 2007]. Minka et al. [2018] extend TrueSkill with online gaming telemetry such as experience, squad membership, in-game kills, quitting behavior, and cross-mode skill. Guo et al. [2012] extend TrueSkill to model score-based match outcomes instead of plain win/draw/loss outcomes for sport modeling. In the vanilla TrueSkill formulation, each players match performance is modeled as a noisy Gaussian draw around latent skill with a common performance variance β^2 , which provides a natural baseline against which heteroscedastic extensions can be compared.

Heteroscedastic paired comparisons. Relaxations of homoscedastic noise assumptions also have substantial precedent. Within the Thurstonian tradition, unequal-variance formulations such as Case III and later covariance-structure/SEM approaches allow pair-specific or item-specific error variances instead of a single common comparison noise level, and explicitly study equal- versus unequal-variance specifications [David, 1988]. In dynamic paired-comparison models for sports and other evolving competitions, Glickman introduced stochastic state variances so that latent strengths could vary with time in a volatility-aware manner [Glickman, 2001], and Glicko-2 further assigned each player a volatility parameter capturing the expected fluctuation of that players rating over time [Glickman, 2022].

88 3 Our Approach

89 We begin by briefly reviewing vanilla TrueSkill in the static two-player setting in order to introduce
 90 notation (see also Appendix A). We then introduce two heteroscedastic extensions. The first assigns a
 91 latent relative-precision multiplier to each match and is designed to improve robustness to atypical
 92 outcomes. The second assigns a latent relative-precision multiplier to each player and is designed to
 93 capture persistent differences in player consistency. Throughout, we use a shape–rate parameterization
 94 of the Gamma distribution. The relative-precision multipliers have Gamma priors with mean one, so
 95 the global noise scale remains controlled by the baseline TrueSkill variance parameters.

96 3.1 Baseline: Vanilla TrueSkill

97 In vanilla TrueSkill, each player i is associated with a latent skill variable

$$S_i \sim \mathcal{N}(\mu_i, \sigma_i^2),$$

98 where μ_i is the current estimate of the player’s skill and σ_i^2 quantifies uncertainty in that estimate. In
 99 a given match, the player’s realized performance is modeled as

$$P_i \sim \mathcal{N}(S_i, \beta^2),$$

100 where the global hyperparameter β^2 represents match-to-match performance variability and is shared
 101 across all players and matches. For a match between players i and j , the performance difference is

$$D = P_i - P_j.$$

102 The observed outcome is determined by thresholding this difference. In the win–loss case without
 103 draws, player i wins if $D > 0$. More generally, with draw margin $\epsilon \geq 0$, the outcome is determined
 104 by whether $D > \epsilon$ (player i wins), $|D| \leq \epsilon$ (draw), or $D < -\epsilon$ (player j wins).

105 After observing the match outcome, the posterior over latent skills is updated by approximate
 106 inference in the factor graph shown in the Appendix B (Figure 5 and Table 3). Intuitively, winning
 107 increases μ_i and losing decreases μ_i , while repeated evidence reduces σ_i . A convenient representation
 108 uses the Gaussian natural parameters $\rho = \sigma^{-2}$ and $\tau = \mu \cdot \sigma^{-2}$, because multiplication of Gaussian
 109 messages then reduces to simple addition of precisions ρ and precision-weighted means τ .

110 The key limitation of this formulation is the assumption that the same performance variance β^2
 111 applies to all players and all matches. The extensions proposed below relax this assumption by
 112 replacing the fixed noise scale β^2 with latent precision variables.

113 3.2 Model 1: Per-Match Precision Multiplier (Robustness)

114 To make TrueSkill more robust to atypical outcomes, we replace the fixed Gaussian performance-
 115 difference noise with a Student- t scale mixture. In vanilla TrueSkill, the performance difference D is
 116 Gaussian. We instead introduce, for each match m , a dimensionless relative-precision multiplier

$$B_m \sim \text{Gamma}(\nu/2, \nu/2),$$

117 so that $\mathbb{E}[B_m] = 1$ and $\text{Var}(B_m) = 2/\nu$. Let players i_m and j_m participate in match m . With
 118 $\tilde{D}_m := S_{i_m} - S_{j_m}$, we model the latent performance difference as

$$D_m \sim \mathcal{N}\left(\tilde{D}_m, \frac{\kappa^2}{B_m}\right),$$

119 where κ^2 is the baseline performance-difference variance. One may set $\kappa^2 = 2\beta^2$ to retain the vanilla
 120 TrueSkill difference-noise scale, or use $\kappa^2 = 2\beta^2(\nu - 2)/\nu$ (with $\nu > 2$) to match $\text{Var}(D_m)$ to the
 121 vanilla variance $2\beta^2$. Marginalizing over B_m yields

$$D_m \sim t_\nu(S_{i_m} - S_{j_m}, \kappa).$$

122 Thus, the model replaces the Gaussian comparison noise of vanilla TrueSkill with a heavy-tailed
 123 Student- t comparison-noise model. The auxiliary variable B_m is useful for inference: large residual
 124 matches induce small posterior values of B_m , thereby reducing their influence on posterior skill
 125 estimates. Furthermore B_m can be read as a posterior reliability score for match m : values below
 126 one indicate that the match provides weaker-than-baseline evidence about skill, while values above
 127 one indicate cleaner-than-baseline evidence.

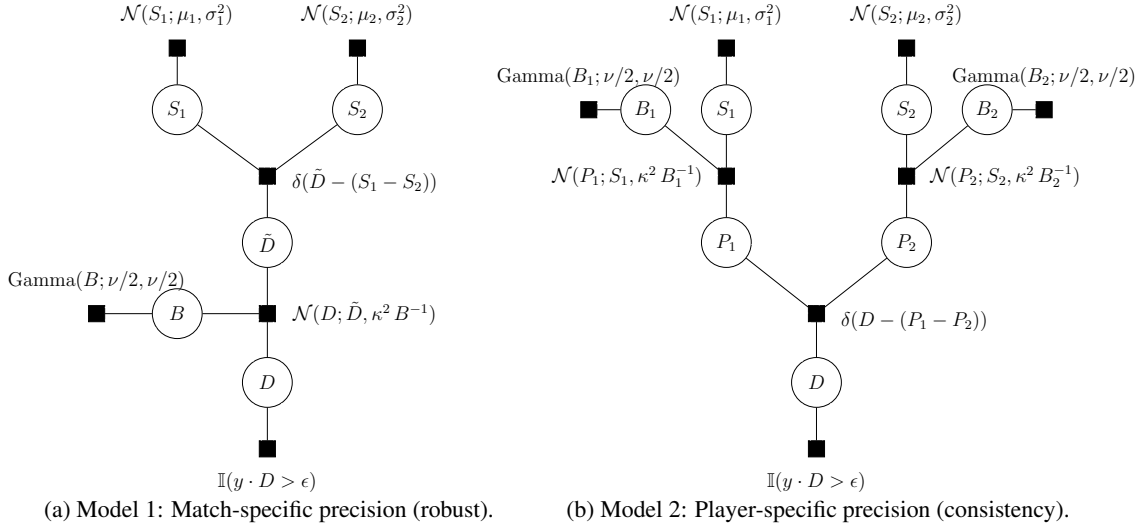


Figure 1: The factor graphs of the proposed TrueSkill extension models. Round nodes represent variables: S_i for player *skills*, P_i for realized player *performances*, and D for the latent performance *difference*. Square nodes represent *factors* (functions of the connected variables). The full graph factorizes into the product of the factors, where $\mathcal{N}(S_1; \mu_1, \sigma_1^2) \cdot \mathcal{N}(S_2; \mu_2, \sigma_2^2)$ encodes the prior over the skills, $\text{Gamma}(B_1; \nu/2, \nu/2) \cdot \text{Gamma}(B_2; \nu/2, \nu/2)$ encodes the prior over the relative precision multipliers, and the product of the remaining factors encode the likelihood of the observed game outcome y ($y = 1$ for a win of player 1 and $y = -1$ for a win of player 2). **(a)** In the robust model, each match is assigned a latent relative-precision multiplier B , which rescales the performance-difference variance as κ^2/B . Marginalizing B yields a Student- t comparison-noise model. **(b)** In the consistency model, each player i is assigned a latent relative-precision multiplier B_i , which rescales that player’s performance variance as κ^2/B_i .

128 3.3 Model 2: Per-Player Precision Multiplier (Consistency)

129 To model persistent differences in player-to-player consistency, we assign each player a dimensionless
 130 relative-precision multiplier. The key difference here compared to Model 1 is that we introduce a
 131 precision multiplier for each player, instead of for each match. For each player i , let

$$B_i \sim \text{Gamma}\left(\frac{\nu}{2}, \frac{\nu}{2}\right),$$

132 so that $\mathbb{E}[B_i] = 1$ and $\text{Var}(B_i) = \frac{2}{\nu}$. Player i ’s performance in match m is

$$P_i^{(m)} \sim \mathcal{N}\left(S_i, \frac{\kappa^2}{B_i}\right).$$

133 Setting κ^2 to β^2 retains its interpretation as the global baseline performance variance, while B_i
 134 captures player-specific relative consistency. Large values of B_i correspond to players whose
 135 performances concentrate more tightly around their latent skill, whereas small values correspond to
 136 more volatile players. For match m between players i_m and j_m , the latent performance difference is

$$D_m = P_{i_m}^{(m)} - P_{j_m}^{(m)}.$$

137 Unlike the robust match-specific model, where each match has its own local reliability multiplier, the
 138 consistency model shares B_i across all matches involving player i . Therefore, B_i captures persistent
 139 residual performance variability under the model rather than independent match-level outliers. The
 140 parameter ν controls the prior amount of player-to-player heterogeneity in consistency: large ν
 141 shrinks all players toward the common vanilla TrueSkill variance, while small ν permits larger
 142 differences in performance variability.

143 3.4 Approximate Inference for Models 1 and 2

144 Exact posterior inference is intractable in both proposed extensions. As in vanilla TrueSkill, the
 145 truncation factor induced by the observed outcome already breaks conjugacy. In addition, the new

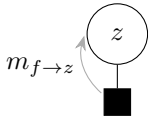
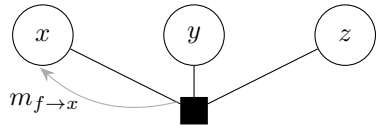
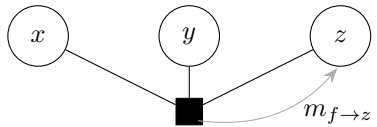
Factor	Update equation
 Gamma($z; \alpha, \lambda$)	$\alpha_z \leftarrow \alpha$ $\lambda_z \leftarrow \lambda$
 $\mathcal{N}(x; y, z^{-1})$	$\rho_{f \to x}^{\text{new}} \leftarrow \frac{\alpha_z}{\lambda_z}, \quad \tau_{f \to x}^{\text{new}} \leftarrow \frac{\alpha_z}{\lambda_z} \cdot \frac{\tau_y}{\rho_y}$ $m_{f \to y}$ follows from $\mathcal{N}(y; x, z^{-1}) = \mathcal{N}(x; y, z^{-1})$.
 $\mathcal{N}(x; y, z^{-1})$	$\xi_y := \frac{1}{\rho_y} + \left(\frac{\tau_y}{\rho_y} \right)^2, \quad \xi_x := \frac{1}{\rho_x} + \left(\frac{\tau_x}{\rho_x} \right)^2$ $\alpha_{f \to z}^{\text{new}} \leftarrow \frac{3}{2}, \quad \lambda_{f \to z}^{\text{new}} \leftarrow \frac{\xi_y + \xi_x}{2} - \frac{\tau_y \cdot \tau_x}{\rho_y \cdot \rho_x}$

Table 1: Novel factor types and update equations for the proposed consistent and robust models.

variable-precision factors couple Gaussian latent variables with Gamma latent variables through terms of the form $\mathcal{N}(X; Y, Z^{-1})$, which do not admit closed-form expectation propagation (EP) projections (for a corresponding proof, see Appendix D).

To address this, we propose a hybrid approximation that combines EP for the Gaussian comparison subgraph with variational message updates for the non-conjugate Gaussian–Gamma interactions, in which the outgoing Gaussian and Gamma messages are chosen by matching moments under the current approximate marginals. Crucially, all updates are implemented as factor-to-variable message replacements rather than as repeated multiplication into the current belief. For each variable v , the approximate marginal q is represented as

$$q(v) \propto \prod_{f \in \text{neighbors}(v)} m_{f \to v}(v).$$

When updating a message $m_{f \to v}$, we first form the cavity

$$q_{\setminus f}(v) \propto \frac{q(v)}{m_{f \to v}(v)},$$

compute a new outgoing message $\hat{m}_{f \to v}$, replace $m_{f \to v} \leftarrow \hat{m}_{f \to v}$, and then set

$$q(v) \propto q_{\setminus f}(v) \hat{m}_{f \to v}(v).$$

A generic Gaussian–Gamma factor. Both extensions involve local factors of the form

$$f(X, Y, Z) = \mathcal{N}(X; Y, Z^{-1}) \propto Z^{1/2} \exp\left(-\frac{Z}{2}(Y - X)^2\right),$$

that couples Gaussian variables X and Y with a Gamma-distributed variable Z . We derive message updates for this factor using variational inference: Under the current approximate marginals, we form outgoing messages by replacing exact local coupling with expectations of neighboring variables.

Lemma 1 (Variational messages for a Gaussian–Gamma factor). *Let $f(X, Y, Z) = \mathcal{N}(X; Y, Z^{-1})$ where X and Y are Gaussian variables and Z is a Gamma variable. Then the variational outgoing messages from f are*

$$m_{f \to X}(x) \propto \mathcal{G}(x; \mathbb{E}[Y]\mathbb{E}[Z], \mathbb{E}[Z]),$$

164

$$m_{f \rightarrow Y}(y) \propto \mathcal{G}(y; \mathbb{E}[X]\mathbb{E}[Z], \mathbb{E}[Z]),$$

165 and

$$m_{f \rightarrow Z}(z) \propto \text{Gamma}\left(z; \frac{3}{2}, \frac{\mathbb{E}[X^2] + \mathbb{E}[Y^2]}{2} - \mathbb{E}[X]\mathbb{E}[Y]\right).$$

166 *Proof.* With respect to X , it holds that

$$f(X, Y, Z) \propto \exp\left(\underbrace{\begin{bmatrix} ZY \\ Z \end{bmatrix}}_{\eta_X(Y, Z)^\top}^\top \underbrace{\begin{bmatrix} X \\ -\frac{1}{2}X^2 \end{bmatrix}}_{T_X(X)}\right),$$

167 and therefore for the Gaussian message from the factor f to the variable X

$$m_{f \rightarrow X}(x) \propto \exp(\mathbb{E}[\eta_X(Y, Z)]^\top T_X(x)) \propto \mathcal{G}(x; \mathbb{E}[Y] \cdot \mathbb{E}[Z], \mathbb{E}[Z]),$$

168 where the expectations are taken with respect to the current marginals of X, Y and Z . Here,
 169 $\mathcal{G}(x; \tau, \rho) = \mathcal{N}(x; \frac{\tau}{\rho}, \rho^{-1})$ denotes the pdf of a Normal in natural parameters. Similarly, the outgoing
 170 message to X is Gaussian with

$$m_{f \rightarrow Y}(y) \propto \mathcal{G}(y; \mathbb{E}[X] \cdot \mathbb{E}[Z], \mathbb{E}[Z]).$$

171 For the precision variable, we collect the terms in Z :

$$f(X, Y, Z) \propto \exp\left(\underbrace{\begin{bmatrix} \frac{1}{2} \\ \frac{1}{2}(X - Y)^2 \end{bmatrix}}_{\eta_Z(X, Y)^\top}^\top \underbrace{\begin{bmatrix} \log(Z) \\ -Z \end{bmatrix}}_{T_Z(Z)}\right).$$

172 Hence, the outgoing message to Z is Gamma with

$$m_{f \rightarrow Z}(z) \propto \exp(\mathbb{E}[\eta_Z(X, Y)]^\top T_Z(z)) \propto \text{Gamma}\left(z; \frac{3}{2}, \frac{\mathbb{E}[X^2] + \mathbb{E}[Y^2]}{2} - \mathbb{E}[X] \cdot \mathbb{E}[Y]\right).$$

173 An extended proof is provided in Appendix C. $\square$

174 The derived parameter updates are summarized in Table 1.

175 **Per-match precision (robust).** In the robust model, the joint posterior is

$$p(\mathbf{s}, \mathbf{b}, \mathbf{d} \mid \mathbf{y}) \propto \left(\prod_i p(s_i)\right) \left(\prod_m p(b_m)\right) \left(\prod_m \mathcal{N}\left(d_m; s_{i_m} - s_{j_m}, \frac{\kappa^2}{b_m}\right)\right) \cdot \prod_m p(y_m \mid d_m).$$

176 The outcome likelihood $p(y_m \mid d_m)$ is the threshold likelihood induced by the win/loss or
 177 win/draw/loss observation. We approximate the true posterior p by

$$q(\mathbf{s}, \mathbf{b}, \mathbf{d}) = \left(\prod_i q(s_i)\right) \left(\prod_m q(b_m)\right) \left(\prod_m q(d_m)\right),$$

178 where each $q(s_i)$ and $q(d_m)$ is Gaussian and each $q(b_m)$ is Gamma. The Gaussian skill priors,
 179 difference constraints, and outcome truncation factors are updated using the same EP-style Gaussian
 180 message passing as in vanilla TrueSkill. The scale-mixture factor

$$\mathcal{N}\left(d_m; s_{i_m} - s_{j_m}, \frac{\kappa^2}{b_m}\right)$$

181 is updated using the variational Gaussian–Gamma updates from Lemma 1, with $X \leftarrow d_m$, $Y \leftarrow$
 182 $s_{i_m} - s_{j_m}$ and $Z \leftarrow b_m/\kappa^2$. In particular, the resulting local posterior update is

$$q(b_m) = \text{Gamma}\left(\frac{\nu + 1}{2}, \frac{\nu + \mathbb{E}[(D_m - S_{i_m} + S_{j_m})^2]/\kappa^2}{2}\right).$$

183 The posterior mean,

$$\mathbb{E}_q[B_m] = \frac{\nu + 1}{\nu + \mathbb{E}[(X - Y)^2]/\kappa^2},$$

184 shows the robust reweighting effect: large standardized residuals lead to smaller expected relative
 185 precision and therefore weaker Gaussian messages.

Robust: match precision

Require: Matches $\mathcal{M} = \{(w_m, \ell_m, y_m)\}$ (winner, loser, outcome), priors $q_0(s_i)$, $q_0(b_i)$, convergence criteria τ_{global} , τ_m

- 1: Initialise $q(s_i) = q_0(s_i)$ and $q(b_i) = q_0(b_i)$
- 2: **repeat**
- 3: $\Delta_{\text{global}} \leftarrow 0$
- 4: **for all matches** m **do**
- 5: **repeat**
- 6: $\Delta_m \leftarrow 0$
- 7: EP: Update $m_{\delta \rightarrow \bar{D}_m}$ using vanilla updates (Table 3)
- 8: VI: Update $m_{\mathcal{N} \rightarrow D_m}$ as in Table 1
- 9: EP: Update $m_{\bar{I} \rightarrow D_m}$ using vanilla updates (Table 3)
- 10: VI: Update $m_{\mathcal{N} \rightarrow \bar{D}_m}$ as in Table 1
- 11: VI: Update $m_{\mathcal{N} \rightarrow B_m}$ as in Table 1
- 12: EP: Update $m_{\delta \rightarrow S_{w_m}}$ and $m_{\delta \rightarrow S_{\ell_m}}$ (Table 3)
- 13: Set Δ_m to the largest local belief change.
- 14: **until** $\Delta_m < \tau_m$
- 15: $\Delta_{\text{global}} \leftarrow \max(\Delta_{\text{global}}, \Delta_m)$
- 16: **end for**
- 17: **until** $\Delta_{\text{global}} < \tau_{\text{global}}$
- 18: **return** $\{q(s_i)\}, \{q(b_i)\}$

Consistent: player precision

Require: Matches $\mathcal{M} = \{(w_m, \ell_m, y_m)\}$ (winner, loser, outcome), priors $q_0(s_i)$, $q_0(b_i)$, convergence criteria τ_{global} , τ_m

- 1: Initialise $q(s_i) = q_0(s_i)$ and $q(b_i) = q_0(b_i)$
- 2: **repeat**
- 3: $\Delta_{\text{global}} \leftarrow 0$
- 4: **for all matches** m **do**
- 5: **repeat**
- 6: $\Delta_m \leftarrow 0$
- 7: VI: Update $m_{\mathcal{N} \rightarrow P_{w_m}}$ and $m_{\mathcal{N} \rightarrow P_{\ell_m}}$ (Table 1)
- 8: EP: Update $m_{\delta \rightarrow D}$ and $m_{\bar{I} \rightarrow D}$ (Table 3)
- 9: EP: Update $m_{\delta \rightarrow P_{w_m}}$ and $m_{\delta \rightarrow P_{\ell_m}}$ (Table 3)
- 10: VI: Update $m_{\mathcal{N} \rightarrow B_{w_m}}$ and $m_{\mathcal{N} \rightarrow B_{\ell_m}}$ (Table 1)
- 11: VI: Update $m_{\mathcal{N} \rightarrow S_{w_m}}$ and $m_{\mathcal{N} \rightarrow S_{\ell_m}}$ (Table 1)
- 12: Set Δ_m to the largest local belief change.
- 13: **until** $\Delta_m < \tau_m$
- 14: $\Delta_{\text{global}} \leftarrow \max(\Delta_{\text{global}}, \Delta_m)$
- 15: **end for**
- 16: **until** $\Delta_{\text{global}} < \tau_{\text{global}}$
- 17: **return** $\{q(s_i)\}, \{q(b_i)\}$

Figure 2: Hybrid EP–VI message passing for robust and consistent TrueSkill.

186 **Per-player precision (consistency).** In the consistency model, the joint posterior is

$$p(\mathbf{s}, \mathbf{b}, \mathbf{p}, \mathbf{d} \mid \mathbf{y}) \propto \left(\prod_i p(s_i) p(b_i) \right) \left(\prod_m \mathcal{N} \left(p_{i_m}^{(m)}; s_{i_m}, \frac{\kappa^2}{b_{i_m}} \right) \mathcal{N} \left(p_{j_m}^{(m)}; s_{j_m}, \frac{\kappa^2}{b_{j_m}} \right) \right) \\ \cdot \prod_m \delta \left(d_m - p_{i_m}^{(m)} + p_{j_m}^{(m)} \right) \prod_m p(y_m \mid d_m).$$

187 We use the approximate factorization

$$q(\mathbf{s}, \mathbf{b}, \mathbf{p}, \mathbf{d}) = \left(\prod_i q(s_i) \right) \left(\prod_i q(b_i) \right) \left(\prod_m q(p_{i_m}^{(m)}) q(p_{j_m}^{(m)}) q(d_m) \right),$$

188 with Gaussian beliefs for skills, performances, and differences, and Gamma beliefs for player-specific
189 relative precisions. The deterministic difference factor and the outcome truncation factor are handled
190 as in vanilla TrueSkill. The scale-mixture factor

$$\mathcal{N} \left(p_i^{(m)}; s_i, \frac{\kappa^2}{b_i} \right)$$

191 is updated as described in Lemma 1 with $X \leftarrow p_i^{(m)}$, $Y \leftarrow s_i$, and $Z \leftarrow b_i/\kappa^2$. Because B_i is shared
192 across all matches involving player i , its posterior accumulates residual evidence over those matches:

$$q(b_i) = \text{Gamma} \left(\frac{\nu + n_i}{2}, \frac{\nu}{2} + \frac{\kappa^{-2}}{2} \sum_{m: i \in m} \mathbb{E} \left[\left(P_i^{(m)} - S_i \right)^2 \right] \right),$$

193 where n_i is the number of performance observations involving player i . Thus, players whose latent
194 performances remain close to their latent skill receive larger posterior relative precision, while players
195 with larger residual performance fluctuations receive smaller posterior relative precision.

196 **Computational remarks.** In both models, inference proceeds by iterative message passing until
197 convergence of the approximate marginals as detailed in Figure 2. The resulting algorithm is best
198 understood as a *hybrid EP–VI procedure*. It inherits from EP the cavity-based updates and moment
199 projections used in the Gaussian comparison graph, exactly as in vanilla TrueSkill. It inherits
200 from variational inference the factorized treatment of the latent precision variables and the use of
201 expectations of neighboring variables to produce tractable Gaussian and Gamma messages. This
202 combination is appealing because the outcome truncation factors remain well handled by EP, while
203 the additional latent precision variables can be updated in closed form.

204 Each local message update has constant cost, so the total computational cost is linear in the number
205 of matches per outer iteration. Relative to vanilla TrueSkill, the main additional cost arises from
206 maintaining Gamma beliefs over the relative-precision multipliers.

Identifiability and scale. As in vanilla TrueSkill, the proposed models are not identifiable in absolute location or scale unless one or more components are externally anchored. Because outcomes depend on skill differences, the likelihood is invariant under a global shift $s_i \mapsto s_i + a$ for any constant a . In the binary no-draw setting, the likelihood also depends only on the ratio between skill differences and the performance-noise scale. Thus, the transformation $s_i \mapsto c s_i$, $\beta \mapsto c \beta$, $\kappa \mapsto c \kappa$ leaves the likelihood unchanged for any $c > 0$. With draws, the same scale invariance holds only if the draw margin is scaled as $\epsilon \mapsto c \epsilon$; otherwise a fixed draw margin anchors the scale.

The canonical relative-precision multipliers B_m and B_i are dimensionless and centered at one, so they do not determine the global skill scale. Instead, the global scale is controlled by the baseline variance parameters β^2 and κ^2 , together with the skill prior and, when present, the draw margin. The parameter ν controls the amount of heterogeneity around this baseline, controlling the tail-heaviness of the robust model in Model 1 or the prior variability of player-specific consistency in Model 2.

In the player-specific consistency model, there remains a practical trade-off between skill uncertainty and inferred consistency: irregular outcomes may be explained either by uncertain skill estimates or by low player relative precision. We address this by fixing the global scale through the skill prior and baseline performance variance, and by using Gamma priors centered at $\mathbb{E}[B_i] = 1$, which shrink player-specific consistency toward vanilla TrueSkill in the absence of sufficient evidence.

4 Evaluation

Corruption robustness (synthetic, to validate against true values). We evaluate the robustness of the proposed models on a synthetic ranking task with $k = 40$ players and $n = 1\,000$ randomly drawn pairwise comparisons. The true skill order is fixed as $1 < 2 < \dots < k$, and uncorrupted comparisons are generated consistently with this ordering. We consider two corruption mechanisms. In the *match-noise* setting, a fraction δ of comparisons is selected uniformly at random and the outcome is flipped. In the *player-noise* setting, a subset of players is designated as noisy (inconsistent), and comparisons involving these players are flipped with probability chosen such that the expected corrupted-match rate is approximately δ . Ranking quality is measured by Kendall’s τ between the inferred skill means and the true skill ranking. Outlier detection is measured by AUROC, using low inferred precision as evidence of corruption.

Figure 3 shows that both of our models outperform vanilla TrueSkill in the presence of noise: Both models achieve higher values of Kendall’s τ than the vanilla model in both settings (a, b), indicating that they can recover the true ranking better. Under independent match corruption, the per-match robust model assigns lower precision to corrupted comparisons and achieves strong corrupted-match detection (a & c, green dotted line). Under player-correlated corruption, the consistent model identifies unreliable players through low per-player precision and preserves the global ranking more effectively than vanilla TrueSkill (b & c, blue solid line). Figure 3 c & d further show that, for player-correlated noise, the consistent model produces inferred skills that remain closer to the true rank order, while players with many corrupted incident matches exhibit larger uncertainty, while the vanilla model is as confident in them as in players without corrupted incidents.

Blitz chess (real-world). To showcase the large-scale applicability of our proposed TrueSkill models, we evaluate them on a large corpus of blitz chess games extracted from the ChessBase Big Database 2025 recorded in the three-year period 2022-2024 and the nine-year period 2016-2024, consisting of $n = 611\,057$ games from $k = 13\,037$ players (and $n = 1\,052\,371$ games from $k = 26\,149$ players, respectively). This corresponds to a factor graph with $1\,846\,208$ ($3\,183\,262$) variables and $2\,457\,265$ ($4\,235\,633$) factor nodes for the robust extension (Model 1), and $1\,859\,245$ ($3\,209\,411$) and $2\,470\,302$ ($4\,261\,782$) nodes for the consistency extension (Model 2). For the three-years data, the hybrid EP-VI message passing scheme converges in 26.42 seconds for Model 1 (30.27 seconds for Model 2) in a single-threaded Julia implementation on a consumer laptop (Apple M3 2024, 16GB RAM). Running on the approximately twice larger nine-year data took 60.08 seconds and 55.12 seconds, respectively, indicating linear runtime scaling.

Player skills are initialized with a Gaussian prior $s_i \sim \mathcal{N}(1\,500, 350^2)$, and the fixed performance scale is set to $\beta = 220$. For the robust and consistent models, performance precision is given a Gamma prior corresponding to a Student- t noise model with $\nu = 4$ degrees of freedom. Smaller

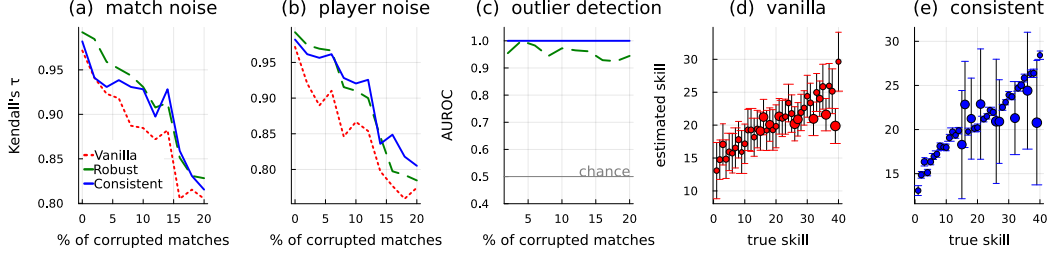


Figure 3: Performance Comparison: Robustness of our proposed TrueSkill extensions under structured corruption. **(a, b)** Kendall’s τ (higher is better) between the true and inferred skill ranking under independent match corruption in (a) and player-correlated corruption in (b), respectively. **(c)** AUROC for detecting corrupted matches using the per-match precision estimates from the robust model (green dotted line) and noisy players using the per-player precision estimates from the consistent model (blue solid line). **(d, e)** Inferred skills vs. true skills for vanilla TrueSkill and the consistent model under player-correlated corruption at the largest corruption level (20 %); error bars indicate $\pm 3\sigma_i$, and marker size is proportional to the fraction of a player’s matches that were corrupted.

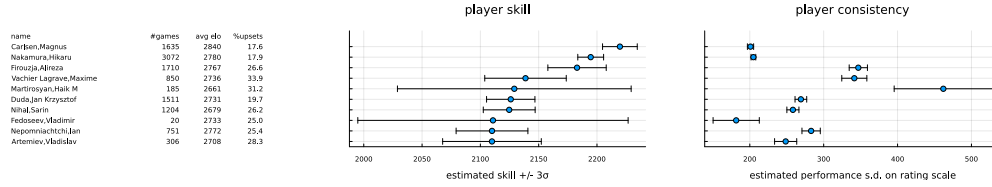


Figure 4: Consistent TrueSkill estimates for the top-10 players in blitz chess between 2022 and 2024 according to our model. Upsets describe winning against a better, or losing against a worse player, according to ELO. **(left)** Posterior skill mean, with horizontal error bars equal to $3\sigma_i$. **(right)** Estimated performance standard deviation scale $\beta \cdot \mathbb{E}_q[B_i^{-1}]^{0.5}$; horizontal error bars show uncertainty in the inferred deviation. Larger values indicate inconsistent playing behavior (lower is better).

values of ν allow heavier-tailed noise, so this setting lets the model partially absorb surprising outcomes as noisy games or volatile player performance.

For visualization, we select the top-10 players according to our estimated skill and report their estimated skills in Figure 4. Furthermore, as a proxy of the players’ consistency, we report $\beta \mathbb{E}_q[B_i^{-1}]^{0.5}$ which can be interpreted as the player’s estimated performance standard deviation on the rating scale. Figure 4 shows several interesting results. First, the top-10 players according to our model align closely with widely accepted rankings of the strongest blitz players in chess, with Carlsen and Nakamura identified as the highest-skilled and most consistent. Players with comparatively higher upset rates, such as Vachier-Lagrave, are inferred to exhibit greater performance variability, consistent with their more volatile game outcomes. As in vanilla TrueSkill, posterior uncertainty increases in regions of sparse data, reflecting limited observational evidence. Notably, despite the dataset containing only 20 games for Fedoseev, the model still assigns him a high skill estimate, albeit with correspondingly large uncertainty in both skill and consistency.

5 Conclusion

We introduced two heteroscedastic extensions of TrueSkill. The first assigns a precision multiplier to each match, producing a robust heavy-tailed comparison model that can downweight anomalous outcomes. The second assigns a precision multiplier to each player, capturing persistent differences in performance consistency across players. Both extensions retain the modular factor-graph structure of TrueSkill and admit efficient approximate inference through a hybrid EP-VI message-passing procedure. Empirically, the models improve robustness on controlled corruption experiments, allow investigation of a players consistency besides their skill, and scale to large chess datasets. The proposed model opens different directions for future research, such as (i) combined models, (ii) multi-player extensions, (iii) through time variants, and (iv) application in other domains.

References

- Junzhe Chen, Xuming Hu, Shuodi Liu, Shiyu Huang, Wei-Wei Tu, Zhaofeng He, and Lijie Wen. LLMArena: Assessing capabilities of large language models in dynamic multi-agent environments. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13055–13077, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.705. URL <https://aclanthology.org/2024.acl-long.705/>.
- Pierre Dangauthier, Ralf Herbrich, Tom Minka, and Thore Graepel. Trueskill through time: Revisiting the history of chess. In *Advances in Neural Information Processing Systems 20*, pages 337–344, 2007.
- H. A. David. *The Method of Paired Comparisons*. Oxford University Press, New York, 1988.
- Arpad E. Elo. *The Rating of Chessplayers, Past and Present*. Batsford, London, 1978.
- Mark E. Glickman. Parameter estimation in large dynamic paired comparison experiments. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 48(3):377–394, 1999. doi: 10.1111/1467-9876.00159.
- Mark E. Glickman. Dynamic paired comparison models with stochastic variances. *Journal of Applied Statistics*, 28(6):673–689, 2001. doi: 10.1080/02664760120059219.
- Mark E. Glickman. Example of the Glicko-2 system. Online manuscript, March 2022. URL <https://www.glicko.net/glicko/glicko2.pdf>. March 22, 2022.
- Thore Graepel, Joaquin Quiñonero Candela, Thomas Borchert, and Ralf Herbrich. Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft’s bing search engine. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML’10, page 1320, Madison, WI, USA, 2010. Omnipress.
- Shengbo Guo, Scott Sanner, Thore Graepel, and Wray Buntine. Score-based bayesian skill learning. In Peter A. Flach, Tijl De Bie, and Nello Cristianini, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 106–121, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-33460-3.
- Ralf Herbrich, Tom Minka, and Thore Graepel. TrueSkill: A bayesian skill rating system. In *Advances in Neural Information Processing Systems 19 (NIPS 2006)*, pages 569–576. MIT Press, 2007.
- Thomas P Minka. Expectation propagation for approximate bayesian inference. *arXiv preprint arXiv:1301.2294*, 2013.
- Tom Minka, Ryan Cleven, and Yordan Zaykov. TrueSkill 2: An improved bayesian skill rating system. Technical Report MSR-TR-2018-8, Microsoft Research, March 2018. URL <https://www.microsoft.com/en-us/research/publication/trueskill-2-improved-bayesian-skill-rating-system/>.
- Nikhil Naik, Jade Philipoom, Ramesh Raskar, and César Hidalgo. Streetscore-predicting the perceived safety of one million streetscapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 779–785, 2014.
- Keisuke Sakaguchi, Matt Post, and Benjamin Van Durme. Efficient elicitation of annotations for human evaluation of machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 1–11, 2014.
- Cheng Zhang, Judith Butepage, Hedvig Kjellstrom, and Stephan Mandt. Advances in variational inference, 2018. URL <https://arxiv.org/abs/1711.05597>.

Table 2: Notation used throughout the paper.

Symbol	Meaning
i, j	Player indices
m	Match index
i_m, j_m	Players participating in match m
y_m	Observed outcome of match m
ϵ	Draw margin
S_i	Latent skill of player i
μ_i	Prior or current mean skill estimate for player i
σ_i^2	Prior or current skill uncertainty for player i
$P_i^{(m)}$	Latent performance of player i in match m
D_m	Latent performance difference $P_{i_m}^{(m)} - P_{j_m}^{(m)}$ in match m
$\tilde{D}_m$	Skill difference $S_{i_m} - S_{j_m}$
β^2	Global performance variance in vanilla TrueSkill
B_m	Match-specific relative-precision multiplier variable for match m
B_i	Player-specific relative-precision multiplier of player i
κ^2	Baseline variance of the player performance or performance difference
$\text{Gamma}(\alpha, \lambda)$	Gamma distribution with shape α and rate λ
$q(\cdot)$	Approximate posterior belief
$m_{f \rightarrow X}(\cdot)$	Message from factor f to variable X
ρ	Gaussian precision, $\rho = \sigma^{-2}$
τ	Gaussian precision-weighted mean, $\tau = \mu/\sigma^2$
$\mathcal{G}(x; \tau, \rho)$	Gaussian density in natural parameters, $\mathcal{N}(x; \tau/\rho, \rho^{-1})$
$\Phi(\cdot)$	Standard normal cumulative distribution function
$t_\nu(\cdot)$	Standard Student- t cumulative distribution function with ν degrees of freedom

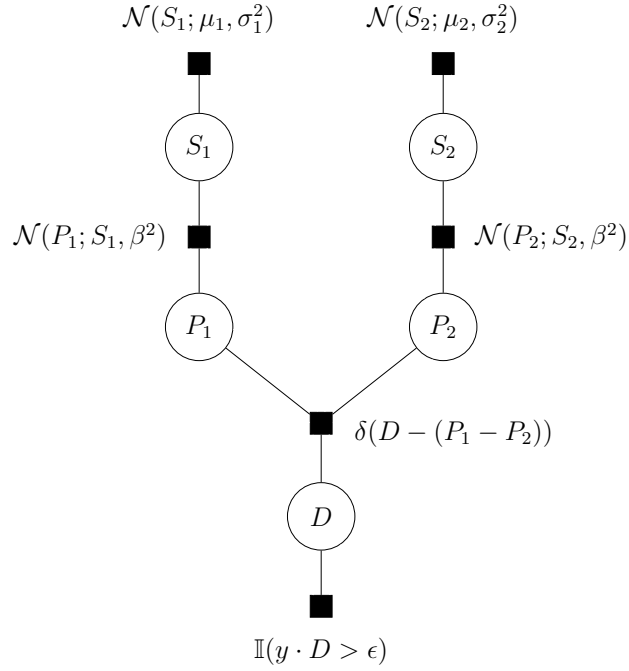


Figure 5: The vanilla TrueSkill factor graph, see [Herbrich et al., 2007]. Round nodes represent variables: S_i for the *skill* and P_i for the *performance* of player i , while D represents their *difference* in performance. Square nodes represent *factors* (functions of the connected variables). The full graph factorizes into the product of the factors, where $\mathcal{N}(S_i; \mu_i, \sigma_i^2)$ encode the prior and the product of the remaining factors encode the likelihood of the observed game outcome y ($y = 1$ for a win of player 1 and $y = -1$ for a win of player 2).

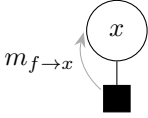
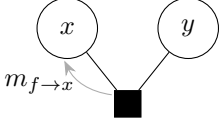
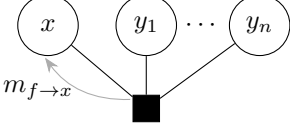
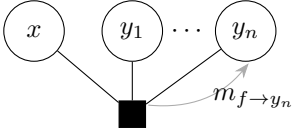
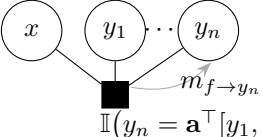
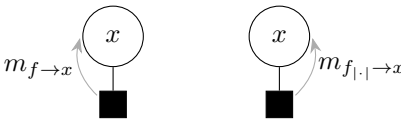
Factor	Update equation
 $\mathcal{N}(x; m, v^2)$	$\rho_x \leftarrow \frac{1}{v^2}, \quad \tau_x \leftarrow \frac{m}{v^2}$
 $\mathcal{N}(x; y, c^2)$	$\begin{aligned} \rho_{f \to x}^{\text{new}} &\leftarrow a(\rho_y - \rho_{f \to y}) \\ \tau_{f \to x}^{\text{new}} &\leftarrow a(\tau_y - \tau_{f \to y}) \\ a &:= \left(1 + c^2(\rho_y - \rho_{f \to y})\right)^{-1} \\ m_{f \to y} &\text{ follows from } \mathcal{N}(x; y, c^2) = \mathcal{N}(y; x, c^2). \end{aligned}$
 $\mathbb{I}(x = \mathbf{a}^\top \mathbf{y})$	$\begin{aligned} \rho_{f \to x}^{\text{new}} &\leftarrow \left(\sum_{j=1}^n \frac{a_j^2}{\rho_{y_j} - \rho_{f \to y_j}} \right)^{-1} \\ \tau_{f \to x}^{\text{new}} &\leftarrow \rho_{f \to x}^{\text{new}} \left(\sum_{j=1}^n a_j \cdot \frac{\tau_{y_j} - \tau_{f \to y_j}}{\rho_{y_j} - \rho_{f \to y_j}} \right) \end{aligned}$
 $\mathbb{I}(x = \mathbf{b}^\top \mathbf{y})$	 $\mathbf{a} = \frac{1}{b_n} \begin{bmatrix} -b_1 \\ \vdots \\ -b_{n-1} \\ +1 \end{bmatrix}$ $\mathbb{I}(y_n = \mathbf{a}^\top [y_1, \dots, y_{n-1}, x])$
 $\mathbb{I}(x > \epsilon) \quad \mathbb{I}(x \leq \epsilon)$	$\begin{aligned} \rho_x^{\text{new}} &\leftarrow \frac{c}{1 - W_f(d/\sqrt{c}, \epsilon\sqrt{c})} \\ \tau_x^{\text{new}} &\leftarrow \frac{d + \sqrt{c} V_f(d/\sqrt{c}, \epsilon\sqrt{c})}{1 - W_f(d/\sqrt{c}, \epsilon\sqrt{c})} \\ c &:= \rho_x - \rho_{f \to x}, \quad d := \tau_x - \tau_{f \to x} \end{aligned}$

Table 3: Factor types and update equations of vanilla TrueSkill, see Herbrich et al. [2007].

330 C Complete Proof of Lemma 1

331 *Proof.* The factor can be written as

$$f(X, Y, Z) \propto Z^{1/2} \exp\left\{-\frac{Z}{2}(X - Y)^2\right\}.$$

332 For the message to X , we collect only the terms depending on X :

$$-\frac{Z}{2}(X - Y)^2 = -\frac{Z}{2}X^2 + ZYX - \frac{Z}{2}Y^2.$$

333 Dropping terms independent of X , this gives

$$f(X, Y, Z) \propto \exp\left\{ZYX - \frac{Z}{2}X^2\right\}.$$

334 Equivalently,

$$f(X, Y, Z) \propto \exp\left(\underbrace{\begin{bmatrix} ZY \\ Z \end{bmatrix}^\top}_{\eta_X(Y, Z)^\top} \underbrace{\begin{bmatrix} X \\ -\frac{1}{2}X^2 \end{bmatrix}}_{T_X(X)}\right).$$

335 The variational message replaces the natural parameters by their expectations with respect to the
336 current neighboring marginals:

$$m_{f \rightarrow X}(x) \propto \exp(\mathbb{E}[\eta_X(Y, Z)]^\top T_X(x)).$$

337 Under the mean-field factorization,

$$\mathbb{E}[ZY] = \mathbb{E}[Z]\mathbb{E}[Y],$$

338 and therefore

$$m_{f \rightarrow X}(x) \propto \exp\left\{\mathbb{E}[Z]\mathbb{E}[Y]x - \frac{\mathbb{E}[Z]}{2}x^2\right\}.$$

339 This is a Gaussian message with natural parameters

$$\tau_X = \mathbb{E}[Y]\mathbb{E}[Z], \quad \rho_X = \mathbb{E}[Z],$$

340 so

$$m_{f \rightarrow X}(x) \propto \mathcal{G}(x; \mathbb{E}[Y]\mathbb{E}[Z], \mathbb{E}[Z]).$$

341 The argument for the message to Y is symmetric. Collecting the terms depending on Y gives

$$-\frac{Z}{2}(X - Y)^2 = -\frac{Z}{2}Y^2 + ZXY - \frac{Z}{2}X^2.$$

342 Dropping terms independent of Y and taking expectations over the neighboring variables yields

$$m_{f \rightarrow Y}(y) \propto \exp\left\{\mathbb{E}[Z]\mathbb{E}[X]y - \frac{\mathbb{E}[Z]}{2}y^2\right\}.$$

343 Hence

$$m_{f \rightarrow Y}(y) \propto \mathcal{G}(y; \mathbb{E}[X]\mathbb{E}[Z], \mathbb{E}[Z]).$$

344 Finally, for the message to Z , we collect the terms depending on Z :

$$f(X, Y, Z) \propto Z^{1/2} \exp\left\{-\frac{Z}{2}(X - Y)^2\right\}.$$

345 This can be expressed in exponential-family form as

$$f(X, Y, Z) \propto \exp\left(\underbrace{\begin{bmatrix} \frac{1}{2} \\ \frac{1}{2}(X - Y)^2 \end{bmatrix}^\top}_{\eta_Z(X, Y)^\top} \underbrace{\begin{bmatrix} \log Z \\ -Z \end{bmatrix}}_{T_Z(Z)}\right).$$

Thus the variational outgoing message is

$$m_{f \rightarrow Z}(z) \propto \exp(\mathbb{E}[\eta_Z(X, Y)]^\top T_Z(z)),$$

which gives

$$m_{f \rightarrow Z}(z) \propto z^{1/2} \exp\left\{-\frac{z}{2} \mathbb{E}[(X - Y)^2]\right\}.$$

Since

$$\mathbb{E}[(X - Y)^2] = \mathbb{E}[X^2] + \mathbb{E}[Y^2] - 2\mathbb{E}[XY],$$

and, under the mean-field approximation,

$$\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y],$$

we obtain

$$\frac{1}{2} \mathbb{E}[(X - Y)^2] = \frac{\mathbb{E}[X^2] + \mathbb{E}[Y^2]}{2} - \mathbb{E}[X]\mathbb{E}[Y].$$

Because

$$z^{1/2} = z^{3/2-1},$$

the message is Gamma with shape 3/2 and rate

$$\frac{\mathbb{E}[X^2] + \mathbb{E}[Y^2]}{2} - \mathbb{E}[X]\mathbb{E}[Y].$$

Therefore,

$$m_{f \rightarrow Z}(z) \propto \text{Gamma}\left(z; \frac{3}{2}, \frac{\mathbb{E}[X^2] + \mathbb{E}[Y^2]}{2} - \mathbb{E}[X]\mathbb{E}[Y]\right).$$

□

D Proof of Limited Use of Expectation Propagation Techniques

We show that EP cannot be consistently applied in the presence of Gaussian–Gamma precision coupling, due to variance inflation induced by scale mixtures.

Proof of EP inconsistency via variance mismatch. Consider a factor of the form

$$f(x, y, z) = \mathcal{N}(x; y, z^{-1}),$$

where y is a latent mean parameter and z is a latent precision variable with $z \sim \text{Gamma}(\alpha, \beta)$.

In EP, the marginal distribution over y induced by the factor and incoming messages is approximated by a Gaussian via moment matching:

$$\hat{p}(y) = \mathcal{N}(y; \mu, \sigma^2), \quad \mu = \mathbb{E}[Y], \quad \sigma^2 = \text{Var}(Y).$$

Due to the integration over the random precision z , the true marginal becomes a Gaussian scale mixture:

$$p(y) = \int \mathcal{N}(y; \mu, z^{-1}) p(z) dz,$$

which induces variance inflation relative to any fixed-precision Gaussian approximation.

In particular, there exist parameter regimes (e.g. small α) such that

$$\text{Var}(Y) > \text{Var}_{\text{cavity}}(Y),$$

where $\text{Var}_{\text{cavity}}(Y)$ denotes the variance of the incoming EP cavity distribution.

The EP site update for y is obtained via precision subtraction:

$$\rho_{f \rightarrow y} = \frac{1}{\sigma_{\text{post}}^2} - \frac{1}{\sigma_{\text{cavity}}^2}.$$

If $\sigma_{\text{post}}^2 > \sigma_{\text{cavity}}^2$, this implies

$$\rho_{f \rightarrow y} < 0,$$

which is not admissible for a valid Gaussian site.

Therefore, no consistent Gaussian EP factorization exists for this model class, as EP moment matching would require negative variances. □

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: [Yes]

Justification: We provide a novel message passing method for TrueSkill extensions.

Guidelines:

- The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: We discuss our results in Section 4, page 8-9.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate “Limitations” section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren’t acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: See Appendix (Proofs).

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We use reproducible problem examples and provide supplemental code.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide supplemental code and will make it open access.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://neurips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://neurips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: We explain all chosen (hyper)parameters.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: See Figure 3 and 4, page 9.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: See Section 4 (single-threaded Julia implementation on a consumer laptop (Apple M3 2024, 16GB RAM)).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: Given the considered problem, there are no ethical issues.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We point to potential applications of the solution.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: There are no such risks.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [N/A]

Justification: We cite relevant work.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset’s creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper does not release new assets.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

680 Question: Does the paper describe the usage of LLMs if it is an important, original, or
681 non-standard component of the core methods in this research? Note that if the LLM is used
682 only for writing, editing, or formatting purposes and does *not* impact the core methodology,
683 scientific rigor, or originality of the research, declaration is not required.

684 Answer: [N/A]

685 Justification: We used LLMs only for improving grammar and spelling.

686 Guidelines:

- 687 • The answer [N/A] means that the core method development in this research does not
- 688 involve LLMs as any important, original, or non-standard components.
- 689 • Please refer to our LLM policy in the NeurIPS handbook for what should or should not
- 690 be described.