

1 Player skill graphical models

1.a

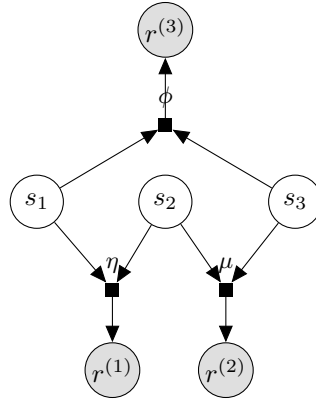


Figure 1: Factor graph of simple player skill model when observing three games between three players. The games are denoted as the factors ϕ , μ and η . Game outcomes are denoted as $r^{(k)}$, where $k \in \{1, 2, 3\}$ being the different games played. The result of the games, denoted $r^{(k)} \in \{0, 1\}$ with 0 indicating a loss and 1 indicating a win. The players' skills are identified as s_1 , s_2 and s_3 , where the index denotes the ID of the player $\in \{1, 2, 3\}$ respectively.

1.b

1.b.1 $I(s_1, s_2 | r^{(2)}) \checkmark$

There are 2 paths leading from s_1 to s_2 , namely $s_1 - \eta - s_2$ and $s_1 - \phi - s_3 - \mu - s_2$. The former path is blocked, as η has two incoming edges and neither η , nor $r^{(1)}$ (being all descendants) are in the conditioning set (being $r^{(2)}$). The latter is also blocked, but the case is slightly more complicated. Firstly, μ has two incoming edges, however, its descendant is in the conditioning set ($r^{(2)}$), therefore, the path is not blocked there. Secondly, ϕ also has two incoming edges and neither ϕ , nor $r^{(3)}$ are in the conditioning set, therefore, the path is blocked there. Thus, as the two paths are blocked $\Rightarrow I(s_1, s_2 | r^{(2)})$.

1.b.2 $I(s_1, s_2 | r^{(2)}, r^{(3)}) \times$

As identified in Section 1.b.1, there are 2 paths. The path $s_1 - \eta - s_2$ is still blocked, as η , nor any of its descendants are in the conditioning set. The path $s_1 - \phi - s_3 - \mu - s_2$, however, is now unblocked, as for all factors and variables with 2 incoming edges, there is a child that belongs

to the conditioning set, namely $r^{(3)}$ is a child of ϕ and $r^{(2)}$ is a child of μ , both of them being in the conditioning set. Therefore, s_1, s_2 are conditionally dependent on the set $\{r^{(2)}, r^{(3)}\}$.

1.c

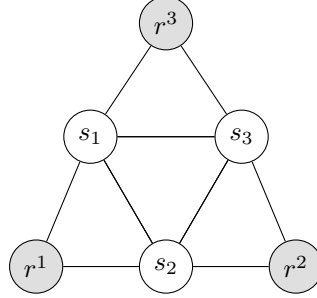


Figure 2: Markov network of simple player skill model when observing three games between three players. The observed and latent variables are denoted in a similar way, as Figure 1. Game outcomes are denoted as $r^{(k)}$, where $k \in \{1, 2, 3\}$ being the different games played. The result of the games, denoted $r^{(k)} \in \{0, 1\}$ with 0 indicating a loss and 1 indicating a win. The players' skills are identified as s_1, s_2 and s_3 , where the index denotes the ID of the player $\in \{1, 2, 3\}$ respectively.

1.d

In order to avoid repetition I will use the indices i, j, k , where $i \neq j \neq k$. Equation 1 and 2 generalise the conditional independence statements in Figure 2. Also, the following rules must hold: $s_i, s_j \in ne(r^{(i)}) || s_i, s_j \in ne(r^{(j)})$ for Equation 1 and $s_i, s_j \in ne(r^{(i)})$ for Equation 2.

$$I(r^{(i)}, r^{(j)} | s_i, s_j, \{s_k, r^{(k)}\}) \quad (1)$$

$$I(r^{(i)}, s_k | s_i, s_j, \{r^{(j)}, r^{(k)}\}) \quad (2)$$

The set in the conditioning part of Equation 1 means that including any combination or no combination of the elements inside the set would still work.

There are no conditional independence statements between the variables s_1, s_2, s_3 , as disregarding how many r and s variables we remove, due to the u-separation rule, we cannot make them conditionally independent.

In other words, the games are only dependent on the players that play them.

2 Question 2: Gaussian player skill model

2.a

From the figure, one can express the joint probability distribution as follows:

$$p(p_b, p_w, s_b, s_w) = \frac{1}{Z} \phi_1(p_w) \phi_2(s_w, p_w) \phi_3(s_b) \phi_4(s_b, p_b) \phi_5(p_w, p_b, r) \quad (3)$$

Thereafter, using chain rule, $p(p_b, p_w | s_b, s_w)$ can be expressed as follows:

$$p(p_b, p_w | s_b, s_w) = \frac{p(p_b, p_w, s_b, s_w)}{p(s_b, s_w)} = \frac{p(p_b, p_w, s_b, s_w)}{\int_{p_b, p_w} p(p_b, p_w, s_b, s_w)} = \quad (4)$$

$$\frac{\frac{1}{Z} \phi_1(s_w) \phi_2(s_w, p_w) \phi_3(s_b) \phi_4(s_b, p_b) \phi_5(p_w, p_b, r)}{\int_{p_b, p_w} \frac{1}{Z} \phi_1(s_w) \phi_2(s_w, p_w) \phi_3(s_b) \phi_4(s_b, p_b) \phi_5(p_w, p_b, r)} \quad (5)$$

$$\frac{\frac{1}{Z} \phi_1(p_w) \phi_2(s_w, p_w) \phi_3(p_b) \phi_4(s_b, p_b) \phi_5(p_w, p_b, r)}{\frac{1}{Z} \phi_1(p_w) \phi_3(p_b) \int_{p_b, p_w} \phi_2(s_w, p_w) \phi_4(s_b, p_b) \phi_5(p_w, p_b, r)} \quad (6)$$

$$\frac{\phi_2(s_w, p_w) \phi_4(s_b, p_b) \phi_5(p_w, p_b, r)}{\int_{p_b, p_w} \phi_2(s_w, p_w) \phi_4(s_b, p_b) \phi_5(p_w, p_b, r)} \quad (7)$$

$$\frac{\phi_2(s_w, p_w) \phi_4(s_b, p_b)}{\int_{p_b, p_w} \phi_2(s_w, p_w) \phi_4(s_b, p_b)} \quad (8)$$

$$\phi_2(s_w, p_w) \phi_4(s_b, p_b) \quad (9)$$

2.b

2.b.1 $\mathbb{E}[\theta | s_b, s_w]$

$$\mathbb{E}[\theta | s_w, s_b] = \frac{1}{\sqrt{2}\beta} \mathbb{E}[(p_w - s_w) + (p_b - s_b) | s_w, s_b] \quad (10)$$

$$\frac{1}{\sqrt{2}\beta} \left(\mathbb{E}[p_w + p_b | s_w, s_b] - s_w - s_b \right) \quad (11)$$

$$\frac{1}{\sqrt{2}\beta} \left(\mathbb{E}[p_w | s_w, s_b] + \mathbb{E}[p_b | s_w, s_b] - s_w - s_b \right) \quad (12)$$

$$\frac{1}{\sqrt{2}\beta} \left(s_w + s_b - s_w - s_b \right) = 0 \quad (13)$$

2.b.2 $\mathbb{E}[\psi | s_b, s_w]$

$$\mathbb{E}[\psi | s_w, s_b] = \frac{1}{\sqrt{2}\beta} \mathbb{E}[(p_w - s_w) - (p_b - s_b) | s_w, s_b] \quad (14)$$

$$\frac{1}{\sqrt{2}\beta} \left(\mathbb{E}[p_w - p_b | s_w, s_b] - s_w + s_b \right) \quad (15)$$

$$\frac{1}{\sqrt{2}\beta} \left(\mathbb{E}[p_w | s_w, s_b] - \mathbb{E}[p_b | s_w, s_b] - s_w + s_b \right) \quad (16)$$

$$\frac{1}{\sqrt{2}\beta} \left(s_w - s_b - s_w + s_b \right) = 0 \quad (17)$$

2.b.3 $\mathbb{E}[\theta^2 \mid s_b, s_w]$

$$\mathbb{E}[\theta^2 \mid s_w, s_b] = \frac{1}{2\beta^2} \mathbb{E}[(p_w - s_w) + (p_b - s_b)]^2 \mid s_w, s_b \quad (18)$$

$$\frac{1}{2\beta^2} \left(\underbrace{\mathbb{E}[(p_w - s_w)^2 \mid s_w, s_b]}_A + \underbrace{2\mathbb{E}[(p_w - s_w)(p_b - s_b) \mid s_w, s_b]}_B + \underbrace{\mathbb{E}[(p_b - s_b)^2 \mid s_w, s_b]}_C \right) \quad (19)$$

$$A = \mathbb{E}[p_w^2 \mid s_w, s_b] - 2s_w \mathbb{E}[p_w \mid s_w, s_b] + s_w^2 \quad (20)$$

$$A = \text{Var}(p_w) + \mathbb{E}[p_w \mid s_w, s_b]^2 - 2s_w^2 + s_w^2 = \text{Var}(p_w) = \beta^2 \quad (21)$$

$$B = 2\mathbb{E}[p_b p_w \mid s_w, s_b] - 2s_b \mathbb{E}[p_w \mid s_w, s_b] - 2s_w \mathbb{E}[p_b \mid s_w, s_b] + 2s_b s_w \quad (22)$$

$$B = 2 \left(\mathbb{E}[p_b \mid s_w, s_b] \mathbb{E}[p_w \mid s_w, s_b] + \text{Cov}(p_b, p_w) \right) - 2s_b s_w - 2s_w s_b + 2s_b s_w \quad (23)$$

$$B = 2 \left(\mathbb{E}[p_b \mid s_w, s_b] \mathbb{E}[p_w \mid s_w, s_b] + \text{Cov}(p_b, p_w) \right) - 2s_b s_w \quad (24)$$

$$B = 2 \left(s_b s_w + \underbrace{\text{Cov}(p_b, p_w)}_{p_w \text{ and } p_b \text{ are CI} \Rightarrow \text{Cov}(p_w, p_b) = 0} \right) - 2s_b s_w = 0 \quad (25)$$

$$C = \mathbb{E}[p_b^2 \mid s_w, s_b] - 2s_b \mathbb{E}[p_b \mid s_w, s_b] + s_b^2 \quad (26)$$

$$C = \text{Var}(p_b) + \mathbb{E}[p_b \mid s_w, s_b]^2 - 2s_b^2 + s_b^2 = \text{Var}(p_b) = \beta^2 \quad (27)$$

From Equation 19:

$$\frac{1}{2\beta^2} (A + B + C) = \frac{\beta^2 + 0 + \beta^2}{2\beta^2} = 1 \quad (28)$$

2.b.4 $\mathbb{E}[\psi^2 \mid s_b, s_w]$

$$\mathbb{E}[\psi^2 \mid s_w, s_b] = \frac{1}{2\beta^2} \mathbb{E}[(p_w - s_w) - (p_b - s_b)]^2 \mid s_w, s_b \quad (29)$$

$$\frac{1}{2\beta^2} \left(\underbrace{\mathbb{E}[(p_w - s_w)^2 \mid s_w, s_b]}_D - \underbrace{2\mathbb{E}[(p_w - s_w)(p_b - s_b) \mid s_w, s_b]}_E + \underbrace{\mathbb{E}[(p_b - s_b)^2 \mid s_w, s_b]}_F \right) \quad (30)$$

$$D = \mathbb{E}[p_w^2 \mid s_w, s_b] - 2s_w \mathbb{E}[p_w \mid s_w, s_b] + s_w^2 \quad (31)$$

$$D = \text{Var}(p_w) + \mathbb{E}[p_w \mid s_w, s_b]^2 - 2s_w^2 + s_w^2 = \text{Var}(p_w) = \beta^2 \quad (32)$$

$$E = 2\mathbb{E}[p_b p_w \mid s_w, s_b] - 2s_b \mathbb{E}[p_w \mid s_w, s_b] - 2s_w \mathbb{E}[p_b \mid s_w, s_b] + 2s_b s_w \quad (33)$$

$$E = 2 \left(\mathbb{E}[p_b \mid s_w, s_b] \mathbb{E}[p_w \mid s_w, s_b] + \text{Cov}(p_b, p_w) \right) - 2s_b s_w - 2s_w s_b + 2s_b s_w \quad (34)$$

$$E = 2 \left(\mathbb{E}[p_b \mid s_w, s_b] \mathbb{E}[p_w \mid s_w, s_b] + \text{Cov}(p_b, p_w) \right) - 2s_b s_w \quad (35)$$

$$E = 2 \left(s_b s_w + \underbrace{\text{Cov}(p_b, p_w)}_{p_w \text{ and } p_b \text{ are CI} \Rightarrow \text{Cov}(p_w, p_b) = 0} \right) - 2s_b s_w = 0 \quad (36)$$

$$F = \mathbb{E}[p_b^2 \mid s_w, s_b] - 2s_b \mathbb{E}[p_b \mid s_w, s_b] + s_b^2 \quad (37)$$

$$F = \text{Var}(p_b) + \mathbb{E}[p_b \mid s_w, s_b]^2 - 2s_b^2 + s_b^2 = \text{Var}(p_b) = \beta^2 \quad (38)$$

From Equation 30:

$$\frac{1}{2\beta^2}(D - E + F) = \frac{\beta^2 - 0 + \beta^2}{2\beta^2} = 1 \quad (39)$$

2.b.5 $\mathbb{E}[\psi\theta \mid s_b, s_w]$

$$\mathbb{E}[\psi\theta \mid s_w, s_b] = \frac{1}{2\beta^2} \mathbb{E}[(p_w - s_w)^2 - (p_w - s_w)(p_b - s_b) + (p_w - s_w)(p_b - s_b) - (p_b - s_b)^2 \mid s_w, s_b] \quad (40)$$

$$\mathbb{E}[\psi\theta \mid s_w, s_b] = \frac{1}{2\beta^2} \mathbb{E}[(p_w - s_w)^2 - (p_w - s_w)(p_b - s_b) + (p_w - s_w)(p_b - s_b) - (p_b - s_b)^2 \mid s_w, s_b] \quad (41)$$

$$\mathbb{E}[\psi\theta \mid s_w, s_b] = \frac{1}{2\beta^2} \left(\underbrace{\mathbb{E}[(p_w - s_w)^2 \mid s_w, s_b]}_G - \underbrace{\mathbb{E}[(p_b - s_b)^2 \mid s_w, s_b]}_H \right) \quad (42)$$

$$G = \mathbb{E}[p_w^2 \mid s_w, s_b] - 2s_w \mathbb{E}[p_w \mid s_w, s_b] + s_w^2 \quad (43)$$

$$G = \text{Var}(p_w) + \mathbb{E}[p_w \mid s_w, s_b]^2 - 2s_w^2 + s_w^2 = \text{Var}(p_w) = \beta^2 \quad (44)$$

$$H = \mathbb{E}[p_b^2 \mid s_w, s_b] - 2s_b \mathbb{E}[p_b \mid s_w, s_b] + s_b^2 \quad (45)$$

$$H = \text{Var}(p_b) + \mathbb{E}[p_b \mid s_w, s_b]^2 - 2s_b^2 + s_b^2 = \text{Var}(p_b) = \beta^2 \quad (46)$$

From Equation 42:

$$G - H = \frac{\beta^2 - \beta^2}{2\beta^2} = 0 \quad (47)$$

2.b.6 $\mathbb{P}[\theta, \psi \mid s_b, s_w]$

$\mathbb{P}(\theta \mid s_b, s_w)$ is a distribution, however, as it is composed of p_w, p_b, s_w, s_b , that are all Gaussian distributions, $\mathbb{P}(\theta \mid s_b, s_w)$ is also a Gaussian distribution. Its mean($\mathbb{E}[\theta \mid s_b, s_w]$) is 0 and its variance is 1 ($\mathbb{E}[\theta^2 \mid s_b, s_w]$), therefore, $\mathbb{P}(\theta \mid s_b, s_w)$ is a standard normal Gaussian distribution.

$\mathbb{P}(\psi \mid s_b, s_w)$ is a distribution, however, as it is composed of p_w, p_b, s_w, s_b , that are all Gaussian distributions, $\mathbb{P}(\psi \mid s_b, s_w)$ is also a Gaussian distribution. Its mean($\mathbb{E}[\psi \mid s_b, s_w]$) is 0 and its variance is 1 ($\mathbb{E}[\psi^2 \mid s_b, s_w]$), therefore, $\mathbb{P}(\psi \mid s_b, s_w)$ is a standard normal Gaussian distribution.

Based on the 2 aforementioned statements, the joint distribution $\mathbb{P}(\theta, \psi \mid s_b, s_w)$ is a joint standard multivariate Gaussian distribution, expressed as follows:

$$\mathbb{P}(\theta, \psi \mid s_b, s_w) = \frac{1}{\sqrt{(2\pi)^2 \mid \Sigma \mid}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right) \quad (48)$$

With the following x, μ and Σ :

$$x = \begin{bmatrix} \theta \\ \psi \end{bmatrix} \quad (49)$$

$$\mu = \begin{bmatrix} \mathbb{E}[\theta \mid s_b, s_w] \\ \mathbb{E}[\psi \mid s_b, s_w] \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (50)$$

$$\Sigma = \begin{bmatrix} \text{Var}(\theta) & \text{Cov}(\theta, \psi) \\ \text{Cov}(\theta, \psi) & \text{Var}(\psi) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (51)$$

The $\text{Cov}[\theta, \psi \mid s_b, s_w] = 0$ as $\mathbb{E}[\psi\theta \mid s_b, s_w] = 0$, $\mathbb{E}[\psi \mid s_b, s_w] = 0$ and $\mathbb{E}[\theta \mid s_b, s_w] = 0$, meaning that θ and ψ are independent.

Leading to the following form:

$$\mathbb{P}(\theta, \psi \mid s_b, s_w) = \frac{1}{2\pi} \exp\left(-\frac{1}{2}(\theta^2 + \psi^2)\right) \quad (52)$$

In order to prove this result one could use the change of variable.

Firstly express p_b and p_w in terms of ψ and θ .

$$p_b = -\sqrt{2}\beta\psi + p_w - s_w + s_b \quad (53)$$

$$p_w = \sqrt{2}\beta\theta + s_w - p_b + s_b \quad (54)$$

Substituting Equation 53 in Equation 54 results in:

$$p_w = \sqrt{2}\beta\theta + s_w + \sqrt{2}\beta\psi - p_w + s_w - s_b + s_b \quad (55)$$

$$p_w = \frac{\sqrt{2}\beta(\theta + \psi) + 2s_w}{2} \quad (56)$$

Substituting Equation 54 in Equation 53 results in:

$$p_b = -\sqrt{2}\beta\psi + \sqrt{2}\beta\theta + s_w - p_b + s_b - s_w + s_b \quad (57)$$

$$p_b = \frac{\sqrt{2}\beta(\theta - \psi) + 2s_b}{2} \quad (58)$$

Substituting the factors for their values in Equation 9, we get the following:

$$p(p_b, p_w | s_b, s_w) = \mathcal{N}(p_b) \mathcal{N}(p_w) = \quad (59)$$

$$\frac{1}{\beta\sqrt{2\pi}} e^{-(p_b - s_b)^2 / 2\beta^2} \frac{1}{\beta\sqrt{2\pi}} e^{-(p_w - s_w)^2 / 2\beta^2} \quad (60)$$

In order to change the variable, we would have to compute the derivative of a 2x2 Jacobian matrix given below:

$$J = \begin{bmatrix} \frac{\partial p_w}{\partial \theta} & \frac{\partial p_b}{\partial \theta} \\ \frac{\partial p_w}{\partial \psi} & \frac{\partial p_b}{\partial \psi} \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{2}\beta}{2} & \frac{\sqrt{2}\beta}{2} \\ \frac{\sqrt{2}\beta}{2} & -\frac{\sqrt{2}\beta}{2} \end{bmatrix} \quad (61)$$

The absolute value of the derivative derivative is equal to:

$$|\det(J)| = \left| \frac{\partial p_w}{\partial \theta} \frac{\partial p_b}{\partial \psi} - \frac{\partial p_b}{\partial \theta} \frac{\partial p_w}{\partial \psi} \right| = \left| \frac{\sqrt{2}\beta}{2} \left(-\frac{\sqrt{2}\beta}{2} \right) - \frac{\sqrt{2}\beta}{2} \frac{\sqrt{2}\beta}{2} \right| = \left| -\frac{2\beta^2}{4} - \frac{2\beta^2}{4} \right| = |-\beta^2| = \beta^2 \quad (62)$$

Now we can express $p(p_b, p_w | s_b, s_w)$ as $p(\theta, \psi | s_b, s_w)$, when substituting p_w and p_b from Equation 56 and 58 in Equation 63 and multiplying by the absolute value of the derivative of the Jacobian, expressed in Equation 62.

$$\int \int \frac{1}{\beta\sqrt{2\pi}} e^{-\left(\frac{\sqrt{2}\beta}{2}(\theta - \psi) + s_b - s_b\right)^2 / 2\beta^2} \frac{1}{\beta\sqrt{2\pi}} e^{-\left(\frac{\sqrt{2}\beta}{2}(\theta + \psi) + s_w - s_w\right)^2 / 2\beta^2} |\det(J)| d\theta d\psi \quad (63)$$

$$\int \int \frac{1}{2\pi\beta^2} e^{-\frac{\beta^2}{2\beta^2} \left(\frac{1}{2}(\theta - \psi)^2 + \frac{1}{2}(\theta + \psi)^2\right)} \beta^2 d\theta d\psi = \quad (64)$$

$$\int \int \frac{1}{2\pi} e^{-\frac{1}{2}(\theta^2 - \theta\psi + \theta\psi + \psi^2)} d\theta d\psi \quad (65)$$

$$p(\theta, \psi | s_b, s_w) = \frac{1}{2\pi} e^{-\frac{1}{2}(\theta^2 + \psi^2)} \quad (66)$$

2.c

We can express $\mathbb{P}(r = 1 | p_b, p_w)$ as the inequality:

$$\mathbb{1}[p_b > p_w] \quad (67)$$

Using the change of variable from Equation 56 and 58:

$$\frac{\sqrt{2}\beta(\theta - \psi) + 2s_b}{2} - \frac{\sqrt{2}\beta(\theta + \psi) + 2s_w}{2} > 0 \quad (68)$$

$$\sqrt{2}\beta(\theta - \psi) + 2s_b - \sqrt{2}\beta(\theta + \psi) + 2s_w > 0 \quad (69)$$

$$-\sqrt{2}\beta\psi + 2s_b - \sqrt{2}\beta\theta + 2s_w > 0 \quad (70)$$

$$2s_b + 2s_w > 2\sqrt{2}\beta\psi \quad (71)$$

$$\mathbb{P}(r = 1 | \theta, \psi, s_b, s_w) = \mathbb{1} \left[\frac{s_b + s_w}{\sqrt{2}\beta} > \psi \right] \quad (72)$$

2.d

$$\mathbb{P}(r = 1 | s_b, s_w) = \frac{\mathbb{P}(r = 1, s_b, s_w)}{\mathbb{P}(s_b, s_w)} \quad (73)$$

$$\mathbb{P}(r = 1 | s_b, s_w) = \frac{\int_{\theta, \psi} \mathbb{P}(r = 1, \theta, \psi, s_b, s_w) d\theta d\psi}{\mathbb{P}(s_b, s_w)} \quad (74)$$

$$\mathbb{P}(r = 1 | s_b, s_w) = \frac{\int_{\theta, \psi} \mathbb{P}(r = 1 | \theta, \psi, s_b, s_w) \mathbb{P}(\theta, \psi, s_b, s_w) d\theta d\psi}{\mathbb{P}(s_b, s_w)} \quad (75)$$

$$\mathbb{P}(r = 1 | s_b, s_w) = \frac{\mathbb{P}(s_b, s_w) \int_{\theta, \psi} \mathbb{P}(r = 1 | \theta, \psi, s_b, s_w) \mathbb{P}(\theta, \psi | s_b, s_w) d\theta d\psi}{\mathbb{P}(s_b, s_w)} \quad (76)$$

Substituting Equation 72, and splitting $p(\theta, \psi | s_b, s_w)$, as they are conditionally independent ($\mathbb{E}[\psi\theta | s_b, s_w] = 0$, $\mathbb{E}[\theta | s_b, s_w] = 0$, $\mathbb{E}[\psi | s_b, s_w] = 0$, therefore, $\text{Cov}(\psi, \theta) = 0$), we get:

$$\mathbb{P}(r = 1 | s_b, s_w) = \int_{\psi} \mathbb{1} \left[\frac{s_b + s_w}{\sqrt{2}\beta} > \psi \right] \mathbb{P}(\psi | s_b, s_w) d\psi \underbrace{\int_{\theta} \mathbb{P}(\theta | s_b, s_w) d\theta}_1 \quad (77)$$

Also, as $\mathbb{P}(\psi | s_b, s_w)$ has mean of 0 and variance of 1, we can model it as such:

$$\mathbb{P}(r = 1 | s_b, s_w) = \int \mathbb{1} \left[\frac{s_b + s_w}{\sqrt{2}\beta} > \psi \right] \mathcal{N}(\psi; 0, 1) d\psi = \Phi \left[\frac{s_b - s_w}{\sqrt{2}\beta} \right] \quad (78)$$

2.e

In order for a game to be observed it has to be either won by the black player or by the white player. Here, I am ignoring the fact that it could be draw, as described in the assignment sheet. Thus, we can mathematically model this as follows:

$$\mathbb{P}(r | s_b, s_w) = I(r = 1) \mathbb{P}(r = 1 | s_b, s_w) + I(r = 0) \mathbb{P}(r = 0 | s_b, s_w) \quad (79)$$

As this is a Bernoulli distribution, we can express $\mathbb{P}(r = 0 | s_b, s_w)$ as:

$$\mathbb{P}(r = 0 | s_b, s_w) = 1 - \mathbb{P}(r = 1 | s_b, s_w) \quad (80)$$

Substituting the result from Equation 78 in Equation 80:

$$\mathbb{P}(r = 0 | s_b, s_w) = 1 - \Phi \left[\frac{s_b - s_w}{\sqrt{2}\beta} \right] \quad (81)$$

Therefore, substituting Equation 81 and 78 in Equation 79:

$$\mathbb{P}(r | s_b, s_w) = I(r = 1) \Phi \left[\frac{s_b - s_w}{\sqrt{2}\beta} \right] + I(r = 0) \left(1 - \Phi \left[\frac{s_b - s_w}{\sqrt{2}\beta} \right] \right) \quad (82)$$

The above result is derived for one game. Based on Section 1.d, I assume that each observed game is independently and identically distributed. I.e. every game is independent from the other games and only has dependency relations to the players involved in this particular game, thus the expression can be rewritten as follows:

$$\mathbb{P} \left[\{r^{(k)}\}_{k=1}^N \mid \{s_i\}_{i=1}^M \right] = \prod_{k=1}^N \left[I(r^{(k)} = 1) \Phi \left[\frac{s_b - s_w}{\sqrt{2}\beta} \right] + I(r^{(k)} = 0) \left(1 - \Phi \left[\frac{s_b - s_w}{\sqrt{2}\beta} \right] \right) \right] \quad (83)$$

2.f

Adopting Bayes Rule, we can rewrite the equation in the following fashion:

$$\mathbb{P} \left[\{s_i\}_{i=1}^M \mid \{r^{(k)}\}_{k=1}^N \right] = \frac{\mathbb{P} \left[\{r^{(k)}\}_{k=1}^N \mid \{s_i\}_{i=1}^M \right] \mathbb{P} \left[\{s_i\}_{i=1}^M \right]}{\mathbb{P} \left[\{r^{(k)}\}_{k=1}^N \right]} \quad (84)$$

$$\mathbb{P} \left[\{s_i\}_{i=1}^M \mid \{r^{(k)}\}_{k=1}^N \right] \propto \mathbb{P} \left[\{r^{(k)}\}_{k=1}^N \mid \{s_i\}_{i=1}^M \right] \mathbb{P} \left[\{s_i\}_{i=1}^M \right] \quad (85)$$

Taking into account the equality $1 - \Phi(x) = \Phi(-x)$, we can rewrite Equation 83 as such:

$$\prod_{k=1}^N \left[I(r^{(k)} = 1) \Phi \left[\frac{s_b - s_w}{\sqrt{2}\beta} \right] + I(r^{(k)} = 0) \left(\Phi \left[\frac{s_w - s_b}{\sqrt{2}\beta} \right] \right) \right] \quad (86)$$

When $r^{(k)} = 1$, the black player is the winner, similarly, when $r^{(k)} = 0$, the white player is the winner, thus one can notice that the winner is always on the left of the equation, and the loser is always on the right. In other words, if we present the winner's skill as s_{winner} and the loser's skill as s_{loser} , we can express the relations as follows:

$$s_{winner} - s_{loser} = (s_b - s_w)(-1)^{(1-r^{(k)})} \quad (87)$$

Thus, we can rewrite Equation 86 as follows:

$$\prod_{k=1}^N \left[I(r^{(k)} = 1) \Phi \left[\frac{s_{winner} - s_{loser}}{\sqrt{2}\beta} \right] + I(r^{(k)} = 0) \left(\Phi \left[\frac{s_{winner} - s_{loser}}{\sqrt{2}\beta} \right] \right) \right] \quad (88)$$

$$\prod_{k=1}^N \left[\Phi \left[\frac{s_{winner} - s_{loser}}{\sqrt{2}\beta} \right] \left(I(r^{(k)} = 1) + I(r^{(k)} = 0) \right) \right] \quad (89)$$

As the sum of the two indicator functions will always produce 1, thus, we can rewrite Equation 89 as follows:

$$\mathbb{P}\left[\{r^{(k)}\}_{k=1}^N \mid \{s_i\}_{i=1}^M\right] = \prod_{k=1}^N \left[\Phi\left[\frac{s_{winner} - s_{loser}}{\sqrt{2}\beta}\right] \right] \quad (90)$$

We can model the relation between s_{winner} and s_{loser} in terms of vectors. As we presented their relationship in Equation 87, it can also be expressed as a product of two vectors, namely s and $x^{(k)}$. More specifically, s will contain all the M players' skills. While $x^{(k)}$ will be a zero vector of magnitude M , where the winning player's skill will be substituted by 1 and the losing player's skill by -1 . For instance, if $M = 30$ (number of players), players 2 and 28 play a game, and we want to calculate the probability of 2 winning, then we would write it as $s = [0, 0, 1, \dots, -1, 0]^T$. This model, however, does not account for the relations between the players' colours, neither it accounts for *komi* points. As the player's standard deviation (β), is constant in the model, we can include it in the $x^{(k)}$ vector. Therefore, instead of 1 and -1 we can write $\frac{1}{\sqrt{2}\beta}$ and $-\frac{1}{\sqrt{2}\beta}$ respectively. Thus, using the new notation, we can rewrite Equation 90 as follows:

$$\mathbb{P}\left[\{r^{(k)}\}_{k=1}^N \mid \{s_i\}_{i=1}^M\right] = \prod_{k=1}^N \left[\Phi[s^T x^{(k)}] \right] \quad (91)$$

We can model the prior ($\mathbb{P}[\{s_i\}_{i=1}^M]$), in matrix notation as well - $\mathcal{N}(s, m_0, V_0)$. As all players' skills has been initialised to have mean of 0, thus m_0 will be a vector of 0s. Every player will has a variance of σ^2 , thus we can model V_0 as a diagonal matrix with σ^2 on the main diagonal.

$$V_0 = \begin{bmatrix} \sigma^2 & 0 & \dots & 0 \\ 0 & \sigma^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma^2 \end{bmatrix} \quad (92)$$

Substituting the likelihood, Equation 91, in Equation 85, and the prior for players' skills, we get:

$$\mathbb{P}\left[\{s_i\}_{i=1}^M \mid \{r^{(k)}\}_{k=1}^N\right] \propto \prod_{k=1}^N \left[\Phi[s^T x^{(k)}] \right] \mathcal{N}(s, m_0, V_0) \quad (93)$$

3 Approximate inference with Go game data

3.a

In order to interpret the graphs, I will firstly illustrate the process of expectation propagation with messages, passing through the graph, given in Figure 2 of the assignment sheet, which copy is presented at Figure 3. There, the flow of messages for only one game of Go are shown and are labelled in red and blue. Further, I will associate some of the messages with the different plots and explain what they represent.

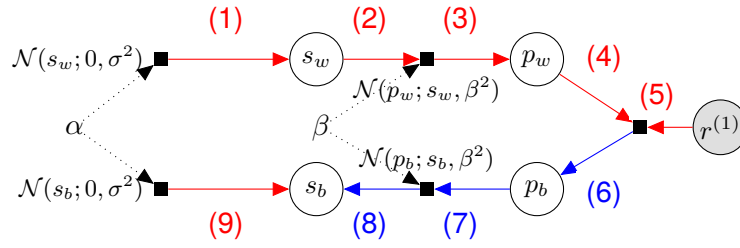


Figure 3: Factor graph of Gaussian player skill model for a single game. The variables $r^{(1)}$, s_b and s_w are as defined for Figure 1. The variables p_b and p_w are the performances of the black and white player respectively in the game. The red edges are the original messages, whereas the blue edges are the approximated ones for each individual factor locally.

In Figure 3, message (1) is only a Gaussian ($\mathcal{N}(s_w; 0, \sigma^2)$) from the factor on the left. Message (2) is a product of the all connected factors to the variable s_w , which in this case is only 1 - the Gaussian, therefore, it is just a copy of that Gaussian. In order to calculate message (3), we have to multiply the output from message (2) with the factor $\mathcal{N}(p_w; s_w, \beta^2)$ and integrate over s_w . This convolution could be estimated exactly [1, Equation 2.115] and produces another Gaussian distribution. Message (4) is a copy of message (3), as it represents the product of all incoming messages, which in this case is only the output from (3). Message (5) is a Bernoulli distribution. To compute message (6), we have to product messages (4) and (5), integrate over p_w and sum over $r^{(1)}$. If we consider the case, where black are the winners, i.e. $r^{(1)} = 1 \Leftrightarrow \mathbb{1}[p_b > p_w]$, then message (5) becomes 1, similarly it becomes 0, when $r^{(1)} = 0 \Leftrightarrow \mathbb{1}[p_w \geq p_b]$. From this, we can deduct that the regions. where p_w is winning will be multiplied by 0 [2]. Finally, when we integrate over p_w , we receive the true factor, showed in green on Figure 4.

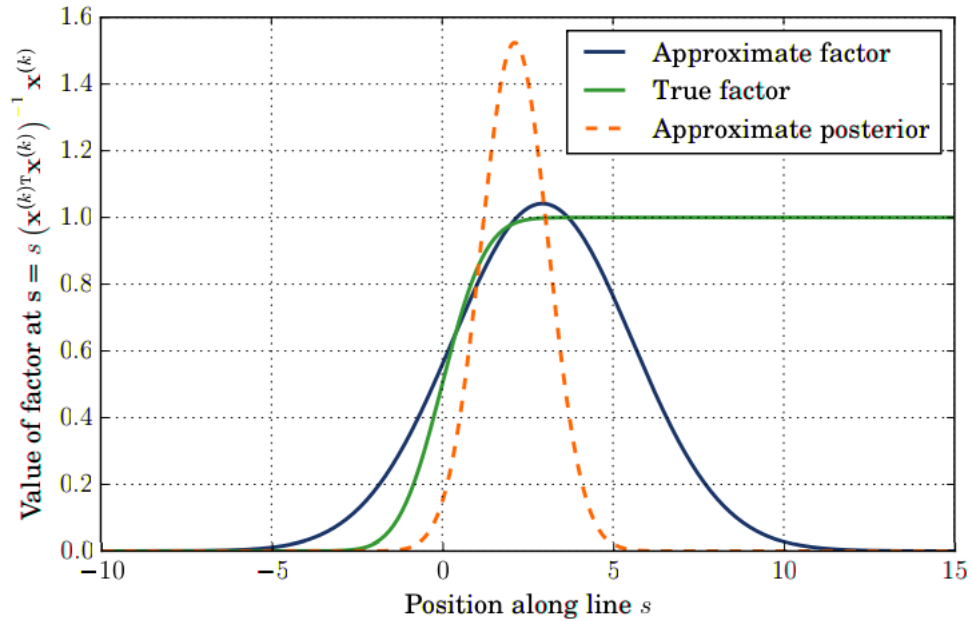


Figure 4: Values of corresponding approximate and true factors and scaled value of approximate posterior density along line $s = s|x^{(k)}|^{-1}x^{(k)}$ for a single factor in the diagonal covariance approximate densities, adopted from the assignment sheet.

This is the case, as for each value of p_b , a "cut" Gaussian (area being 0, due to $r^{(1)} = 0$) is being integrated, which results in the Cumulative Density Function of a Gaussian (shown in green on Figure 4) [2]. As the true factor has mean and variance of infinity (CDF), we cannot simply approximate it. As we want the approximation to be local, i.e. at each factor, we need to retrieve an approximate factor, which is in fact the message on the same edge (6), but in the opposite direction, from p_b to the factor [2]. This will be a Gaussian, passed from (9) – (8) – (7) – (6) in the same fashion as messages (1) – ... – (4), previously discussed. I will denote this message as $\text{context}(c_k)$ [2], which will help me outline the relationship between the green and blue plots. The true factor will be denoted as f_t . The factor approximation is denoted as f_a . The messages can be visualised on Figure 5.

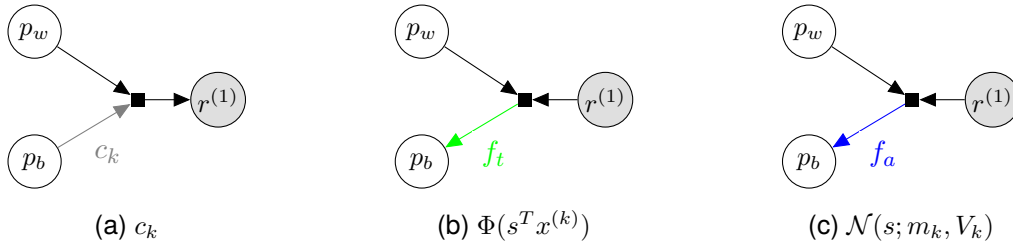


Figure 5: Message passing during EP, colour coded to match Figure 4.

What we are trying to achieve is to get from the CDF in Figure 5b to the Gaussian in Figure 5c. The approximation can be mathematically expressed as follows:

$$c_k f_a = \text{approx}(f_t c_k) \quad (94)$$

Multiplying the true factor(f_t) with context message(c_k) will give us the true marginal. However, their product will not be a Gaussian, but it will still have a finite mean and variance. Thus, we can simply create a Gaussian distribution with the same mean and variance (*approx* function), resulting in:

$$\mathcal{N}(x; \mathbb{E}[f_t c_k]; \text{Var}(f_t c_k)) \quad (95)$$

Finally, we can substitute this new distribution in Equation 94, in order to retrieve the final result for the blue factor approximation:

$$c_k f_a = \mathcal{N}(x; \mathbb{E}[f_t c_k]; \text{Var}(f_t c_k)) = c_k \mathcal{N}(s; m_k, V_k) \quad (96)$$

Now if we divide by the context message (c_k), we will receive the factor approximation, denoted as message (6) in Figure 3.

Continuing with the messages in Figure 3, now that we have that the message approximation (6) is a Gaussian, message (7) is just a copy of it, but it also differs from its original message, that was supposed to be a normal CDF. Message (8) is again a convolution of two Gaussians - the factor and the message, which results in a Gaussian. Also, please note that message (8) is also different from its original message.

Now we can now calculate the approximate posterior for the black player (s_b) - the orange plot on Figure 4. This can be done by multiplying the two Gaussians from message (8) and (9),

that will result in another Gaussian. This way, when another game with the same black player is played, his new skill posterior can be directly applied. Meaning that instead of a resulting in a CDF, it is now a Gaussian, having 2 parameters, same as the original s_b and s_w . This can be also thought of as the concept of conjugate priors - between different games we try to have identical distributions so that our model keeps consistent and does not grow in complexity. Figure 6 and 7 help to imagine intuitively how the skill of a player will change, depending on the other player. For instance, if the black player has more skill, than the white player, and wins, his skill should not be significantly updated, as it was expected that the white player would win. Similarly, the white player's skill is reduced, but not by much, and his certainty is growing (i.e. lower variance). This is illustrated on Figure 6.

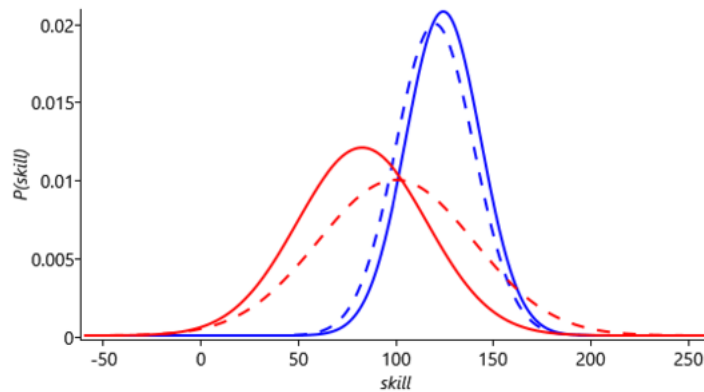


Figure 6: Black player wins, where black player is in blue and the white player is represented by the red curve. The priors are showed in dashed lines, whereas the posteriors are shown in solid lines. Adopted from [2].

If, however, the black player loses, his skill is reduced in comparison with the white player. The white player's skill grows above the black player one, and as a result it is significantly higher than its prior skill level. This is illustrated in Figure 7.

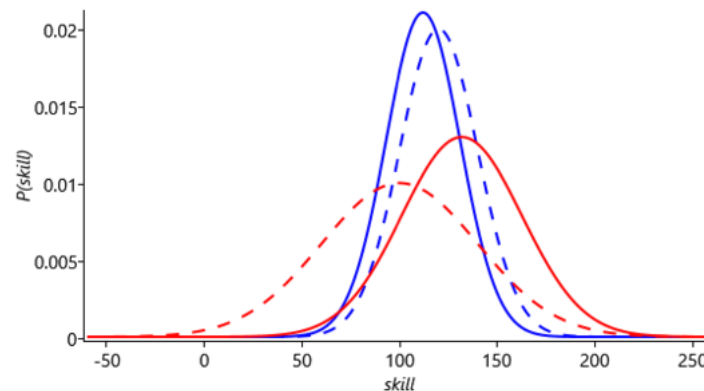


Figure 7: Black player loses, where black player is in blue and the white player is represented by the red curve. The priors are showed in dashed lines, whereas the posteriors are shown in solid lines. Adopted from [2].

These posterior marginals (solid lines in Figure 6 and 7) are an example of $p_a = \mathcal{N}(s; m; V)$ (dashed orange line in Figure 4).

3.b

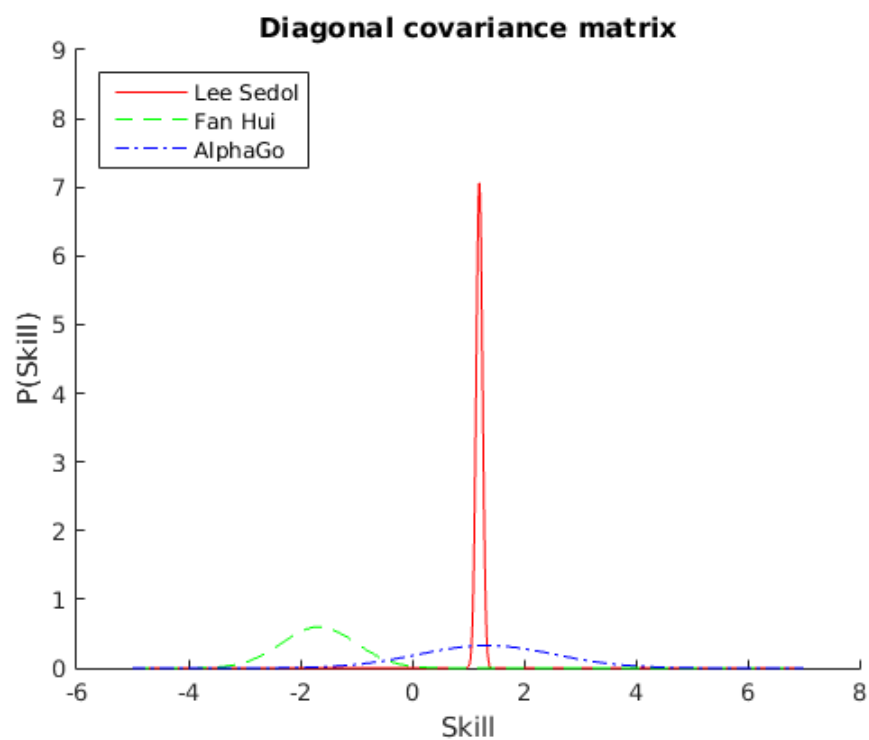


Figure 8: Approximate posterior marginal distributions for AlphaGo, Fan Hui, Lee Sedol using diagonal covariance approximation.

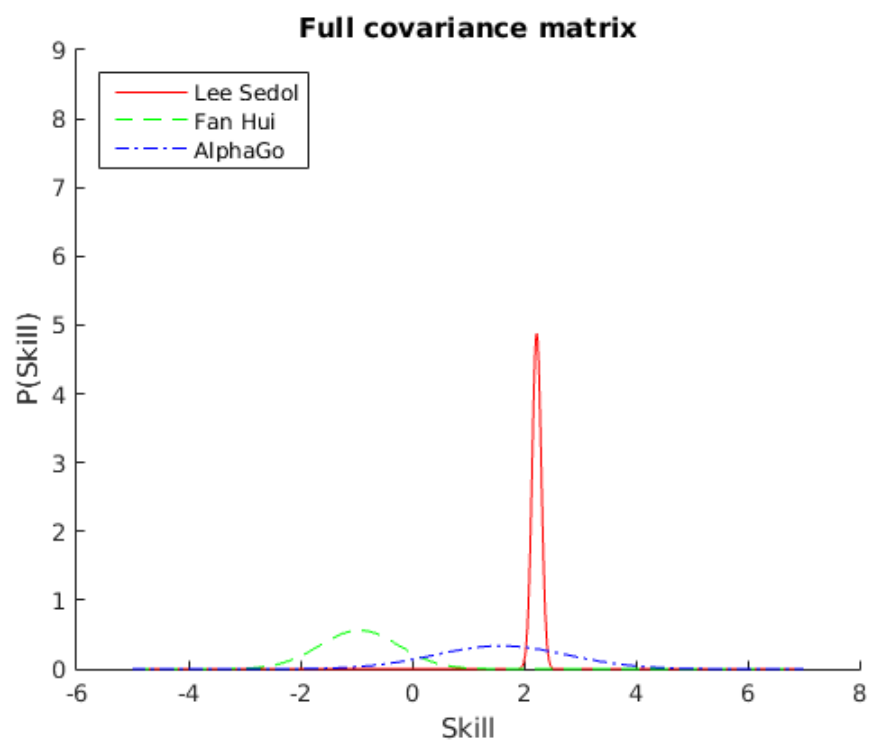


Figure 9: Approximate posterior marginal distributions for AlphaGo, Fan Hui, Lee Sedol using full covariance approximation.

Listing 1: Code for plotting approximate marginal posterior using full and diagonal approximations

```
1 function [] = task3b()
2 %TASK3B Plots the approximate Gaussian posterior of each player with ↵
   diagonal
3 %and full covariance matrices
4
5     load('go_player_skill_model/diag_covar.mat');
6
7     % set axis
8     x = -5:.0012:7;
9
10    figure(1);
11    hold on;
12    title('Diagonal covariance matrix');
13    plot(x, normpdf(x, approx_mean(lee_sedol_id), approx_covar(↵
       lee_sedol_id)^0.5), '-r');
14    plot(x, normpdf(x, approx_mean(fan_hui_id), approx_covar(fan_hui_id)↵
       ^0.5), '--g');
15    plot(x, normpdf(x, approx_mean(alpha_go_id), approx_covar(alpha_go_id)↵
       ^0.5), '-.b');
16    ylim([0 9]);
17    legend('Lee Sedol', 'Fan Hui', 'AlphaGo', 'Location', 'northwest');
18    xlabel('Skill');
19    ylabel('P(Skill)');
20    hold off;
21
22    load('go_player_skill_model/full_covar.mat');
23
24    figure(2);
25    hold on;
26    title('Full covariance matrix');
27    plot(x, normpdf(x, approx_mean(lee_sedol_id), approx_covar(↵
       lee_sedol_id,lee_sedol_id)^0.5), '-r');
28    plot(x, normpdf(x, approx_mean(fan_hui_id), approx_covar(fan_hui_id,↵
       fan_hui_id)^0.5), '--g');
29    plot(x, normpdf(x, approx_mean(alpha_go_id), approx_covar(alpha_go_id,↵
       alpha_go_id)^0.5), '-.b');
30    ylim([0 9]);
31    legend('Lee Sedol', 'Fan Hui', 'AlphaGo', 'Location', 'northwest');
32    xlabel('Skill');
33    ylabel('P(Skill)');
34    hold off;
35    drawnow;
36 end
```

Distributions clearly show that the skill of the two "real" players - Lee Sedol and Fan Hui is well defined, as their variance using both full and diagonal approximation is way smaller, than the variance of AlphaGo. This can be explained by the fact that the two players have played more games than AlphaGo has.

Furthermore, it is interesting to see the difference between using the diagonal and full approximations with respect to the variance. For example, Fan Hui's variance when using the full approximation is larger than when using diagonal approximation. Same goes for Lee Sedol. Whereas, AlphaGo's variance behaves differently - it is lower for the full covariance matrix. Logically, this can be explained by the fact that the full covariance matrix contains more information, i.e. the covariance with respect to the other 1048 players, whereas the diagonal approximation only involves the variance. While this is partially true, there are more factors that have to be taken into account. Firstly, the full covariance matrix doesn't mean that each player played against each other, even though, the entries are non-zero. The reason for this is the inverse of the covariance matrix. Secondly, the difference between the covariance of Fan Hui and AlphaGo is very small, it can be considered as insignificant.

Finally, one can notice that there is a shift of the mean for all 3 players. When using the diagonal approximation one can see that Lee Sedol's and AlphaGo's skill mean are quite close, even though, their uncertainty level is very different. When using the full covariance matrix, however, we can clearly see, that Lee Sedol's skill is better defined in terms of the graph presented on Figure 9. In addition, the skill mean of Lee Sedol is clearly above AlphaGo's one when using the full covariance matrix.

3.c

$$P(r = 1 | s_b = 1049, s_w = 26, approx = diag) = 0.52726 \quad (97)$$

$$P(r = 1 | s_b = 1049, s_w = 26, approx = full) = 0.36533 \quad (98)$$

Listing 2: Code for computing the probability of AlphaGo winning when playing black against Lee Sedol using both full and diagonal covariance matrices.

```

1 function [] = task3c()
2 %TASK3C Calculate the probability of AlphaGo winning, when playing as ↔
   black
3 %using both diagonal and full covariance matrices
4
5     format long
6
7     load('go_player_skill_model/diag_covar.mat');
8
9     % create your x vector to be a column vector with n_players size
10    x = zeros(n_players, 1);
11
```



```

12 % set Alpha Go with black
13 x(alpha_go_id,1) = 1 / sqrt(2*performance_var);
14
15 % set Lee Sedol with white
16 x(lee_sedol_id,1) = -1 / sqrt(2*performance_var);
17
18 numerator = approx_mean * x;
19
20 denominator_diag = sqrt((x' * diag(approx_covar) * x) + 1);
21
22 eq_diag = numerator / denominator_diag;
23
24 disp(['P(r=1|s_b=',num2str(alpha_go_id), ',s_w=',num2str(lee_sedol_id)↵
    ',approx=diag)=',num2str(normcdf(eq_diag))]);
25
26 load('go_player_skill_model/full_covar.mat');
27
28 numerator_full = approx_mean * x;
29
30 denominator_full = sqrt((x' * approx_covar * x) + 1);
31
32 eq_full = numerator_full / denominator_full;
33
34 disp(['P(r=1|s_b=',num2str(alpha_go_id), ',s_w=',num2str(lee_sedol_id)↵
    ',approx=full)=',num2str(normcdf(eq_full))]);
35
36 end

```

3.d

Considering the two results in Equation 97 and 98, as well as the two graphs in Figure 9 and 8, one could argue that the results are expected. Meaning that when using the diagonal covariance matrix, where the means of Lee Sedol and Alpha Go are quite close, AlphaGo has 52.726% of winning. When using the full covariance matrix, one could also observe that as the distributions are further apart, with Lee Sedol's PDF having advantage, it is reasonable to assume that AlphaGo has a lower chance of winning - 36.533%.

However, as previously mentioned in Section 3.b, the full covariance matrix is more informative, as it has the covariances between the rest of the players. In other words it takes into account the previously played games between players when computing the marginal posterior. Having said that, the accuracy of the estimated posterior might be hard to evaluate. The reason is that the skill of a player is something not actually observable, but can be regarded as hidden variable. Thus, it would take lots of trial and error in order to determine whether the approximation is correct or not. These experiments would have to be based on observations of game outcomes. This can be confirmed by the fact that on both Figure 4 and 10 the diagonal and full

covariance matrix approximate the true factor almost identically.

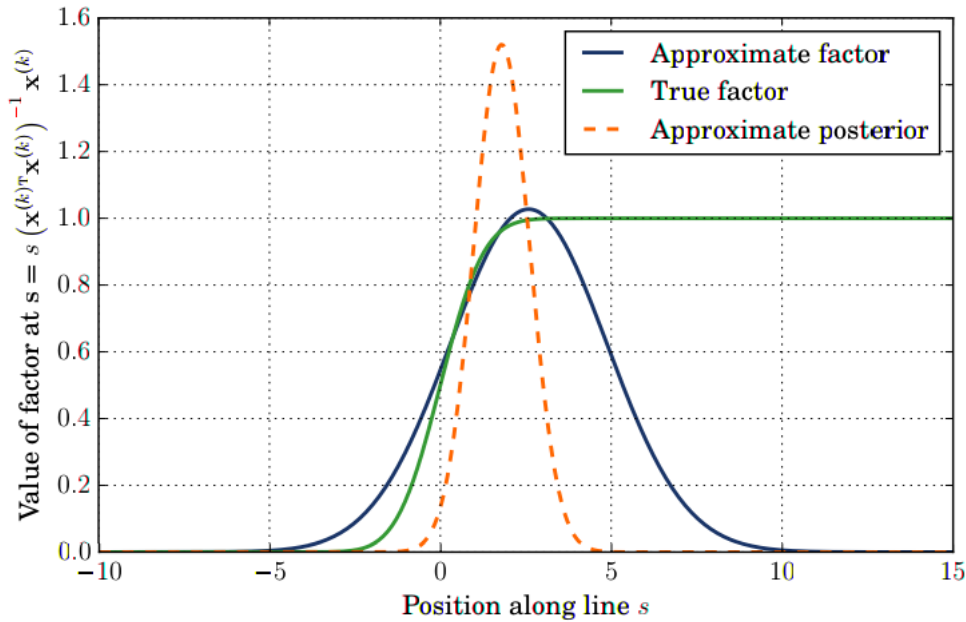


Figure 10: Values of corresponding approximate and true factors and scaled value of approximate posterior density along line $s = s|x^{(k)}|^{-1}x^{(k)}$ for a single factor in the full covariance approximate densities.

On the other hand, if we have to consider millions of Go players, then the full matrix computation will become very inefficient, as it is computationally expensive to inverse the full covariance matrix. Whereas, inverting the main diagonal of a matrix is essentially inverting a vector, which will take linear time. In fact, Schölkopf et al. [7] argue that adopting the full covariance approximation approach, an extra complexity of $\mathcal{O}(K^3N^3)$ is added. Furthermore, they show that when using the diagonal matrix, we lose the "explicit posterior coupling" [7].

4

4.a

As shown in Section 2.f, when computing the Cumulative Density Function, the model considers only the pure skill of the player. A possible extension of the model would be to add an additional constant that would account for the disadvantage of playing with the white stones. This extension, *komi*, however, needs to be weighted, based on the difference in the skills of the two players. For example, consider two games, in the first one s_w^1 plays s_b^2 , and the difference between them is 2000, and in the second game s_b^1 plays s_w^3 , where the skill difference is only 10. If a constant *komi* is added to the CDF calculation, then it might result in very different approximations. This weighted factor, by which the *komi* should be weighted might be hard to compute, as it would involve numerous observations of game outcomes. Mathematically this extension can be expressed as follows:

$$\Phi \left[\frac{s_b - s_w}{\sqrt{2}\beta} + \frac{K}{K_{factor}} \right] \quad (99)$$

A similar implementation was considered in [8], with the slight difference of accounting for handicapped games. In my opinion, accounting for the extension would shift the approximation of the CDF towards the skill of the white player, in order to compensate for the disadvantage of starting second. This will eventually result in more accurate calculations of the approximate posteriors for the skills of both players.

4.b

In the model so far, we have assumed that the player skill will be fixed, even though we are trying to approximate it, we do not allow it to change dynamically over time. Such situations might happen, when for example, a player has a "bad" day, when he is not performing up to his usual standards. In order to model this, we can extend the model to account for skill at any given time $t \in \{1..T\}$, such that, $p(s^t | s^{t-1}) = \mathcal{N}(skill_{change}; 0, s_{change})$. In other words, the skill changes through time, where that change is modelled with a zero-mean Gaussian. This means that the skill varies from game to game and on average it does not vary much, i.e. $\mu_{change} = 0$. Therefore, our skill posterior will be updated as follows [2]:

$$s^t = s^{t-1} + s_{change} = \mathcal{N}(s^t; \mu_{s^{t-e}}, \text{Var}(skill_{change})) \quad (100)$$

Therefore, the updated factor graph for 2 consecutive games between 2 paired players would look like so:

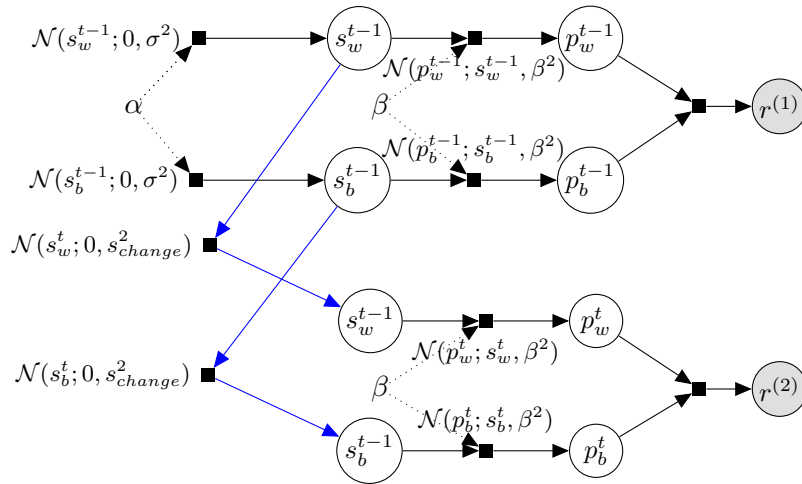


Figure 11: Factor graph of Gaussian player skill model for a two consecutive games between 2 paired players. The variables used are similar to the ones in Figure 3, with the difference of the time step $t \in \{1..T\}$. In addition the figure presents a model that accounts for the change of skill between games by introducing a two new factors ($\mathcal{N}(s_w^t; 0, s_{change}^2)$ and $\mathcal{N}(s_b^t; 0, s_{change}^2)$). They act as the new prior in the following game, introducing more uncertainty between games.

In [2], Winn and Bishop suggest that we firstly run expectation propagation in the game with outcome $r^{(1)}$. Then, the outgoing messages, from s_w^{t-1} and s_b^{t-1} to s_w^t and s_b^t , are updated

skill distributions that pass through the new Gaussian Factors. However, when convoluting these messages with the Gaussian factor (for example $\mathcal{N}(s_w^t; 0, s_{change}^2)$), essentially, we are introducing more uncertainty between games. This convolution is then used as the new prior for the next game. While introducing more uncertainty with every game may not be intuitive, it allows us to model the skill change of a player through time [2]. Essentially, this new model allows to reflect the possibility of a change in players' skill.

Figure 12 shows the improvement made by Winn and Bishop using this model over the previous one. The red line denotes the true skill change, the blue line shows the inferred skill change under the original model used in Section 2 and 3. The green curve shows the updated model's inference, supporting the change of skill over time.

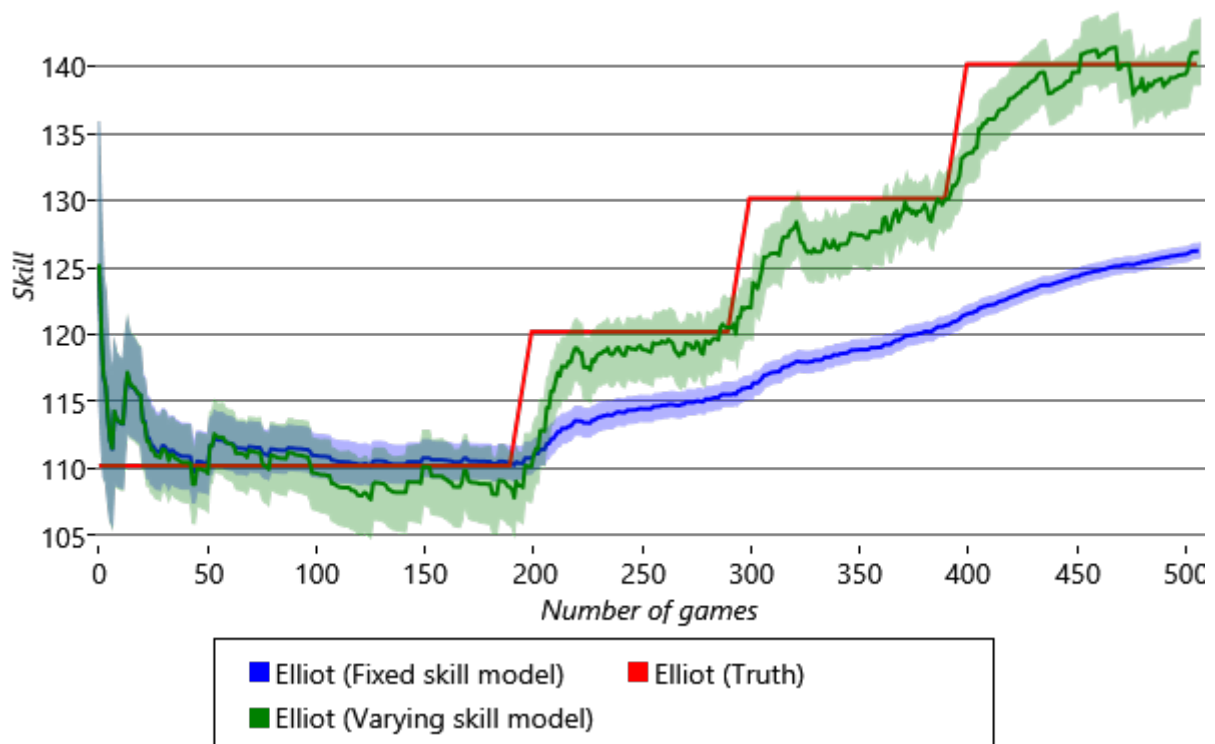


Figure 12: Figure adopted from [2]. Elliot is the player who skill is being modelled. Red line denotes the true skill change, blue line shows the inferred skill change under the original model (Figure 3), green curve shows the updated model's inference, supporting the change of skill over time.

This approach has several drawbacks - it depends on the order of the update, as well as it does not account for the very old games [3]. For instance, on Figure 12, if the first 200 games are played 2 years ago, then the model might not be relevant, as other factors could potentially influence players' skill. The second drawback is that inference is only propagated forward in time [3]. For example, if player A plays an unseen opponent B and wins, but this opponent turns out to be very good, then the skill of player A is not properly adjusted.

An alternative, suggested by Dangauthier et al. [3] is to use a method called TrueSkill Through Time (TTT). Their improvement can be partially applied to the aforementioned model. A significant change would be to track the game outcomes based on different years. For example,

$r^{(1)}$ and $r^{(2)}$ will become $r^{y(1)}$ and $r^{y(2)}$ to relate them to games in the same time period (years in this case). This is described as the Vanilla TrueSkill in [3]. In order to improve this Vanilla TrueSkill, we have to run the full EP algorithm until convergence for a certain time period (y). An important note is that the previous update of the result of a game must be removed before the new one is applied. Thus, the model is consistent, but less approximate [3].

4.c

In Section 3, the expectation propagation inference method is used, that allow intractable distributions to be approximated by Gaussians, it is considered to be the generalisation of loopy belief propagation. In Section 3.a, it is shown in detail how from a normal CDF, we can approximate a Gaussian distribution and use that instead, in order to have a conjugate prior for the next game - a Gaussian distribution.

One alternative for this model is to use Gibbs Sampling. When evaluating the integral, given in Equation 101, the first term ($\mathbb{P}[r^* = 1 \mid s_{b_*}, s_{w_*}]$) is relatively easy to compute, whereas the second term ($\mathbb{P}[s_{b_*}, s_{w_*} \mid \{r^{(k)}\}_{k=1}^N]$) could be very computationally expensive. Rather, we can sample from it, minimising this cost.

$$\mathbb{P}[r^* = 1 \mid \{r^{(k)}\}_{k=1}^N] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathbb{P}[r^* = 1 \mid s_{b_*}, s_{w_*}] \mathbb{P}[s_{b_*}, s_{w_*} \mid \{r^{(k)}\}_{k=1}^N] ds_{b_*} ds_{w_*} \quad (101)$$

While Gibbs sampling takes a lot of time to give an exact result [6], but could be more computationally efficient for this domain. In addition, MCMC algorithms (Gibbs sampling) is general, but potentially could be expensive, while EP utilises the simplicity in a model [4].

A clear disadvantage of EP is that it might not converge [1]. However, this is not the case for our model, thus it will not be discussed in detail. Another possible disadvantage of the EP might come from the fact of how it is estimating the Kullback-Leibler (KL) divergence. As KL is not a symmetric ($KL(p||q) \neq KL(q||p)$), estimating $KL(p||q)$ could lead to poor estimations [1, Figure 10.3]. A suitable alternative, both for convergence (even not applicable) and minimising multi-modal distributions would be to use Variational Message Passing (VMP). It rather computes $KL(q||p)$ and is guaranteed to converge (even if not applicable for this domain) [1, 6]. Thus, it would mean that the accuracy of the approximations using VMP would be higher than when using EP. Furthermore, while VMP could require more iterations, than EP, these iterations are cheaper [5, 9]. In addition, VMP gives one the flexibility to specify very broad class of conjugate exponential approximating family [9], if needed. It must be noted, however, that there exists an optimisation for EP cost function, that would guarantee convergence[1].

References

- [1] Christopher M Bishop. Pattern recognition. *Machine Learning*, 2006.
- [2] Christopher M Bishop. Model-based machine learning. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 371(1984):20120222, 2013.
- [3] Pierre Dangauthier, Ralf Herbrich, Tom Minka, and Thore Graepel. Trueskill through time: Revisiting the history of chess. In *Advances in Neural Information Processing Systems*, pages 337–344, 2007.
- [4] Thomas Minka. Expectation propagation.
- [5] Thomas P Minka. Expectation propagation for approximate bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 362–369. Morgan Kaufmann Publishers Inc., 2001.
- [6] Microsoft Research. Working with different inference algorithms.
- [7] Bernhard Schölkopf, John Platt, and Thomas Hofmann. *Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference*, volume 19. Mit Press, 2007.
- [8] Marius Stanescu. *Rating systems with multiple factors*. PhD thesis, Citeseer, 2011.
- [9] John M Winn and Christopher M Bishop. Variational message passing. In *Journal of Machine Learning Research*, pages 661–694, 2005.