Something that has been talked a bit about recently on twitter is the the use of unique playerIDs so that fan analysts, punters and bloggers can track players through time.

There are some things that need to be thought about when creating unique playerIDS for analysis.

* Do you want your IDS to be unique across datasets or do you need them to work across datasets.

**The quick way without wanting to join datasets.**

Lets say you are using footywire data only, the quickest way would be to simply rely on domain knowledge give all unique players+club combination a unique ID and then manually change the players that have changed clubs.

Let me give a running example.

**Step one – concatentate together player and team**

The obvious issue here is that if we have a players like [Adam Treloar](https://www.footywire.com/afl/footy/pp-collingwood-magpies--adam-treloar) who have played for multiple teams

The other obvious issue is that we have players that have the same name like [Jack Trengove](https://twitter.com/RankingSw/status/1095260427464765440)

So when coming up with a playerID for the first time we have to think about what is our quickest solution for use.

If we were only interested within year, we could just create a quick playerID based on joining together player names and teams.

Which we can do below.

library(fitzRoy)

library(tidyverse)

## ── Attaching packages ───────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 3.1.0 ✔ purrr 0.3.0

## ✔ tibble 2.0.1 ✔ dplyr 0.7.8

## ✔ tidyr 0.8.2 ✔ stringr 1.3.1

## ✔ readr 1.3.1 ✔ forcats 0.3.0

## ── Conflicts ──────────────────── tidyverse\_conflicts() ──

## ✖ dplyr::filter() masks stats::filter()

## ✖ dplyr::lag() masks stats::lag()

df<-fitzRoy::player\_stats

df$player\_id\_step1<-paste(df$Player, df$Team)

However the issue with that is we can’t track a player like [Adam Treloar](https://www.footywire.com/afl/footy/pp-collingwood-magpies--adam-treloar) through the dataset as he will have two IDS one for when he was at Collingwood and one from this time at GWS.

So instead of that we might want to use names instead. This has the issue that if players have the same name but played on different teams they will be given the same ID.

To find out how big of an issue this would be we could summarise the name count by unique teams within an AFL season.

check<-df%>%distinct(Season, Player, Team)%>%

group\_by(Season, Player)%>%

summarise(player\_counts=n())%>%

arrange(desc(player\_counts))

head(check)

## # A tibble: 6 x 3

## # Groups: Season [3]

## Season Player player\_counts

##

## 1 2010 Scott Thompson 2

## 2 2011 Mitchell Brown 2

## 3 2011 Scott Thompson 2

## 4 2011 Tom Lynch 2

## 5 2012 Mitchell Brown 2

## 6 2012 Scott Thompson 2

What this means is that in 2010 we had 2 players named Scott Tompson and they played for different teams.

But we have to manually recode this, so to find out how many players we would have to do this for we can do this by just filtering out player\_counts>1 i.e. when there are multiple in the same year but for different teams.

df%>%distinct(Season, Player, Team)%>%

group\_by(Season, Player)%>%

summarise(player\_counts=n())%>%

arrange(desc(player\_counts))%>%filter(player\_counts>1)

## # A tibble: 21 x 3

## # Groups: Season [9]

## Season Player player\_counts

##

## 1 2010 Scott Thompson 2

## 2 2011 Mitchell Brown 2

## 3 2011 Scott Thompson 2

## 4 2011 Tom Lynch 2

## 5 2012 Mitchell Brown 2

## 6 2012 Scott Thompson 2

## 7 2012 Tom Lynch 2

## 8 2013 Mitchell Brown 2

## 9 2013 Scott Thompson 2

## 10 2013 Tom Lynch 2

## # … with 11 more rows

So that doesn’t seem too bad if we just coded each player name with a unique ID, our next step would be to then manually change any players who need to get a new ID, this would rely on domain knowledge done this way.

**Using player and team as an ID**

This wouldnt be the best if we were tracking players across years. Perhaps we are looking at age effects or we want to visualise the cummulative count of a players stats.

To get a feel for why that is a bit more tedius than the above, lets just look at the amount of players who played for different clubs (these would all have to be manually recoded)

df1<-df%>%distinct(Player)

id <- rownames(df1)

df2<-cbind(df1,id)

df\_check<-df%>%left\_join(df2,by ="Player")

df\_check%>%

distinct(Player, id, Team, Season)%>%

arrange(Player)%>%

distinct(Player, id, Team)%>%

group\_by(Player, id)%>%

summarise(count\_player=n())%>%

arrange(desc(count\_player))%>%

filter(count\_player>1)

## # A tibble: 320 x 3

## # Groups: Player [320]

## Player id count\_player

##

## 1 Andrejs Everitt 274 3

## 2 Ben Hudson 283 3

## 3 Jed Lamb 898 3

## 4 Jeremy Laidler 570 3

## 5 Jonathan Giles 739 3

## 6 Jonathan Simpkin 800 3

## 7 Josh Caddy 724 3

## 8 Koby Stevens 528 3

## 9 Kyle Cheney 540 3

## 10 Matthew Spangher 360 3

## # … with 310 more rows

So we can see that’s actually quite a bit of players we’d have to go through manually.

So what we need to do is essentially have two lists ready to go for our unique playerIDs. We need one for our dataset that we know will have errors (Scott Thompsons having the same ID.

The second list we would keep is the one that contains players that share names, this will form the basis of our recoding of the IDS so that everyone has a unique one.

df<-fitzRoy::player\_stats

df1<-df%>%distinct(Player)

id <- rownames(df1)

length(id)

## [1] 1392

df2<-cbind(df1,id)

So here we have df2, which contains the player names with the ids, this is our second list.

Now lets join on the IDS to the dataset of [footywire](https://www.footywire.com/) data

To do this we can left\_join our data.

df\_id<-df%>%left\_join(df2,by ="Player")

df\_id$id<-as.numeric\_version(df\_id$id)

So df\_id as it stands now will give players that share names like Scott Thompson the same ID even though they are different players.

df\_id%>%

select(Season, Team, Player, id)%>%

filter(Player %in% c("Scott Thompson"))%>%

distinct(Season, Team, Player, as.factor(id))

## Season Team Player as.factor(id)

## 1 2010 North Melbourne Scott Thompson 263

## 2 2010 Adelaide Scott Thompson 263

## 3 2011 Adelaide Scott Thompson 263

## 4 2011 North Melbourne Scott Thompson 263

## 5 2012 Adelaide Scott Thompson 263

## 6 2012 North Melbourne Scott Thompson 263

## 7 2013 Adelaide Scott Thompson 263

## 8 2013 North Melbourne Scott Thompson 263

## 9 2014 Adelaide Scott Thompson 263

## 10 2014 North Melbourne Scott Thompson 263

## 11 2015 North Melbourne Scott Thompson 263

## 12 2015 Adelaide Scott Thompson 263

## 13 2016 North Melbourne Scott Thompson 263

## 14 2016 Adelaide Scott Thompson 263

## 15 2017 North Melbourne Scott Thompson 263

## 16 2017 Adelaide Scott Thompson 263

## 17 2018 North Melbourne Scott Thompson 263

Lets replace as an example [Scott Thompson for the Crows](https://www.footywire.com/afl/footy/pp-adelaide-crows--scott-thompson) with a different id from 263 which will be[Scott Thompsons id who played for the Kangaroos](https://www.footywire.com/afl/footy/pp-kangaroos--scott-thompson-1)

So to come up with the new id for Scott thompson for the Crows, we need to make sure we don’t reuse a number. A simple solution would just be to take the maximum the dataset and give Scott Thompson max+1 as an ID.

max(df\_id$id)

## [1] '1392'

So lets replace the 263 with 1392 +1

df\_id$id[df\_id$Player =="Scott Thompson" & df\_id$Team=="Adelaide"] = 1393

We can then check it as follows.

df\_id%>%

select(Season, Team, Player, id)%>%

filter(Player %in% c("Scott Thompson"))%>%

distinct(Season, Team, Player, as.factor(id))

## Season Team Player as.factor(id)

## 1 2010 North Melbourne Scott Thompson 263

## 2 2010 Adelaide Scott Thompson 1393

## 3 2011 Adelaide Scott Thompson 1393

## 4 2011 North Melbourne Scott Thompson 263

## 5 2012 Adelaide Scott Thompson 1393

## 6 2012 North Melbourne Scott Thompson 263

## 7 2013 Adelaide Scott Thompson 1393

## 8 2013 North Melbourne Scott Thompson 263

## 9 2014 Adelaide Scott Thompson 1393

## 10 2014 North Melbourne Scott Thompson 263

## 11 2015 North Melbourne Scott Thompson 263

## 12 2015 Adelaide Scott Thompson 1393

## 13 2016 North Melbourne Scott Thompson 263

## 14 2016 Adelaide Scott Thompson 1393

## 15 2017 North Melbourne Scott Thompson 263

## 16 2017 Adelaide Scott Thompson 1393

## 17 2018 North Melbourne Scott Thompson 263

Looking at this it seems to have worked.

Now we just have to do this for all the players that have the same name in our dataset.

So lets get our list of players that share the same name.

players\_share\_name<-df%>%

distinct(Season, Player, Team)%>%

group\_by(Season, Player)%>%

summarise(player\_counts=n())%>%

arrange(desc(player\_counts))%>%filter(player\_counts>1)%>%

select(Player)

## Adding missing grouping variables: `Season`

df%>%filter(Player %in% players\_share\_name$Player)%>%

distinct(Season, Player, Team)%>%

arrange(Player)

## Season Player Team

## 1 2010 Mitchell Brown West Coast

## 2 2011 Mitchell Brown West Coast

## 3 2011 Mitchell Brown Geelong

## 4 2012 Mitchell Brown Geelong

## 5 2012 Mitchell Brown West Coast

## 6 2013 Mitchell Brown Geelong

## 7 2013 Mitchell Brown West Coast

## 8 2014 Mitchell Brown Geelong

## 9 2014 Mitchell Brown West Coast

## 10 2015 Mitchell Brown West Coast

## 11 2016 Mitchell Brown Essendon

## 12 2016 Mitchell Brown West Coast

## 13 2017 Mitchell Brown Essendon

## 14 2018 Mitchell Brown Essendon

## 15 2010 Scott Thompson North Melbourne

## 16 2010 Scott Thompson Adelaide

## 17 2011 Scott Thompson Adelaide

## 18 2011 Scott Thompson North Melbourne

## 19 2012 Scott Thompson Adelaide

## 20 2012 Scott Thompson North Melbourne

## 21 2013 Scott Thompson Adelaide

## 22 2013 Scott Thompson North Melbourne

## 23 2014 Scott Thompson Adelaide

## 24 2014 Scott Thompson North Melbourne

## 25 2015 Scott Thompson North Melbourne

## 26 2015 Scott Thompson Adelaide

## 27 2016 Scott Thompson North Melbourne

## 28 2016 Scott Thompson Adelaide

## 29 2017 Scott Thompson North Melbourne

## 30 2017 Scott Thompson Adelaide

## 31 2018 Scott Thompson North Melbourne

## 32 2010 Tom Lynch St Kilda

## 33 2011 Tom Lynch Gold Coast

## 34 2011 Tom Lynch St Kilda

## 35 2012 Tom Lynch Gold Coast

## 36 2012 Tom Lynch Adelaide

## 37 2013 Tom Lynch Adelaide

## 38 2013 Tom Lynch Gold Coast

## 39 2014 Tom Lynch Gold Coast

## 40 2014 Tom Lynch Adelaide

## 41 2015 Tom Lynch Adelaide

## 42 2015 Tom Lynch Gold Coast

## 43 2016 Tom Lynch Gold Coast

## 44 2016 Tom Lynch Adelaide

## 45 2017 Tom Lynch Gold Coast

## 46 2017 Tom Lynch Adelaide

## 47 2018 Tom Lynch Gold Coast

## 48 2018 Tom Lynch Adelaide

So what we can see here that within the fitzRoy dataset that contains footywire data, we just have to make the following replacements.

* Replace the ID for Scott Thompson at [Adelaide](https://www.footywire.com/afl/footy/pp-adelaide-crows--scott-thompson) vs [North Melbourne](https://www.footywire.com/afl/footy/pp-kangaroos--scott-thompson-1)
* Replace the ID for Tom Lynch at [Adelaide](https://www.footywire.com/afl/footy/pp-adelaide-crows--tom-lynch) and [Gold Coast now at Richmond](https://www.footywire.com/afl/footy/pp-richmond-tigers--thomas-lynch)
* Replace the ID for Mitchell Brown who [played at Geelong and Essendon](https://www.footywire.com/afl/footy/pc-essendon-bombers--mitchell-brown-1) vs Mitchell Brown at [West Coast](https://www.footywire.com/afl/footy/pp-west-coast-eagles--mitchell-brown)

So lets make those replacements!

df\_id$id[df\_id$Player =="Scott Thompson" & df\_id$Team=="Adelaide"] = 1393

df\_id$id[df\_id$Player =="Mitchell Brown" & df\_id$Team=="West Coast"] = 1394

df\_id$id[df\_id$Player =="Tom Lynch" & df\_id$Team=="Adelaide"] = 1395

Check to see it all worked

df\_id%>%filter(Player %in% c("Tom Lynch", "Scott Thompson", "Mitchell Brown"))%>%

distinct(Player, Team, Season, as.factor(id))

## Season Player Team as.factor(id)

## 1 2010 Mitchell Brown West Coast 1394

## 2 2010 Scott Thompson North Melbourne 263

## 3 2010 Scott Thompson Adelaide 1393

## 4 2010 Tom Lynch St Kilda 554

## 5 2011 Scott Thompson Adelaide 1393

## 6 2011 Mitchell Brown West Coast 1394

## 7 2011 Scott Thompson North Melbourne 263

## 8 2011 Tom Lynch Gold Coast 554

## 9 2011 Tom Lynch St Kilda 554

## 10 2011 Mitchell Brown Geelong 164

## 11 2012 Tom Lynch Gold Coast 554

## 12 2012 Scott Thompson Adelaide 1393

## 13 2012 Tom Lynch Adelaide 1395

## 14 2012 Scott Thompson North Melbourne 263

## 15 2012 Mitchell Brown Geelong 164

## 16 2012 Mitchell Brown West Coast 1394

## 17 2013 Scott Thompson Adelaide 1393

## 18 2013 Scott Thompson North Melbourne 263

## 19 2013 Mitchell Brown Geelong 164

## 20 2013 Mitchell Brown West Coast 1394

## 21 2013 Tom Lynch Adelaide 1395

## 22 2013 Tom Lynch Gold Coast 554

## 23 2014 Tom Lynch Gold Coast 554

## 24 2014 Mitchell Brown Geelong 164

## 25 2014 Scott Thompson Adelaide 1393

## 26 2014 Scott Thompson North Melbourne 263

## 27 2014 Mitchell Brown West Coast 1394

## 28 2014 Tom Lynch Adelaide 1395

## 29 2015 Mitchell Brown West Coast 1394

## 30 2015 Tom Lynch Adelaide 1395

## 31 2015 Scott Thompson North Melbourne 263

## 32 2015 Tom Lynch Gold Coast 554

## 33 2015 Scott Thompson Adelaide 1393

## 34 2016 Tom Lynch Gold Coast 554

## 35 2016 Mitchell Brown Essendon 164

## 36 2016 Scott Thompson North Melbourne 263

## 37 2016 Scott Thompson Adelaide 1393

## 38 2016 Tom Lynch Adelaide 1395

## 39 2016 Mitchell Brown West Coast 1394

## 40 2017 Tom Lynch Gold Coast 554

## 41 2017 Scott Thompson North Melbourne 263

## 42 2017 Tom Lynch Adelaide 1395

## 43 2017 Mitchell Brown Essendon 164

## 44 2017 Scott Thompson Adelaide 1393

## 45 2018 Mitchell Brown Essendon 164

## 46 2018 Tom Lynch Gold Coast 554

## 47 2018 Scott Thompson North Melbourne 263

## 48 2018 Tom Lynch Adelaide 1395

* Now you should be able to track a players history throughout fitzRoy.
* Now you should be able to check that all of afltables ids as contained within fitzRoy are unique
* hopefully enough is there so you can join the datasets together (afltables and footywire)