Dataset overview

Below are all the datasets that are contained within the package.

Season summary

A data frame containing summary details of each season of Survivor, including the winner, runner ups and location. This is a nested data frame given there maybe 1 or 2 runner-ups. By using a nested data frame the grain is maintained to 1 row per season.

```
season_summary
```

```
#> # A tibble: 40 x 17
#>    season_name season location country tribe_setup full_name winner
runner_ups
#>
#>  1 Survivor: ~      1 Pulau T~ Malays~ Two tribes~ Richa~ Richard
2 Survivor: ~      2 Herbert~ Austra~ Two tribes~ Tina ~ Tina       3
Survivor: ~      3 Shaba N~ Kenya   Two tribes~ Ethan~ Ethan      4
Survivor: ~      4 Nuku Hi~ Polyne~ Two tribes~ Vecep~ Vecepia    5
Survivor: ~      5 Ko Taru~ Thaila~ Two tribes~ Brian~ Brian      6
Survivor: ~      6 Rio Neg~ Brazil  Two tribes~ Jenna~ Jenna      7
Survivor: ~      7 Pearl I~ Panama  Two tribes~ Sandr~ Sandra     8
Survivor: ~      8 Pearl I~ Panama  Three trib~ Amber~ Amber      9
Survivor: ~      9 Efate, ~ Vanuatu Two tribes~ Chris~ Chris     10
Survivor: ~     10 Koror, ~ Palau   A schoolya~ Tom W~ Tom       # ...
with 30 more rows, and 9 more variables: final_vote ,
#> #   timeslot , premiered , premier_viewers , ended ,
#> #   finale_viewers , reunion_viewers , rank , viewers
```

```
season_summary %>%
  select(season, viewers_premier, viewers_finale, viewers_reunion,
viewers_mean) %>%
  pivot_longer(cols = -season, names_to = "episode", values_to =
"viewers") %>%
  mutate(
    episode = to_title_case(str_replace(episode, "viewers_", ""))
  ) %>%
  ggplot(aes(x = season, y = viewers, colour = episode)) +
  geom_line() +
  geom_point(size = 2) +
  theme_minimal() +
  scale_colour_survivor(16) +
  labs(
    title = "Survivor viewers over the 40 seasons",
    x = "Season",
    y = "Viewers (Millions)",
    colour = "Episode"
```

```
)
```

> The number of viewers for each season of Survivor has been steadily decreasing, however the mean number of viewers has only dropped by 3-4 million over the last 20 seasons (or 10 years).

Castaways

Season and demographic information about each castaway. Within a season the data is ordered by the first voted out to sole survivor indicated by `order` which represents the order they castaways left the island. This may be by being voted off the island, being evacuated due to medical reasons, or quitting. When demographic information is missing, it likely means that the castaway re-entered the game at a later stage by winning the opportunity to return. Castaways that have played in multiple seasons will feature more than once with the age and location representing that point in time.

```
castaways %>%
  filter(season == 40)
```

```
#> # A tibble: 22 x 15
#>    season_name season castaway nickname age   city  state   day
original_tribe
#>
#>  1 Survivor: ~     40 Natalie~ Natalie            2 Sele
#>  2 Survivor: ~     40 Amber M~ Amber    40    Pens~ Flor~     3
Dakal
#>  3 Survivor: ~     40 Danni B~ Danni    43    Shaw~ Kans~     6 Sele
#>  4 Survivor: ~     40 Ethan Z~ Ethan    45    Hill~ New ~     9 Sele
#>  5 Survivor: ~     40 Tyson A~ Tyson           11 Dakal
#>  6 Survivor: ~     40 Rob Mar~ Rob      43    Pens~ Flor~    14 Sele
#>  7 Survivor: ~     40 Parvati~ Parvati  36    Los ~ Cali~    16 Sele
#>  8 Survivor: ~     40 Sandra ~ Sandra   44    Rive~ Flor~    16
Dakal
#>  9 Survivor: ~     40 Yul Kwon Yul      44    Los ~ Cali~    18
Dakal
#> 10 Survivor: ~     40 Wendell~ Wendell  35    Phil~ Penn~    21
Dakal
#> # ... with 12 more rows, and 6 more variables: merged_tribe ,
#> #   result , jury_status , order , swapped_tribe ,
#> #   swapped_tribe2
```

Vote history

This data frame contains a complete history of votes cast across all seasons of Survivor. This allows you to see who voted for who at which tribal council. It also includes details on who had individual immunity as well as who had their votes nullified by a hidden immunity idol. This details the key events for the season.

While there are consistent events across the seasons such as the tribe swap, there are some unique events such as the ‘mutiny’ in Survivor: Cook Islands (Season 13) or the ‘Outcasts’ in Survivor: Pearl Islands (season 7). When castaways change tribes by some means other than a

tribe swap, it is still recorded as ‘swapped’ to maintain a standard.

The data is recorded as ‘swapped’ with a trailing digit if a swap has occurred more than once. This includes absorbed tribes when 3 tribes are reduced to 2 or when Stephanie was ‘absorbed’ in Survivor: Palau (season 10) when everyone but herself was voted off the tribe (and making Palau one of the classic seasons of Survivor). To indicate a change in tribe status these events are also considered ‘swapped’.

This data frame is at the tribal council by castaway grain, so there is a vote for everyone that attended the tribal council. However, there are some edge cases such as when the ‘steal a vote’ advantage is played. In this case, there is a second row for the castaway indicating their second vote.

In the case of a tie and a revote, the first vote is recorded and the result is recorded as ‘Tie’. The deciding vote is recorded as normal. Where there is a double tie, it is recorded as ‘Tie2’ (for lack of a better name). In the case of a double tie and it goes to rocks, the vote is either ‘Black rock’ or ‘White rock’. In the older episodes of Survivor, when there were two ties in a row, rather than going to rocks there was a countback of votes.

```
vh <- vote_history %>%
  filter(
    season == 40,
    episode == 10
  )
vh
```

```
#> # A tibble: 9 x 11
#>   season_name season episode   day tribe_status castaway immunity
vote
#>
#> 1 Survivor: ~     40      10    25 merged       Tony     individ~
Tyson
#> 2 Survivor: ~     40      10    25 merged       Michele       Tyson
#> 3 Survivor: ~     40      10    25 merged       Sarah         Deni~
#> 4 Survivor: ~     40      10    25 merged       Sarah         Tyson
#> 5 Survivor: ~     40      10    25 merged       Ben           Tyson
#> 6 Survivor: ~     40      10    25 merged       Nick          Tyson
#> 7 Survivor: ~     40      10    25 merged       Kim           Soph~
#> 8 Survivor: ~     40      10    25 merged       Sophie        Deni~
#> 9 Survivor: ~     40      10    25 merged       Tyson         Soph~
#> # ... with 3 more variables: nullified , voted_out , order
```

```
vh %>%
  count(vote)
```

```
#> # A tibble: 5 x 2
#>   vote       n
#>
#> 1 Denise     2
#> 2 Immune     1
```

```
#> 3 None       1
#> 4 Sophie     2
#> 5 Tyson      5
```

Events in the game such as fire challenges, rock draws, steal-a-vote advantages, or countbacks (in the early days) often mean a vote wasn't placed for an individual. Rather a challenge may be won, lost, no vote cast, etc but attended tribal council. These events are recorded in the `vote` field. I have included a function `clean_votes` for when only the votes cast for individuals are needed. If the input data frame has the `vote` column it can simply be piped.

```
vh %>%
  clean_votes() %>%
  count(vote)
```

```
#> # A tibble: 3 x 2
#>   vote       n
#>
#> 1 Denise     2
#> 2 Sophie     2
#> 3 Tyson      5
```

Immunity

A nested tidy data frame of immunity challenge results. Each row in this dataset is a tribal council. It is a nested data frame since there may be multiple people or tribes that win immunity. But more so multiple tribes when there are 3 or more tribes in the first phase of the game. You can extract the immunity winners by expanding the data frame. There may be duplicates for the rare event when there are multiple eliminations after a single immunity challenge.

```
immunity %>%
  filter(season == 40) %>%
  unnest(immunity)
```

```
#> # A tibble: 23 x 8
#>    season_name       season episode title         voted_out   day
order immunity
#>
#>  1 Survivor: Winner~     40       1 Greatest of ~ Natalie       2
1 Dakal
#>  2 Survivor: Winner~     40       1 Greatest of ~ Amber         3
2 Sele
#>  3 Survivor: Winner~     40       2 It's Like a ~ Danni         6
3 Dakal
#>  4 Survivor: Winner~     40       3 Out for Blood Ethan         9
4 Dakal
#>  5 Survivor: Winner~     40       4 I Like Reven~ Tyson        11
5 Sele
#>  6 Survivor: Winner~     40       5 The Buddy Sy~ Rob          14
6 Sele
#>  7 Survivor: Winner~     40       5 The Buddy Sy~ Rob          14
6 Dakal
```

```
#>  8 Survivor: Winner~     40       6 Quick on the~ Parvati      16
7 Yara
#>  9 Survivor: Winner~     40       6 Quick on the~ Sandra       16
8 Yara
#> 10 Survivor: Winner~     40       7 We're in the~ Yul          18
9 Yara
#> # ... with 13 more rows
```

Rewards

A nested tidy data frame of reward challenge result where each row is a reward challenge. Typically in the merge, if a single person wins a reward they are allowed to bring others along with them. The first castaway in the expanded list is the winner. Subsequent players are those who the winner brought along with them to the reward. Although, not always. Occasionally in the merge, the castaways are split into two teams for the purpose of the reward, in which case all castaways win the reward rather than a single person. If `reward` is missing there was no reward challenge for the episode.

```
rewards %>%
  filter(season == 40) %>%
  unnest(reward)
```

```
#> # A tibble: 29 x 6
#>    season_name            season episode title
day reward
#>
#>  1 Survivor: Winners at ~     40       1 Greatest of the Greats
2 Dakal
#>  2 Survivor: Winners at ~     40       1 Greatest of the Greats
3
#>  3 Survivor: Winners at ~     40       2 It's Like a Survivor Econ~
6 Dakal
#>  4 Survivor: Winners at ~     40       3 Out for Blood
9 Dakal
#>  5 Survivor: Winners at ~     40       4 I Like Revenge
11 Sele
#>  6 Survivor: Winners at ~     40       5 The Buddy System on Stero~
14
#>  7 Survivor: Winners at ~     40       6 Quick on the Draw
16 Yara
#>  8 Survivor: Winners at ~     40       7 We're in the Majors
18 Yara
#>  9 Survivor: Winners at ~     40       7 We're in the Majors
18 Sele
#> 10 Survivor: Winners at ~     40       8 This is Where the Battle ~
21 Tyson
#> # ... with 19 more rows
```

Jury votes

This data frame contains the history of jury votes. It is more verbose than it needs to be. However, having a 0-1 column indicating if a vote was placed for the finalist makes it easier to

summarise castaways that received no votes.

```
jury_votes %>%
  filter(season == 40)
```

```
#> # A tibble: 48 x 5
#>    season_name              season castaway finalist  vote
#>
#>  1 Survivor: Winners at War     40 Sarah    Michele      0
#>  2 Survivor: Winners at War     40 Sarah    Natalie      0
#>  3 Survivor: Winners at War     40 Sarah    Tony         1
#>  4 Survivor: Winners at War     40 Ben      Michele      0
#>  5 Survivor: Winners at War     40 Ben      Natalie      0
#>  6 Survivor: Winners at War     40 Ben      Tony         1
#>  7 Survivor: Winners at War     40 Denise   Michele      0
#>  8 Survivor: Winners at War     40 Denise   Natalie      0
#>  9 Survivor: Winners at War     40 Denise   Tony         1
#> 10 Survivor: Winners at War     40 Nick     Michele      0
#> # ... with 38 more rows
```

```
jury_votes %>%
  filter(season == 40) %>%
  group_by(finalist) %>%
  summarise(votes = sum(vote))
```

```
#> # A tibble: 3 x 2
#>   finalist votes
#>
#> 1 Michele      0
#> 2 Natalie      4
#> 3 Tony        12
```

Viewers

A data frame containing the viewer information for every episode across all seasons. It also includes the rating and viewer share information for viewers aged 18 to 49 years.

```
viewers %>%
  filter(season == 40)
```

```
#> # A tibble: 14 x 9
#>    season_name season episode_number_~ episode title episode_date
viewers
#>
#>  1 Survivor: ~     40              583       1 Grea~ 2020-02-12
6.68
#>  2 Survivor: ~     40              584       2 It's~ 2020-02-19
7.16
```

```
#>  3 Survivor: ~     40             585       3 Out ~ 2020-02-26
7.14
#>  4 Survivor: ~     40             586       4 I Li~ 2020-03-04
7.08
#>  5 Survivor: ~     40             587       5 The ~ 2020-03-11
6.91
#>  6 Survivor: ~     40             588       6 Quic~ 2020-03-18
7.83
#>  7 Survivor: ~     40             589       7 We'r~ 2020-03-25
8.18
#>  8 Survivor: ~     40             590       8 This~ 2020-04-01
8.23
#>  9 Survivor: ~     40             591       9 War ~ 2020-04-08
7.85
#> 10 Survivor: ~     40             592      10 The ~ 2020-04-15
8.14
#> 11 Survivor: ~     40             593      11 This~ 2020-04-22
8.16
#> 12 Survivor: ~     40             594      12 Frie~ 2020-04-29
8.08
#> 13 Survivor: ~     40             595      13 The ~ 2020-05-06
7.57
#> 14 Survivor: ~     40             596      14 It A~ 2020-05-13
7.94
#> # ... with 2 more variables: rating_18_49 , share_18_49
```

Tribe colours

This data frame contains the tribe names and colours for each season, including the RGB values. These colours can be joined with the other data frames to customise colours for plots. Another option is to add tribal colours to ggplots with the scale functions.

```
tribe_colours
```

```
#> # A tibble: 139 x 7
#>    season_name                  season tribe_name     r     g     b
tribe_colour
#>
#>  1 Survivor: Winners at War         40 Sele           0   103   214
#0067D6
#>  2 Survivor: Winners at War         40 Dakal        216    14    14
#D80E0E
#>  3 Survivor: Winners at War         40 Yara           4   148    81
#049451
#>  4 Survivor: Winners at War         40 Koru           0     0     0
#000000
#>  5 Survivor: Island of the Ido~     39 Lairo        243   148    66
#F39442
#>  6 Survivor: Island of the Ido~     39 Vokai        217   156   211
#D99CD3
#>  7 Survivor: Island of the Ido~     39 Lumuwaku      48    78   210
#304ED2
```

```
#>  8 Survivor: Edge of Extinction     38 Manu          16    80   186
#1050BA
#>  9 Survivor: Edge of Extinction     38 Lesu           0   148   128
#009480
#> 10 Survivor: Edge of Extinction     38 Kama         250   207    34
#FACF22
#> # ... with 129 more rows
```

Tribe colours for each season of Survivor

ggplot2 scale functions

Included are ggplot2 scale functions (of the form `scale_*_survivor()`) to add tribe colours to ggplot. Simply input the season number desired to use those tribe colours. If the fill or colour aesthetic is the tribe name, this needs to be passed to the scale function as `scale_fill_survivor(…, tribe = tribe)` (for now) where `tribe` is on the input data frame. If the fill or colour aesthetic is independent of the actual tribe names, `tribe` does not need to be specified and will simply use the tribe colours as a colour palette, for example, the viewers line graph above which used the Micronesia colour palette.

```
ssn <- 35
labels <- castaways %>%
  filter(
    season == ssn,
    str_detect(result, "Sole|unner")
  ) %>%
  select(nickname, original_tribe) %>%
  mutate(label = glue("{nickname} ({original_tribe})")) %>%
  select(label, nickname)
jury_votes %>%
  filter(season == ssn) %>%
  left_join(
    castaways %>%
      filter(season == ssn) %>%
      select(nickname, original_tribe),
    by = c("castaway" = "nickname")
    ) %>%
  group_by(finalist, original_tribe) %>%
  summarise(votes = sum(vote)) %>%
  left_join(labels, by = c("finalist" = "nickname")) %>%
  {
    ggplot(., aes(x = label, y = votes, fill = original_tribe)) +
    geom_bar(stat = "identity", width = 0.5) +
    scale_fill_survivor(ssn, tribe = .$original_tribe) +
    theme_minimal() +
    labs(
      x = "Finalist (original tribe)",
      y = "Votes",
      fill = "Original\ntribe",
      title = "Votes received by each finalist"
    )
```

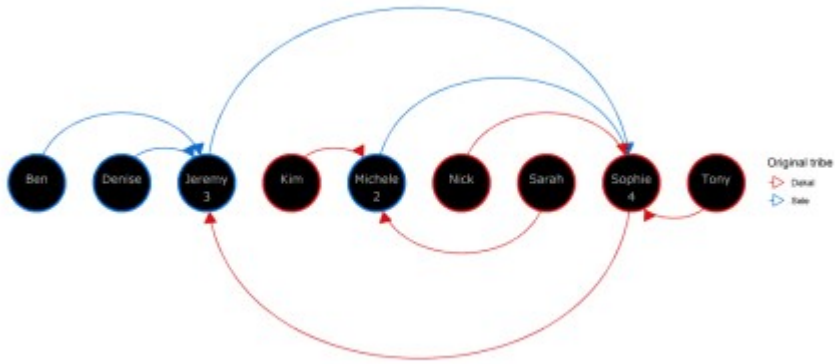
```
  }
```

Visualise the events of each season

This data provides a way to deeper analyse each season and the plays within each episode. For example, we could construct a graph of who voted for who, where the castaway is the node and the edge is who they voted for using the vote history data. While in this representation it's possible to use clustering algorithms to identify alliances in the data. Other uses include identifying the probability of players jumping ship and pivotal votes. This is particularly interesting for the first 1 or 2 tribals of the merge to see if players stick with their original tribe or jump ship.

```
ssn <- 40

df <- vote_history %>%
  filter(
    season == ssn,
    order == 13
    )

nodes <- df %>%
  distinct(castaway) %>%
  mutate(id = 1:n()) %>%
  rename(label = castaway)

edges <- df %>%
  count(castaway, vote) %>%
  left_join(
    nodes %>%
      rename(from = id),
    by = c("castaway" = "label")
  ) %>%
  left_join(
    nodes %>%
      rename(to = id),
    by = c("vote" = "label")
  ) %>%
  mutate(arrows = "to") %>%
  rename(value = n) %>%
  left_join(
  castaways %>%
    filter(season == ssn) %>%
    select(nickname, original_tribe),
  by = c("castaway" = "nickname")
)

labels <- edges %>%
  select(from, to, castaway, original_tribe) %>%
  distinct(from, castaway, original_tribe) %>%
  arrange(castaway) %>%
```

```
  left_join(
    edges %>%
      count(vote),
    by = c("castaway" = "vote")
  )

cols <- tribe_colours$tribe_colour
names(cols) <- tribe_colours$tribe
ggraph(
  edges %>%
    rename(`Original tribe` = original_tribe),
  layout = "linear") +
  geom_edge_arc(aes(colour = `Original tribe`), arrow = arrow(length =
unit(4, "mm"), type = "closed"), end_cap = circle(10, 'mm')) +
  geom_node_point(size = 26, colour = cols[labels$original_tribe]) +
  geom_node_point(size = 24, colour = "black") +
  geom_node_text(aes(label = labels$castaway), colour = "grey", size =
4, vjust = 0, family = ft) +
  geom_node_text(aes(label = labels$n), colour = "grey", size = 4,
vjust = 2, family = ft) +
  scale_edge_colour_manual(values = cols[unique(edges$original_tribe)])
+
  scale_colour_manual(values = cols[unique(edges$original_tribe)]) +
  theme_graph()
```



Vote distribution for episode 11 of Survivor: Winners at War. Sophie was the 13th person voted off the island

New features and future seasons

I intend to update the survivoR package each week during the airing of future seasons. For Survivor and data nuts like myself, this will enable a deeper analysis of each episode, and just neat ways visualise the evolution of the game.