Advent of Code 2020

Zach Bogart

2020-12-12

Day 9: Encoding Error

Click for Problem Statement

```
input = tibble(x = readLines("inputs/09-input.txt")) %>%
  mutate(x = as.numeric(x),
         row = row_number())
```

Part 1

```
check_current_value = function(input, n) {
  start = n-25
  end = n-1

  check_data = input %>%
    slice(start: end) %>%
    pull(x)

  check_grid = expand.grid(check_data, check_data) %>%
    mutate(sum = Var1 + Var2) %>%
    pull(sum) %>%
    unique()

  current_value = pull(input[n, "x"])

  return(current_value %in% check_grid)
}
```

```
data = input %>%
  slice(26:nrow(.))
```

```
part1 = data %>%
  rowwise() %>%
  mutate(in_previous_sum = check_current_value(input, row))
```

```
part1 %>%
  filter(!in_previous_sum)
```

```
## # A tibble: 1 x 3
## # Rowwise:
##           x   row in_previous_sum
##       <dbl> <int> <lgl>
## 1 105950735   565 FALSE
```

Part 2

```
invalid_number = part1 %>%
  filter(!in_previous_sum) %>%
  pull(x)
```

The rest of this is a mess...

Understanding David Robinson's solution:

```
candidates = input$x[input$x < invalid_number] # just keep the numbers that are below part1 answer

x = crossing(start = seq_along(candidates),
         end = seq_along(candidates)) %>% # crossing is wrapper for expand_grid(), deduplicates/sorts
  filter(end > start) %>% # just contiguous in the forward direction
  mutate(subset = map2(start, end, ~ candidates[.x:.y])) %>% # two arguments, slice into crossed option
  mutate(total = map_dbl(subset, ~ sum(.x))) %>% # sum the subset values
  filter(total == invalid_number) %>% # find row where total equals part1 answer
  mutate(smallest = map_dbl(subset, min),
         largest = map_dbl(subset, max)) %>% # get the smallest and largest values
  mutate(answer = smallest + largest) #sum them together

x
```

```
## # A tibble: 1 x 7
##   start   end subset         total smallest largest   answer
##   <int> <int> <list>         <dbl>    <dbl>   <dbl>    <dbl>
## 1   449   465 <dbl [17]> 105950735  4117189 9709726 13826915
```