Improving Duckworth-Lewis: Statistical Methods for Resetting Score Targets in Limited-Overs Cricket

Matthew Knowles

University of York

mk1320@york.ac.uk

May 27, 2022

Plan for the Talk

1. Background
2. Neural Networks in Cricket
3. Resetting Score Targets
4. Conclusions

Cricket

- 2 teams of 11 players. One team bats, the other fields.

Cricket

- 2 teams of 11 players. One team bats, the other fields.
- Score measured in runs. Aim: Score as many as you can before losing 10 wickets.

Cricket

- 2 teams of 11 players. One team bats, the other fields.
- Score measured in runs. Aim: Score as many as you can before losing 10 wickets.
- Focus in this project is limited overs cricket. Games last 50 overs, which takes about 3 hours.

Setting the scene: The need for score resetting

- Cricket is very sensitive to external factors, such as rain and daylight.

Setting the scene: The need for score resetting

- Cricket is very sensitive to external factors, such as rain and daylight.
- If it gets too dark, the ball becomes very hard to see and so the game is stopped.

Setting the scene: The need for score resetting

- Cricket is very sensitive to external factors, such as rain and daylight.
- If it gets too dark, the ball becomes very hard to see and so the game is stopped.
- Similarly, if it rains, the game is stopped due to the adverse affect this has on the pitch.

A motivating example

- To illustrate the issue, consider the following example.

A motivating example

- To illustrate the issue, consider the following example.

Example

Team A scores 320 runs in their 50 overs, losing 8 wickets in the process. While team B is batting, it begins to rain, and the umpires call the game off with team B on 118-2 from 34 overs. After the rain stops, there is only time for 6 overs of play.

A motivating example

- To illustrate the issue, consider the following example.

Example

Team A scores 320 runs in their 50 overs, losing 8 wickets in the process. While team B is batting, it begins to rain, and the umpires call the game off with team B on 118-2 from 34 overs. After the rain stops, there is only time for 6 overs of play.

- Clearly, at this point it is unfair to expect team B to chase down 222 runs in 6 overs instead of the 16 they should have had. So for this reason, score target adjustment is needed to keep the game fair, despite the loss of time.

Duckworth, Lewis

- Statisticians Frank Duckworth and Tony Lewis set about a way to reduce score targets appropriately to overcome challenges like the one in the last example.

Duckworth, Lewis

- Statisticians Frank Duckworth and Tony Lewis set about a way to reduce score targets appropriately to overcome challenges like the one in the last example.
- They introduce the following formula

$$Z(u, w) = Z_0(w)(1e^{-b(w)u})$$

Which gives the runs scored with u overs remaining and w wickets lost. Note that the actual value of Z_0 and b, the decay constant are not given due to commercial agreements.

Duckworth, Lewis

- Statisticians Frank Duckworth and Tony Lewis set about a way to reduce score targets appropriately to overcome challenges like the one in the last example.
- They introduce the following formula

$$Z(u,w) = Z_0(w)(1e^{-b(w)u})$$

Which gives the runs scored with u overs remaining and w wickets lost. Note that the actual value of Z_0 and b, the decay constant are not given due to commercial agreements.

- Consider the special case at the start of an N-over innings. I.e $u = N$ and $w = 0$.

$$Z(N,0) = Z_0(1 - e^{-bN}).$$

Duckworth and Lewis (Contd.)

- We incorperate the above into a ratio that forms the basis of the Duckworth-Lewis method.

$$P(u, w) = \frac{Z(u, w)}{Z(N, 0)}.$$

Duckworth and Lewis (Contd.)

- We incorperate the above into a ratio that forms the basis of the Duckworth-Lewis method.

$$P(u, w) = \frac{Z(u, w)}{Z(N, 0)}.$$

- Computing all values for P with $1 \leq u \leq 50$ and $0 \leq w \leq 10$ gives a table of results. It is from this table that the revised score target is calculated.

Stern

- With the introduction of an even shorter form of the game, "Twenty20", consisting of only 20 overs instead of 50. The method needed to be updated.

Stern

- With the introduction of an even shorter form of the game, "Twenty20", consisting of only 20 overs instead of 50. The method needed to be updated.
- Steven Stern introduced updates in 2015 to account for the new increase in game scores.

Problems with DLS

- DLS has a tendancy to reset score targets that are unrealistic, and as such ruin the competetive nature of a match

Problems with DLS

- DLS has a tendancy to reset score targets that are unrealistic, and as such ruin the competetive nature of a match
- DLS does not account for fielding restrictions that are in place at different pooints in the game

Problems with DLS

- DLS has a tendancy to reset score targets that are unrealistic, and as such ruin the competetive nature of a match
- DLS does not account for fielding restrictions that are in place at different pooints in the game
- It is not easy to understand, especially for players and officials without any mathematical education

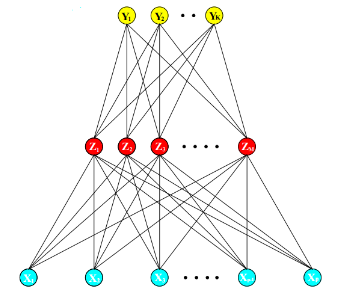
Structure of Neural Networks

- Neural Networks are a mathematical model based on neurons in a brain

Structure of Neural Networks

- Neural Networks are a mathematical model based on neurons in a brain
- To artificially create such a model, we use a system of nodes and synapses

Example



Mathematically Representing Neural Networks

Mathematically Representing Neural Networks

- Layers of nodes (perceptrons). k "hidden" layers sandwiched by an input layer and an output layer.

Mathematically Representing Neural Networks

- Layers of nodes (perceptrons). k “hidden” layers sandwiched by an input layer and an output layer.
- Each node in layer j connects to each node in layer $j + 1$. Connection is defined by a weight and a bias.

Mathematically Representing Neural Networks

- Layers of nodes (perceptrons). k "hidden" layers sandwiched by an input layer and an output layer.
- Each node in layer j connects to each node in layer $j + 1$. Connection is defined by a weight and a bias.
- Let the layer k_i have m nodes, and the layer $k_i + 1$ have n nodes. Then the weights between the layers are given by the matrix

Mathematically Representing Neural Networks

- Layers of nodes (perceptrons). k "hidden" layers sandwiched by an input layer and an output layer.
- Each node in layer j connects to each node in layer $j+1$. Connection is defined by a weight and a bias.
- Let the layer k_i have m nodes, and the layer $k_i + 1$ have n nodes. Then the weights between the layers are given by the matrix

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,m} \\ w_{2,1} & w_{2,2} & \cdots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n,1} & w_{n,2} & \cdots & w_{n,m} \end{bmatrix} \tag{1}$$

Mathematically Representing Neural Networks (contd)

- The values of the nodes in layer $k+1$, denoted $A_{(k+1)}$ is given by the matrix equation $A_{(k+1)} = \sigma(WA_{(k)} + b_{(k)})$. Where b is the vector containing the biases, and $\sigma()$ is the *activation function*.

Mathematically Representing Neural Networks (contd)

- The values of the nodes in layer $k+1$, denoted $A_{(k+1)}$ is given by the matrix equation $A_{(k+1)} = \sigma(WA_{(k)} + b_{(k)})$. Where b is the vector containing the biases, and $\sigma()$ is the *activation function*.
- The choice of activation function is pretty much problem dependent, and depends a little bit on the data. In our case, we used the sigmoid activation function $\sigma : \mathbb{R} \to \mathbb{R}$ defined by $\sigma(x) = (1 + \exp(-x))^{-1}$.

How do Neural Networks "learn"?

- The idea of learning comes from updating the weights and biases to reduce the error made when predicting a result.

How do Neural Networks “learn”?

- The idea of learning comes from updating the weights and biases to reduce the error made when predicting a result.
- We start by initialising values of weights, biases, activation values by drawing random samples from a distribution.

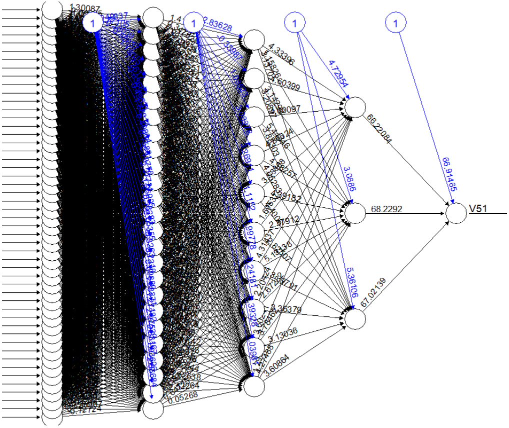
How do Neural Networks “learn”?

- The idea of learning comes from updating the weights and biases to reduce the error made when predicting a result.
- We start by initialising values of weights, biases, activation values by drawing random samples from a distribution.
- We pass training data through the network, and see what the output is. By comparing this output to the actual value for that piece of data, we get an error value.

How do Neural Networks "learn"?

- The idea of learning comes from updating the weights and biases to reduce the error made when predicting a result.
- We start by initialising values of weights, biases, activation values by drawing random samples from a distribution.
- We pass training data through the network, and see what the output is. By comparing this output to the actual value for that piece of data, we get an error value.
- We then employ the backpropogation algorithm to readjust the values of the weights and biases in accordance with the error. This looks to find a set of weights and biases that give the least error on the training data.

The Run-Rate Network



Different Scenarios

- Since the aim is to reset score targets when a full innings cannot be played, we have the issue of having 50 inputs to the network. There are two examples of how we could fix this.

Different Scenarios

- Since the aim is to reset score targets when a full innings cannot be played, we have the issue of having 50 inputs to the network. There are two examples of how we could fix this.

Example

Assume 35 overs of play have occured. Leaving 15 overs without data.

1. Assert a value of 0 on the remaining overs and see what the network predicts
2. Draw random values based on historical perfomance of teams in the remaining overs

Different Scenarios (contd)

- To test which method worked better, we took a test data set, and tested which option worked better.

Different Scenarios (contd)

- To test which method worked better, we took a test data set, and tested which option worked better.
- As it turns out, trying to fill in the gaps, didn't work at all. The predictions were way off and the corrolation between predicted results and actual results was almost 0.

Different Scenarios (contd)

- To test which method worked better, we took a test data set, and tested which option worked better.
- As it turns out, trying to fill in the gaps, didn't work at all. The predictions were way off and the corrolation between predicted results and actual results was almost 0.
- Leaving the unplayed overs blank was much more successfull, and had a corrolation of 0.49. So not great, but much better.

World Cup Simulation Study

- To test the method fully, we took the 3 games from the 2019 world cup that used DLS to decide a result, and used our method instead.

World Cup Simulation Study

- To test the method fully, we took the 3 games from the 2019 world cup that used DLS to decide a result, and used our method instead.
- Of the 3 games, only 1 had the result changed, which meant the final standings of the tournament were virtually unchanged, except Bangladesh finish above South Africa for 7^{th} place.

World Cup Simulation Study

- To test the method fully, we took the 3 games from the 2019 world cup that used DLS to decide a result, and used our method instead.
- Of the 3 games, only 1 had the result changed, which meant the final standings of the tournament were virtually unchanged, except Bangladesh finish above South Africa for 7^{th} place.
- The key difference was in the scores set, the score targets were far more attianable by the batting team, which would have lead to a more competative set of games. The India vs Pakistan game was ruined by DLS in this way. Since they needed 27.2 an over at one point.

Did we improve DLS?

- Not really. Aside from giving slightly more competative score targets, the method was still lacking.

Did we improve DLS?

- Not really. Aside from giving slightly more competative score targets, the method was still lacking.
- From an entertainment point of view, this method is better, but from a pure cricket perspective, DLS is still more reliable

Did we improve DLS?

- Not really. Aside from giving slightly more competative score targets, the method was still lacking.
- From an entertainment point of view, this method is better, but from a pure cricket perspective, DLS is still more reliable
- With more data, this method will become more reliable in the future, but this is constrained by the number of cricket games played each year.

Any Questions?