With the Rugby World Cup 2019 Japan starting on 20th September, I thought I’d take a look at the tournament from a few different statistical angles. For this post I’ll be looking at the problem: given a rugby score, how can we decompose it into possible combinations of tries, conversions, penalties and dropped goals?

**Context**

I have a dataframe of results for almost all professional and international rugby union scores since the 2012/13 season, more than 10,000 matches. This is nice in terms of ‘breadth’ of the sample – however in terms of depth it’s a bit lacking! For each result I only have home/away team and home/away score, for example:

27/07/2019 New Zealand South Africa 16 16

I was curious: is it possible to decompose the results into valid combinations of scoring methods? Then, perhaps as a second stage, estimate the probability of occurrence of each combination for a given score? The first question I will be looking at in this post, and the second will be next up in the series!

**I’ve never seen a match of rugby before! What are the scoring methods you’re referring to?**

TRY (5 points): awarded when an attacking player grounds the ball in the area at the end of the pitch (“in-goal area”).

CONVERSION (2 points): the team who has scored a try immediately gets to kick at goal for another 2 points before kick-off restart.

PENALTY GOAL (3 points): when an infringement is made, a penalty may be awarded to the other team who may then choose to take a penalty kick at goal.

DROP GOAL (3 points): a player may, at any time in play, drop-kick the ball over and between the posts.

PENALTY TRY (7 points): if a foul has stopped the attacking team from scoring then a penalty try is awarded, worth a full 7 points. *Note: these happen fairly rarely so I include this just for completeness but don’t refer to penalty tries hereon.*

I took the videos above from [World Rugby Laws of the Game](https://laws.worldrugby.org/) which is a great resource if you want to learn more about the laws of the game.

**Grouping the Elementary Scoring Methods**

* 3 points: penalty goal or drop goal.
* 5 points: unconverted try (i.e. a try has been scored, but the conversion did not score the extra 2 points)
* 7 points: converted try (i.e. a try has been scored, and the conversion succeeded in scoring the extra 2 points).

**Starting Off the Analysis: Scores 0-7**

| **Score** | **Penalties or drop goals (3pt ea)** | **Unconverted tries (5pt ea)** | **Converted tries (7pt ea)** |
| --- | --- | --- | --- |
| 0 | 0 | 0 | 0 |
| 3 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 |
| 6 | 2 | 0 | 0 |
| 7 | 0 | 0 | 1 |

* 0 is obviously a valid score.
* 3, 5, 7 are obtained from elementary scores only.
* 6 is obtained only from two penalties.
* 1, 2 and 4 are not valid scores as they cannot be sums of

**Onwards! Scores 8-10**

| **sc** | **pd** | **ut** | **ct** |
| --- | --- | --- | --- |
| 8 | 1 | 1 | 0 |
| 9 | 3 | 0 | 0 |
| 10 | 1 | 0 | 1 |
| 10 | 0 | 2 | 0 |

The only way forward is to score 3, 5, or 7 points! So new valid scores/combinations are {previous scores} + {3,5,7}. In the table above:

* 8 is the row for 5 points but plus 1 penalty/drop goal.
* 9 is the row for 6 points but plus 1 penalty/drop goal.
* 10 is either a converted try plus 1 penalty/drop goal OR an unconverted try plus another unconverted try (hence two rows).

**Scripting**

With the general rule established, it is fairly easy to script it:

for each i in 8:150

if i - 3 in validscores, copy row(s) of i - 3, increment pen/dg field by 1 and set score to i

if i - 5 in validscores... "

if i - 7 in validscores... "

I wrote a script in R, along with the results for scores up to 150 points.

|  |
| --- |
| library(dplyr) |
|  | #In rugby, a penalty or drop goal is worth 3 points, ("pd" in the code) |
|  | #an unconverted try is worth 5 points, ("ut" in the code) |
|  | #a converted try is worth 7 points. ("ct" in the code) |
|  | #Any rugby score is a composition of these elements. |
|  | # |
|  | #Let's set these up as the start of our validscores data frame. |
|  | validscores <- as.data.frame( |
|  | rbind(cbind(sc = 0,pd = 0, ut = 0,ct = 0), |
|  | cbind(3,1,0,0), |
|  | cbind(5,0,1,0), |
|  | cbind(6,2,0,0), #note that we include 6 for convenience: it's the result of 2 penalties or drop goals |
|  | cbind(7,0,0,1))) |
|  |  |
|  | for(i in 8:150){ #cover plausible scores (highest score in pro rugby since 2012: NZ women 121 vs HK women 0) |
|  |  |
|  | compositions\_for\_i <- NULL #let's store the valid compositions for score i in this object |
|  |  |
|  | #the score of interest could only be obtained from by adding 3, 5 or 7 to a lower valid score |
|  | #if such previous scores exist, let's find them |
|  | which\_was\_prev\_sc\_after\_pd <- which(i - validscores[,1] == 3) |
|  | which\_was\_prev\_sc\_after\_ut <- which(i - validscores[,1] == 5) |
|  | which\_was\_prev\_sc\_after\_ct <- which(i - validscores[,1] == 7) |
|  |  |
|  | #where the previous score exists: |
|  | if(length(which\_was\_prev\_sc\_after\_pd)>0) {temp1 <- validscores[which\_was\_prev\_sc\_after\_pd,] |
|  | temp1[,2] <- temp1[,2]+1 #let's copy the previous score's composition(s) and increment the appropriate score method |
|  | temp1[,1] <- i |
|  | compositions\_for\_i <- rbind(compositions\_for\_i,temp1)} #let's add this to the compositions df |
|  | if(length(which\_was\_prev\_sc\_after\_ut)>0) {temp1 <- validscores[which\_was\_prev\_sc\_after\_ut,] |
|  | temp1[,3] <- temp1[,3]+1 |
|  | temp1[,1] <- i |
|  | compositions\_for\_i <- rbind(compositions\_for\_i,temp1)} |
|  | if(length(which\_was\_prev\_sc\_after\_ct) > 0) {temp1 <- validscores[which\_was\_prev\_sc\_after\_ct,] |
|  | temp1[,4] <- temp1[,4]+1 |
|  | temp1[,1] <- i |
|  | compositions\_for\_i <- rbind(compositions\_for\_i,temp1)} |
|  |  |
|  | #sometimes we can end up with duplicates as order of scoring doesn't matter, so let's remove those |
|  | if(!is.null(compositions\_for\_i)) compositions\_for\_i <- distinct(compositions\_for\_i) |
|  |  |
|  | #we add these to our master validscores df and repeat for all scores, i |
|  | validscores <- rbind(validscores,compositions\_for\_i) |
|  | } |
|  | View(validscores) |

**What about the New Zealand vs South Africa match mentioned at the start?**

|  |  |  |  |
| --- | --- | --- | --- |
| **sc** | **pd** | **ut** | **ct** |
| 16 | 3 | 0 | 1 |
| 16 | 2 | 2 | 0 |

Both teams scored 16 points. Both teams got there through one converted try and three penalties, corresponding to the first row of two possible ways to reach 16 points.

**Are all scoring combinations equally likely?**

No, for a given score, not all scoring combinations are equally likely because even if all scoring methods were of equal probability (1/3 probability each), they contribute different amounts of points and so this would make the likelihood uneven!

The table below shows all of the possible scoring combinations relating to the score of 48 points.

|  |  |  |  |
| --- | --- | --- | --- |
| **sc** | **pd** | **ut** | **ct** |
| 48 | 16 | 0 | 0 |
| 48 | 12 | 1 | 1 |
| 48 | 11 | 3 | 0 |
| 48 | 9 | 0 | 3 |
| 48 | 9 | 0 | 3 |
| 48 | 7 | 4 | 1 |
| 48 | 6 | 6 | 0 |
| 48 | 5 | 1 | 4 |
| 48 | 4 | 3 | 3 |
| 48 | 3 | 5 | 2 |
| 48 | 2 | 7 | 1 |
| 48 | 2 | 0 | 6 |
| 48 | 1 | 9 | 0 |
| 48 | 1 | 2 | 5 |
| 48 | 0 | 4 | 4 |

It’s pretty unlikely that a team would ‘rack up’ so many points through 16 penalties/drop goals without scoring any tries! Equally, it’s unlikely a team would get there through just scoring tries alone. Intuitively, it would seem that it would be through a mixture that a team would be most likely to get there.

If we know the relative likelihood of occurrence of the three scoring methods to each other then we can calculate the probability of scoring combinations for a given score.