

Merge Raw Pokemon Data

Lucas Jamar

2020-05-31

The steps for merging the data are as follow:

1. Read in raw data.
2. Duplicate and inverse battle data so that the data is larger and symmetric.
3. Of all the available pokemon, keep only the ones with price ≤ 3500 minus the price of 5 cheapest.
4. Battle remaining available pokemon against each opposing pokemon.
5. Get both types of player 1 and 2.
6. Melt weaknesses/ strength data and merge onto the main data using the types of the attacking/ defending player.
7. Write to CSV.

Read in all raw data

```
library(stringr)
library(data.table)

dt <- data.table::fread("data/01_raw/Battle_Results.csv")
all_pokemons <- data.table::fread("data/01_raw/All_Pokemons.csv")
available_pokemons <- data.table::fread("data/01_raw/AvailablePokemons.csv")
weakness_pokemon <- data.table::fread("data/01_raw/Weakness_Pokemon.csv")
submission <- data.table::fread("data/01_raw/Submission.csv")
```

Duplicate train data. Transform player 1 to player 2. Inverse battleresult. Combine with original data. Ensure no duplicates.

```
columns <- data.table::data.table(old = colnames(dt))
columns[, new := stringr::str_remove_all(old, "_1|_2")]
columns[old %like% "_1", new := paste0(new, "_2")]
columns[old %like% "_2", new := paste0(new, "_1")]
dt_copy <- data.table::copy(dt)
dt_copy[, BattleResult := -BattleResult]
data.table::setnames(dt_copy, columns$old, columns$new)
dt <- data.table::rbindlist(list(dt, dt_copy), use.names = TRUE)
rm(dt_copy)
dt <- unique(dt)
dt[, Set := "train"]
```

Keep available pokemons price ≤ 3500 - sum of 5 cheapest pokemons

```
max_price <- 3500
sum_cheapest_5 <- sum(sort(available_pokemons$Price_1)[1:5])
available_pokemons <- available_pokemons[Price_1 <= max_price - sum_cheapest_5]
```

Multiply available pokemons data and opponents data such that each available pokemon fights each opponent

```
available_length <- nrow(available_pokemons)
submission_length <- nrow(submission)
available_pokemons <- available_pokemons[rep(1:available_length, submission_length)]
submission <- submission[rep(1:submission_length, available_length)]
data.table::setorder(available_pokemons)
submission[, SelectedPokemonID := NULL]
submission <- cbind(available_pokemons, submission)
submission[, Set := "submission"]
```

Combine train data and submission data

```
dt <- data.table::rbindlist(list(dt, submission), fill = TRUE)
rm(submission, available_pokemons, columns)
```

Melt weakness data to merge onto total data

```
data.table::setnames(weakness_pokemon, "Types", "TypeAtk")
weakness_pokemon <- data.table::melt.data.table(weakness_pokemon,
  id.vars = "TypeAtk",
  variable.name = "TypeDef",
  value.name = "Weakness"
)
```

Find types for player 1 and 2

```
data.table::setnames(all_pokemons, c("Type_1", "Type_2"), c("Type_a", "Type_b"))
dt[all_pokemons, on = .(Name_1 = Name), c("Type_1a", "Type_1b") := mget(c("Type_a", "Type_b"))]
dt[all_pokemons, on = .(Name_2 = Name), c("Type_2a", "Type_2b") := mget(c("Type_a", "Type_b"))]
rm(all_pokemons)
```

Get strength of each type of 1 against each type of 2

```
dt[weakness_pokemon, on = .(Type_1a = TypeAtk, Type_2a = TypeDef), Strength_1a_2a := Weakness]
dt[weakness_pokemon, on = .(Type_1a = TypeAtk, Type_2b = TypeDef), Strength_1a_2b := Weakness]
dt[weakness_pokemon, on = .(Type_1b = TypeAtk, Type_2a = TypeDef), Strength_1b_2a := Weakness]
dt[weakness_pokemon, on = .(Type_1b = TypeAtk, Type_2b = TypeDef), Strength_1b_2b := Weakness]
```

Get strength of each type of 2 against each type of 1

```
dt[weakness_pokemon, on = .(Type_2a = TypeAtk, Type_1a = TypeDef), Strength_2a_1a := Weakness]
dt[weakness_pokemon, on = .(Type_2a = TypeAtk, Type_1b = TypeDef), Strength_2a_1b := Weakness]
dt[weakness_pokemon, on = .(Type_2b = TypeAtk, Type_1a = TypeDef), Strength_2b_1a := Weakness]
dt[weakness_pokemon, on = .(Type_2b = TypeAtk, Type_1b = TypeDef), Strength_2b_1b := Weakness]
rm(weakness_pokemon)
```

Write to CSV

```
data.table::fwrite(dt, "data/03_primary/pokemon.csv")
```