UNIVERSITY OF CAMBRIDGE

MPhil in Machine Learning, Speech and Language Technology
2015-2016

MODULE COURSEWORK FEEDBACK

Student Name: Riashat Islam

Module Title: Machine Learning

CRSiD: ri258

Module Code: 4F13

College: St John's

Coursework Number: 2

I confirm that this piece of work is my own unaided effort and conforms with the Department of Engineering guidelines on plagiarism

I declare the word count for this piece is: 2500

Student's Signature: Riashat Islam

Date Marked:

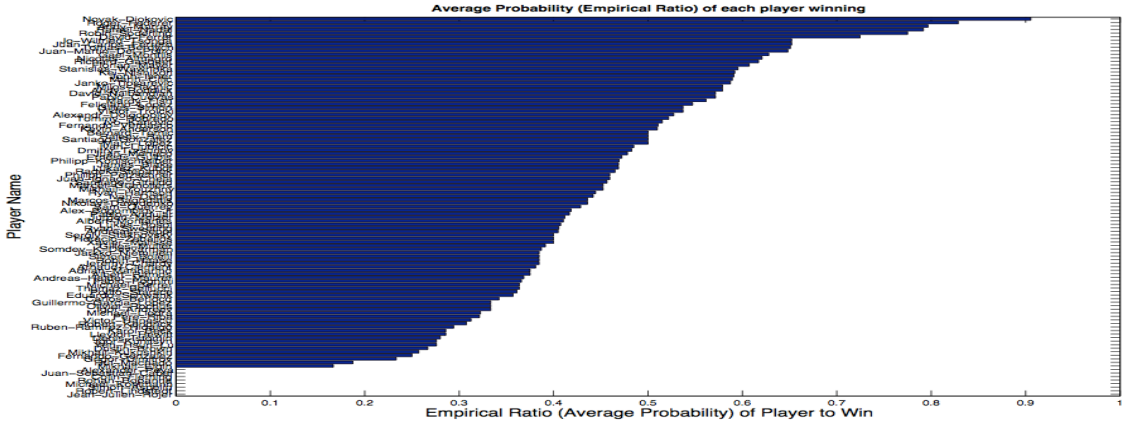
Marker's Name:

This piece of work has been completed to a standard which is *(Please give mark as appropriate)*:

Marker's Comments:

Question a

A ranking for each player based on the empirical ratio of wins over number of games played in 2011 is plotted in the bar graph below. The ratio is simply obtained by counting the number of wins/number of games played.



Below are few lines of code that were used, in addition to the bar plot code given:

```
for i = 1:107
  count_won(i) = sum(G(:,1)==i);                    count_lost(i) = sum(G(:,2)==i);
ratio(i) = count_won(i)/ ( count_won(i) + count_lost(i) );
end                                                 [kk, ii] = sort(ratio, 'descend');
```

This is not a good method to estimate players skills because:

It doesn't take into account whether the player played too many or too few games. A player with less skill might have played a top ranked player in each competition and got eliminated early, which would mean he has played few games than a top ranked player who went to finals of almost every tournament. The ranking does not also take into account whether the player played (and won) too weak or too strong opponents. If a top ranked player played against too many lower ranked player, then he is more likely to win (resulting in higher ranking); there is no measure of who the opponent in the game was.

Question b

The mean to sample from the conditional distribution is given by the following, with 100 iterations of gibbs sampling:

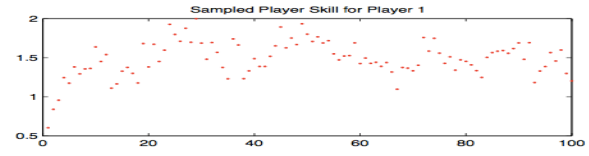
```
m(g) = transpose(t)*((g==G(:,1)) - (g==G(:,2)));
```

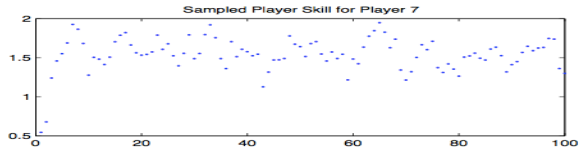
The precision matrices (iS matrix) was computed by:

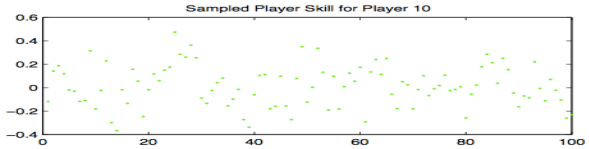
```
  for i = 1:M
    for j = 1:i
      if i==j
        iS(i,j) = sum( (i==G(:,1)) + (i==G(:,2)) );
      else
        iS(i,j) = -sum((i==G(:,1)).*(j==G(:,2))+(i==G(:,2)).*(j==G(:,1)));
        iS(j,i) = iS(i,j);
      end                    end                end
```

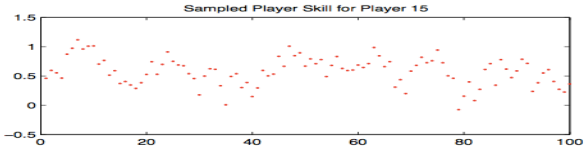
Question c

Below, the sampled player skills against number of iterations for Gibbs sampling is being plotted.



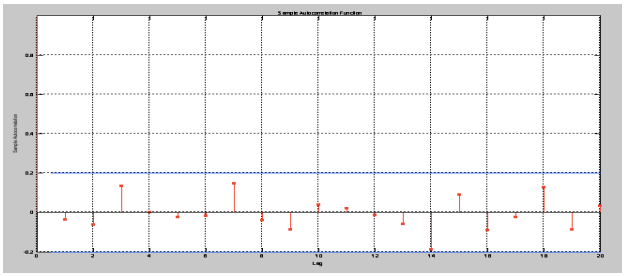






From the plots above, it can be seen that the sampled values for each player usually lies close to the mean value. The relatively close values of the samples shows that Gibbs sampling shows that Gibbs sampling is eventually drawing from a stationary distribution due to convergence guarantees, and hence it can move around the posterior distribution. Gibbs sampling is reducing the task of sampling from the complex posterior joint distribution to sampling from a univariate conditional distribution. Gibbs sampling usually leads to dependent samples from the joint distribution of player skills. However, we made the samples to be roughly 10 units apart to ensure that the samples are independent of each other. Such an approximate length of units apart would draw independent samples from the joint distribution. Therefore, with subsampling interval of 10, we can reach a stationary distribution for gibbs sampling. The code snippet used to get 100 samples:

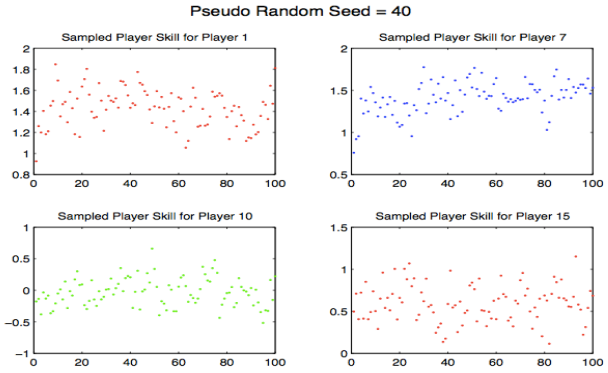
```
for iters = 1:100  ……..……
P_1(iters)=w(1);          P_7(iters)=w(5);          P_10(iters)=w(10);                P_15(iters)=w(15);
P_100(iters)=w(100);                    P_107(iters) = w(107);                          ……..end
```

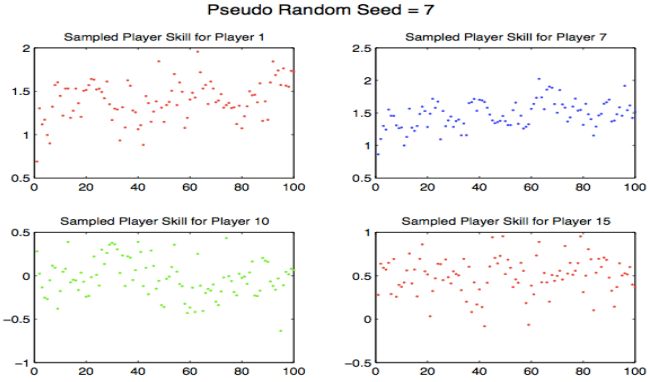


We use the matlab autocorrelation function to choose the length interval for subsampling, in order to ensure that the draws are uncorrelated and independent.
As per the plot below, we use a length interval of 15.

Question d

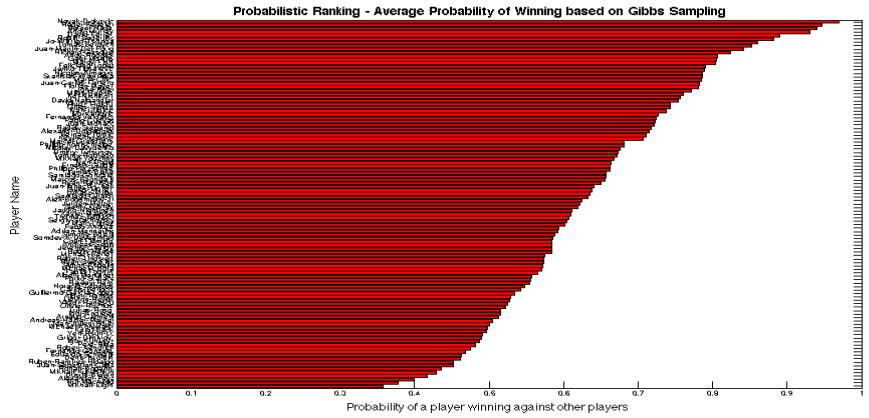
Convergence of a Markov chain means that after a lot of iterations, we would reach a stationary distribution with a mean and convariance value for the distribution. In sampling from the posterior distribution of player skills, it does seem like we reach convergence for Gibbs sampling. With 100 iterations of Gibbs sampler, since the distributions reach an approximately constant mean value, it is showing that Gibbs sampling is eventually generating dependent samples (without subsampling) from the joint distribution of p(**w**|**t,y**)
Below, we show results for different pseudo random seeds. We use a seed of 40 and 7, and for both, the same stationary distribution is reached, with similar fluctuations around mean values, showing that the Gibbs sampling converges.





Question e

We are using 1000 iterations of Gibbs sampling to ensure convergence, using a subsampling interval of 10 (ie, taking the 10th sample of skills every time) to get 100 samples of skills in total. This is mainly because drawing large number of samples (ensuring independence by 10 unit length apart) would ensure convergence of gibbs sampling to a stationary distribution. Because Gibbs sampling may have strong correlations between consecutive samples, we used subsampling of keeping only every 10th sampling from the 1000 iterations of the Gibbs sampler. This will ensure less dependence.



<u>Noting some of the ranking values showing the top 4 players and the probability of them winning based on the sampled skills of all the players:
Showing ranking of first 4 players:</u>

Novak Djokovic	0.9698
Roger Federer	0.9471
Rafael Nadal	0.9394
Andy Murray	0.9309

Using the 100 independent samples of player skills, we then take the mean value for each of the 100 samples for individual 107 players, and then use that to compute the average probability of each player winning against any other chosen player. To compute the ranking, we use the drawn samples from Gibbs to compute Φ (y(w1 - w2)) for each player over all the 100 samples. Using that, we then take the Monte Carlo approximation to find the expected values of the skills w1,w2 etc for each player. With the expected skills, we then sort the skills and find the probabilistic ranking based on the expected values of the skills. The code is given below:

```
for P1=1:M
  for P2=1:M
     for l = 1:size(independent_samples,2)
     prob(P1, l) = normcdf(independent_samples(P1,l) - independent_samples(P2,l));
     end                    end                    end
mean_prob= mean(prob,2);                    [kk, ii] = sort(mean_prob, 'descend');
```

Discussion and Comparison of Results:

The gibbs sampling approach, based on computing posterior distribution, computes the ranking based on the skills of players, and whom they played against. If a highly skillful player defeats a player with lowest skills, then it would not affect his ranking significantly, compared to defeating a similarly highly skilled player. The gibbs sampling approach is based on assigning credits to the players depending on the skills of the opponents they faced, rather than simply counting the ratios which does not take into account that a less skillful player will be more easy to beat.

Using gibbs sampling to compute a probabilistic ranking – we find that an average probability of winning is assigned to each player even though those players may have lost all the matches (as suggested by the plot using ratios). The plot using ratios takes into account only the number of wins, and hence if a player lost all matches, a zero probability was assigned to those players for ranking. Using a posterior distribution over the skills, and computing average probability, the ranking system is now based on the likely skills of the players, and given the skills, how much likely each player is expected to win [ie, the normcdf function in matlab is used for computing p(y|w1,w2) as in lecture notes, showing that even a small probability is assigned to least skillful players.

Similarly, for the players who are highly skilled, a higher probability is assigned to those players using Gibbs sampling, compared to ranking simply based on empirical ratio of wins. The ranking with gibbs sampling shows that the higher ranked players have a higher probability of winning (value larger than the plot using ratios).

Question f

Table 1

Probability of how likely a player is expected to win when playing against other player (top 4 player ranking) (Row -> Winners, Column -> Loser				
	Novak Djokovic	Rafael Nadal	Roger Federer	Andy Murray
Novak Djokovic	-	0.6464	0.6213	0.6871
Rafael Nadal	0.3536	-	0.4728	0.5465
Roger Federer	0.3787	0.5272	-	0.5722
Andy Murray	0.3129	0.4535	0.4278	-

The probability of the top 4 ATP players winning against each other is calculated by taking Φ (w(i)-w(j)) for each players over the 100 samples and then finding the Monte Carlo approximation of the samples to get the expected probability of each player winning against another. The table shows the expected probability of winning E[p(y=1)] = Φ (w(i)-w(j)), and a lower probability shows the player is more likely to lose and is simply (1 - E[p(y=1)]). The values in table above further justifies that Φ (w(i)-w(j)) is a Gaussian which we have approximated using Gibbs sampling, even though the original posterior distribution was complex.

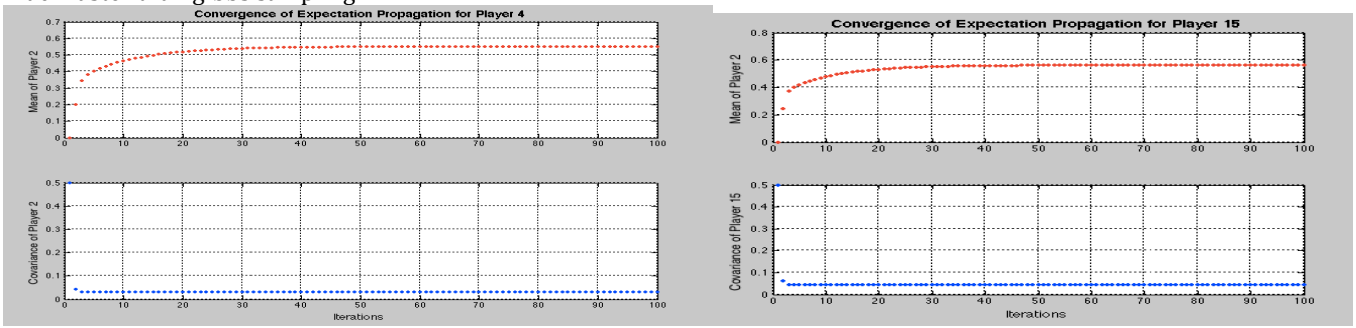
From the table above: value of 0.6464 (1,2) shows that Djokovic is more likely to beat Federer since he has a higher skill, and therefore higher probability of winning. Similarly, value of 0.3129(4,1) shows that Murray is less likely to beat Djokovic since he is a less skillful player compared to Djokovic, and therefore lower probability of winning is assigned to Murray. The table shows probabilities assigned to each player of winning against the other, based on the posterior distribution of the skills that were assigned to each player given the rankings 2011 data. The following code is used:

```
P1 = [16,1,5,11];                    player_skills = independent_samples(P1, :);
for q = 1:4
  for w = 1:4
    X(q,w) = mean(normcdf( player_skills(q,:) - player_skills(w,:) ));        end        end
```

Expectation Propagation

Question g

For expectation propagation, in order to judge convergence, we look at the values of mean and precision of the marginal skills p(w). In EP that is used to compute approximate posterior distributions using moment matching to find best approximation, EP approximates the belief states by only retaining expectations such as the mean and variance, and iterates until these expectations are consistent throughout the network. When a steady value of mean and precision is reached, it means the marginal skill contains updated information, with no further improvements in message updates as messages are passed along the tree. The constant values that are reached for the means and variances shows that EP has converged to a fixed point. This further shows that the marginal distribution of each of the player skills reaches an approximately Gaussian distribution with constant mean and covariance. We use the mean and precision as the objective function, and compare whether a steady value is reached as iteration progresses for multiple players. We used 100 iterations to judge convergence of EP, noting that EP runs much faster than gibbs sampling.



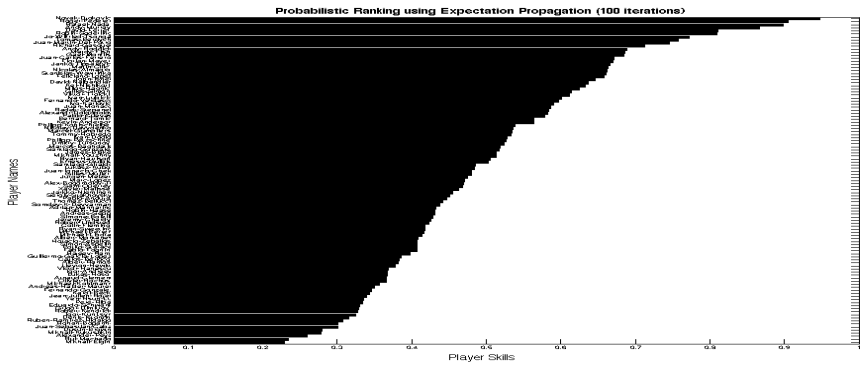
The above plot shows that EP converges much quickly to a point estimate having constant mean and precision values, compared to Gibbs sampling. The following lines of code were used to see convergence values:

```
M1 = zeros(M, no_iterations);                    P2 = zeros(M, no_iterations);
for iter=1:100……..
        M1(:,iter)=Ms; %Mean matrix per iteration
        P2(:,iter)=1./Ps; %Covariance matrix per iteration
end
%and then plot M1 and P2 for the players (example of player 4 shown below)
plot(M1(4,:),'r.');                    plot(P2(4,:),'b.');
```

Question h

The following code was used to compute the probability that each player will win against the other. Given that in EP, we obtain converted messages in terms of means and precisions, to get the posterior marginals p(w), we can then use that to compute p(y=1) which does have a closed form given that we know the marginals p(w) for each player, that are approximated to be Gaussian distributions. The following code were used to compute the probabilistic ranking, based on the converged mean and precision (converted to variances) values to compute Φ (y(υ1-υ2)/ sqrt(1 + σ_1^2 + σ_2^2)) as shown below

```
Mu = V1(:, iter);                    V = V2(:, iter);
rank = zeros(M,1);
for i= 1:M
  for j=1:M
    if(not(i==j))
      rank(i) = rank(i) + normcdf( (Mu(i)-Mu(j)) / sqrt(1 + V(i)  + V(j)) ) ;        end    end                end
avg_rank = rank ./(M-1);          %to find the mean of Φ ( y(υ1-υ2)/ sqrt(1 + σ1² + σ2²) )
[kk, ii] = sort(avg_rank, 'descend');
```



Noting some of the ranking values showing the top 4 players and the probability of them winning based on the posterior marginal distribution of skills of players

Showing ranking of first 4 players:

Novak Djokovic	0.9481
Roger Federer	0.9053
Rafael Nadal	0.8986
Andy Murray	0.660

Comparison of EP Ranking with Gibbs Ranking and Ratio Ranking

Using EP, we can get the skill marginal directly, which is an approximation to the true distribution (approximate inference). For the ranking based on EP, we are using the converged messages (mean and precision) and using that directly to compute the ranking, which is likely to give more accurate results. Comparatively, using Gibbs sampling, instead of computing the marginal, we are getting samples of the skills from the joint posterior distribution, and then using the samples to compute an exact Gaussian distribution that approximates the true distribution. Since Gibbs sampling requires lots of samples, and convergence to a stationary distribution is not guaranteed and easily observable, the ranking based on Gibbs sampling may not be based on converged stationary distribution. Hence EP is more reliable than Gibbs sampling for the ranking system, due to convergence guarantees and explicitly finding approximate values of player skills since it is a deterministic approximation.

Differences from the ranking system plots:

Based on arguments above, and that EP converges much faster than Gibbs sampling to find an approximate distribution, we conclude that EP based ranking is more reliable than Gibbs sampling or ratio based rankings. As shown in the ranking plot above, EP ranking is more spread out between players highly ranked and lowly ranked. Player with the least skills are assigned very low probability of winning, value being much smaller than assigned by Gibbs sampling even though both methods are based on skills of the players. Also, the values between probability of winning has larger difference between each player, showing that EP can form ranking with higher confidence, compared to Gibbs sampling where the probabilities of winning have very small.

Question i

Table 2

Probability of how likely a player is expected to win when playing against other player (top 4 player ranking) (Row -> Winners, Column -> Loser				
	Novak Djokovic	Rafael Nadal	Roger Federer	Andy Murray
Novak Djokovic		0.6554	0.6380	0.7198
Rafael Nadal	0.3446		0.4816	0.5731
Roger Federer	0.3620	0.5184		0.5909
Andy Murray	0.2802	0.4059	0.4245	

Comment on Differences of Results:
Comparing table 1 and table 2, some of the values in table 2 are higher than table 1, showing that EP computes probabilities of winning with higher confidence (based on convergence issues discussed above). For example, Djokovic beats Federer and Murray (0.6380 and 0.7198) with higher probability in EP, compared to in Gibbs (0.6213, 0.6871). Similarly, Murray is more likely to get deferred by Djokovi and Nadal (0.2802 and 0.4059) compared to Gibbs that assigns a higher probability to Murray comparatively (0.3129 and 0.4535). Table shows that EP assigns higher probabilities to players more likely to win based on higher skills, and also assigns lower probabilities to players more likely not to win (considering p(y=1) winning case for all the values, so lower probability means less chances of winning).
We use similar code as was shown above to plot the table of winning probabilities for the first 4 players

Question j

Using Gibbs sampling to find p(w1 > w2), we are simply using the samples of the skills, and using that to compute this probability. Given that we have 100 independent samples of skills, for each player, we are counting whether the sampled skill of one player is higher than all other players and take the average to get the probability. The probability of whether a players skill is higher than other players, considering first 4 players in ATP is then given by:

```
for q = 1:4                    for w = 1:4
higher_skills(q,w) = mean(player_skills(q,:)>player_skills(w,:)); %player_skill is the 100 independent samples drawn using Gibbs sampling                    end                    end
```

Table 3

Probability of player having higher skill compared to the other player (top 4 player ranking) (Row -> Higher skills, Column -> Lower skills				
	Novak Djokovic	Rafael Nadal	Roger Federer	Andy Murray
Novak Djokovic		0.6831	0.6947	0.7471
Rafael Nadal	0.3634		0.5520	0.6145
Roger Federer	0.3529	0.5278		0.6041
Andy Murray	0.3056	0.4770	0.4906	

Again, for the EP ranking, we want to find for example, p(w_Djokovic > w_Nadal), ie, finding the probability of a player having higher skill compared to each other player. Therefore, to find p(w1 >w2) we can use the following derivation:
p(w1>w2) = p(w1-w2 >0) = p(t – s > 0) and considering symmetry even of approximate Gaussian distributions,
p(w1 > w2) = 1 – normcdf (0, mean_player1 – mean_player2, variance_p1 + variance_p2) which means that if z=w1-w2, then we are taking the integral of z from 0 to inifinity.
The explanation for that is: since EP gives the parameters of the posterior marginals p(w) (approximate Gaussians), then we are considering that p(w1,w2) approximately being equal to p(w1) p(w2) . That is, given the posterior marginals, we are approximating the joint posterior and using that along with Gaussian distribution symmetry to find the probability of p(w1 > w2). The following lines of code were used:

```
mean_ep = [ Mu(16), Mu(1), Mu(5), Mu(11)];    var_ep = [ V(16), V(1), V(5), V(11) ];    higher_skills = zeros(4,4);
for z = 1:4                    for c = 1:4
    higher_skills(z,c) = 1 - normcdf(0, mean_ep(z)-mean_ep(c), sqrt(var_ep(z)+var_ep(c)));    end    end
```

Table 4

Probability of player having higher skill compared to the other player (top 4 player ranking) (Row -> Higher skills, Column -> Lower skills				
	Novak Djokovic	Rafael Nadal	Roger Federer	Andy Murray
Novak Djokovic		0.9398	0.9089	0.9853
Rafael Nadal	0.0602		0.4271	0.7665
Roger Federer	0.0911	0.5729		0.8108
Andy Murray	0.0147	0.2335	0.1892	

Comparing tables for Gibbs sampling (Table 1 and 3)
Table 3 shows which player has a higher skill compared to the other players. For example, a value of 0.6831 (1,2) shows that Djokovic has a higher probability of having better skills compared to Nadal, which is also supported by Table 1, showing that Djokovic is more likely to beat Nadal (0.6464). Similarly, Murray has a much lower probability of having a higher skill than Djokovic (0.3056) showing that Murray is much less skilled than Djokovic, which is also supported by the value of 0.3129 in table 1 showing that Murray has a lower probability of winning.

Comparing tables for Gibbs sampling (Table 2 and 4)
Table 4 shows high probabilities for computing the whether a player has a higher skill or not. For example, a value of 0.9398 (1,2) shows tha Djokovic has a much higher probability of having better skills than Nadal. However, not a very large probability is assigned to Djokovic for winning based on table 2 (0.6554) even though based on table 4, it was expected that Djokovic would beat Nadal with higher probability. Similar reasoning goes for Andy Murray showing that probability of Djokovic having higher skills than Murray is 0.9853 which is also supported by the relatively higher value of 0.7198 of Djokovic winning against Murray. Table 4 therefore shows that the higher the relative probability of one player having higher skills than others, the more likely he is to win against that player (with higher probability) – values of 0.4271 and 0.7665 with corresponding winning probabilities of 0.4816 and 0.5731
The table therefore correspond to each other showing that the probability of a player with higher skill than another player in turn also shows that the player is more likely to win against that player, having a higher probability of winning.