Tidy Nutri Data

Saghir & Andreia (ilustat)

2017-12-12 14:03:26

Start: 12 December 2017 (14:03:26)

Importing Nutri Data

Food composition data is publically available and provided by **Instituto Nacional de Saúde Dr. Ricardo Jorge** (portuguese "National Institute of Health" - INSA): http://portfir.insa.pt/foodcomp/introduction *It should not be used with any commercial intent.*

To download the data you can follow these steps: Access to "Composição dos alimentos" tab, then select 'Pesquisa de Alimentos' >> 'Download da TCA'

Save it in the folder `data` under the same working directory of the `nutrient_pt` project and read it as follows:

```
nutri_orig <- read_xlsx("data/insa_tca.xlsx")
#glimpse(nutri_orig)
```

Problem

Problem: Variable names of the the original data and the risk of incorrectly tidying the data.

When we look at the `nutri_long` dataset we see that we could lose the ordering of the `keyVars`. We need to protect again this causing a problem during the data tidying. The problem stems from the original data, looking at the names of `nutri_orig`:

```
names(nutri_orig)[1:12]
```

```
##  [1] "X__1"                    "Nome do alimento"
##  [3] "Grupo"                   "Energia [kcal] (ENERCC)"
##  [5] "X__2"                    "X__3"
##  [7] "Energia [kJ] (ENERCJ)"   "X__4"
##  [9] "X__5"                    "Lípidos (FAT)"
## [11] "X__6"                    "X__7"
```

Variable `Energia [kcal] (ENERCC)` contains the value for nutrient "Energia [kcal] (ENERCC)", `X__2` the associated unit, and `X__3` the associated quantity. It is the same for variable `Lípidos (FAT)` contains the value for nutrient "Lípidos (FAT)", `X__6` the associated unit, and `X__7` the associated quantity. This is the same pattern for all the nutrients as it originates from the original Excel spreadsheet where each nutrients has 3 columns but one *header* in a merged cell above the 3 columns.

Using the column names of the the original data we will create an ordering variable. First we will remove the the first three variables as they will remain *constants* as columns.

The we will create a grouping variable for the nutrients value, unit and quantity called `grpNtr` and `grpType`.

```
ordNames <- as_tibble(list(varName = names(nutri_orig)[-c(1, 2, 3)])) %>%
  mutate(grpNtr = ceiling(row_number()/3)) %>%
  mutate(typeVars = parse_number(varName) %% 2) %>%
```

```
  mutate(xVars = str_detect(varName, "^X__")) %>%
  mutate(grpType = if_else(typeVars == 1, "Quantity", "Unit")) %>%
  mutate(grpType = if_else(xVars == FALSE, "Nutrient", grpType)) %>%
  select(-xVars, -typeVars)
```

```
## Warning: 37 parsing failures.
## row # A tibble: 5 x 4 col     row   col expected                                    actual expected   <in
## ... ................. ... ....................................................... ........ .....
## See problems(...) for more details.
```

```
head(ordNames, n=12)
```

```
## # A tibble: 12 x 3
##                             varName grpNtr  grpType
##                               <chr>  <dbl>    <chr>
##  1         Energia [kcal] (ENERCC)      1 Nutrient
##  2                            X__2      1     Unit
##  3                            X__3      1 Quantity
##  4           Energia [kJ] (ENERCJ)      2 Nutrient
##  5                            X__4      2     Unit
##  6                            X__5      2 Quantity
##  7                  Lípidos (FAT)      3 Nutrient
##  8                            X__6      3     Unit
##  9                            X__7      3 Quantity
## 10 Ácidos gordos saturados (FASAT)      4 Nutrient
## 11                            X__8      4     Unit
## 12                            X__9      4 Quantity
```

Tidying Data Step 1

From the `nutri_orig` data we will create a long format

Now we will merge the group ordering to the `nutri_long` dataset

```
nutri_long <- nutri_orig %>%
  rename(foodID = X__1,
         foodItem = `Nome do alimento`,
         foodGroup = Grupo) %>%
  gather(key = "keyVars", value = "keyVals", -starts_with("food")) %>%
  select(foodID, foodGroup, foodItem, keyVars, keyVals) %>%
  left_join(ordNames, by= c("keyVars" = "varName"))

glimpse(nutri_long)
```

```
## Observations: 136,407
## Variables: 7
## $ foodID    <dbl> 619, 620, 802, 803, 703, 704, 646, 346, 345, 971, 97...
## $ foodGroup <chr> "Açúcar, confeitaria e sobremesas doces à base de ág...
## $ foodItem  <chr> "\"Donut\"", "\"Donut\" recheado com doce de fruta",...
## $ keyVars   <chr> "Energia [kcal] (ENERCC)", "Energia [kcal] (ENERCC)"...
## $ keyVals   <chr> "400", "348", "878", "900", "114", "293", "11", "78"...
## $ grpNtr    <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ grpType   <chr> "Nutrient", "Nutrient", "Nutrient", "Nutrient", "Nut...
```

Tidying Nutri Data

In the `nutri_long` dataset we have 136407 observations. Our final tidy dataset, which we call `nutri_tidy` should contain one third of these values (i.e., 45469).

```
# Prepare the Units and Quantity so that they can be merged to the Nutrient data.
nUnit <- nutri_long %>%
  filter(grpType == "Unit") %>%
  select(foodID, grpNtr, Unit = keyVals)

head(nUnit)
```

```
## # A tibble: 6 x 3
##   foodID grpNtr         Unit
##    <dbl>  <dbl>        <chr>
## 1    619      1 quilocaloria
## 2    620      1 quilocaloria
## 3    802      1 quilocaloria
## 4    803      1 quilocaloria
## 5    703      1 quilocaloria
## 6    704      1 quilocaloria
```

```
nQty <-  nutri_long %>%
  filter(grpType == "Quantity") %>%
  select(foodID, grpNtr, Quantity = keyVals)

head(nQty)
```

```
## # A tibble: 6 x 3
##   foodID grpNtr                  Quantity
##    <dbl>  <dbl>                     <chr>
## 1    619      1 por 100 g de parte edível
## 2    620      1 por 100 g de parte edível
## 3    802      1 por 100 g de parte edível
## 4    803      1 por 100 g de parte edível
## 5    703      1 por 100 g de parte edível
## 6    704      1 por 100 g de parte edível
```

```
# Prepare the base for the Nutrition data by keeping only the nutrient and values
nutri_tidy <- nutri_long %>%
  filter(grpType == "Nutrient") %>%
  rename(Value = keyVals,
         Nutrient = keyVars) %>%
  left_join(nUnit, by = c("foodID", "grpNtr")) %>%
  left_join(nQty, by = c("foodID", "grpNtr")) %>%
  # select(-grpNtr, -grpType) %>%
  mutate(NutrientCode = gsub(".*\\((.*)\\).*", "\\1", Nutrient),
         NutrientID = group_indices(. , Nutrient)) %>%
  mutate(Value = as.numeric(Value)) %>%
  select(foodID, foodGroup, foodItem, NutrientID, Nutrient, NutrientCode, Value, Unit, Quantity) %>%
  arrange(foodID, Nutrient)

  glimpse(nutri_tidy)
```

```
## Observations: 45,469
## Variables: 9
```

```
## $ foodID       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ foodGroup    <chr> "Leite e produtos lácteos", "Leite e produtos lác...
## $ foodItem     <chr> "Leite de cabra cru", "Leite de cabra cru", "Leit...
## $ NutrientID   <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15...
## $ Nutrient     <chr> "a-tocoferol (TOCPHA)", "Ácido linoleico (F18:2CN...
## $ NutrientCode <chr> "TOCPHA", "F18:2CN6", "FAMS", "FAPU", "FASAT", "F...
## $ Value        <dbl> 0.03, 0.10, 1.10, 0.10, 2.60, 0.10, 0.00, 86.90, ...
## $ Unit         <chr> "milligrama", "grama", "grama", "grama", "grama",...
## $ Quantity     <chr> "por 100 g de parte edível", "por 100 g de parte ...
```

Cleaning Data

Data is tidy but not totally cleaned yet

```
nutri_tidy %>% arrange(foodGroup, foodItem) %>% select(foodItem)
```

```
## # A tibble: 45,469 x 1
##       foodItem
##          <chr>
##  1 "\"Donut\""
##  2 "\"Donut\""
##  3 "\"Donut\""
##  4 "\"Donut\""
##  5 "\"Donut\""
##  6 "\"Donut\""
##  7 "\"Donut\""
##  8 "\"Donut\""
##  9 "\"Donut\""
## 10 "\"Donut\""
## # ... with 45,459 more rows
```

We use `stringr` to clean up the values from `foodItem` where we can still find quotation marks (“”)

```
nutri_clean <- nutri_tidy %>%
        # clean observations
        mutate(foodItem = str_replace_all(
                foodItem, "\\p{quotation mark}", ""),
               Nutrient = str_replace_all(
                       Nutrient, "\\+", ""),
               # remove square brackets to facilitate filtering detection later
               Nutrient = str_replace_all(
                       Nutrient, "\\s[\\[\\]]", "_"),
               Nutrient = str_replace_all(
                       Nutrient, "[\\]]", "")
               ) %>%
        arrange(foodGroup, foodItem)
# DT::datatable(nutri_clean)
```

Now when we search for “Donut” on the search window we no longer see it with quotation marks.

Save the cleaned dataset to use it later in the Shiny app

```
save(nutri_clean, file = "data/nutri_clean.RData")
```

```
#load("nutri_clean.RData")
```

Get data into wide format for food items

```r
nutri_wide <- nutri_clean %>%
        select(foodID, foodItem, NutrientID, Nutrient, Value, Unit, -Quantity) %>%
        group_by(foodItem) %>%
        select(-foodID) %>%
        spread(foodItem, Value)

head(nutri_wide)
```

```
## # A tibble: 6 x 1,112
##   NutrientID                              Nutrient       Unit Abacate
##        <int>                                 <chr>      <chr>   <dbl>
## 1          1                 a-tocoferol (TOCPHA) milligrama     2.1
## 2          2           Ácido linoleico (F18:2CN6)      grama     1.1
## 3          3 Ácidos gordos monoinsaturados (FAMS)      grama     6.5
## 4          4  Ácidos gordos polinsaturados (FAPU)      grama     1.2
## 5          5      Ácidos gordos saturados (FASAT)      grama     2.2
## 6          6          Ácidos gordos trans (FATRS)      grama     0.0
## # ... with 1108 more variables: `Abóbora cristalizada` <dbl>, `Abóbora
## #   crua` <dbl>, `Abrótea cozida` <dbl>, `Abrótea crua` <dbl>,
## #   Açorda <dbl>, `Açorda à alentejana` <dbl>, `Açorda de bacalhau` <dbl>,
## #   `Açorda de marisco` <dbl>, `Açorda de ovo` <dbl>, `Açúcar
## #   amarelo` <dbl>, `Açúcar branco` <dbl>, `Agrião cru` <dbl>, `Água
## #   mineral natural gaseificada, Pizões-Moura` <dbl>, `Água mineral
## #   natural gaseificada, Vimeiro` <dbl>, `Água mineral natural
## #   gasocarbónica, Pedras Salgadas` <dbl>, `Água mineral natural,
## #   Luso` <dbl>, `Água, rede pública de abastecimento (Lisboa)` <dbl>,
## #   Aguardente <dbl>, `Aipo cru` <dbl>, `Alcachofra cozida` <dbl>,
## #   `Alcachofra crua` <dbl>, `Alface crua` <dbl>, `Alheira cozida sem
## #   adição de sal` <dbl>, `Alheira crua` <dbl>, `Alheira grelhada sem
## #   adição de sal` <dbl>, `Alho cru` <dbl>, `Alho em pó` <dbl>, `Alho
## #   francês cru` <dbl>, `Almôndega cozinhada` <dbl>, `Almôndegas de carne
## #   de vaca` <dbl>, `Almôndegas saloias` <dbl>, `Amêijoa aberta ao natural
## #   sem sal` <dbl>, `Amêijoa crua` <dbl>, Amêijoas <dbl>, `Ameixa
## #   branca` <dbl>, `Ameixa encarnada` <dbl>, `Ameixa rainha
## #   Cláudia` <dbl>, `Ameixa seca` <dbl>, `Ameixa, conserva em calda de
## #   açúcar` <dbl>, `Amêndoa, miolo, com pele` <dbl>, `Amêndoa, miolo,
## #   torrada, sem pele` <dbl>, `Amendoim, miolo` <dbl>, `Amendoim, miolo,
## #   torrado com sal` <dbl>, `Amendoim, miolo, torrado sem sal` <dbl>,
## #   Ananás <dbl>, `Ananás, conserva em calda de açúcar` <dbl>,
## #   Anona <dbl>, `Arroz à valenciana` <dbl>, `Arroz carolino branqueado
## #   cru` <dbl>, `Arroz com refogado` <dbl>, `Arroz comum cru` <dbl>,
## #   `Arroz cozido simples` <dbl>, `Arroz de bacalhau` <dbl>, `Arroz de
## #   bacalhau cpm margarina` <dbl>, `Arroz de cabidela` <dbl>, `Arroz de
## #   cenoura` <dbl>, `Arroz de cenoura com azeite` <dbl>, `Arroz de
## #   ervilhas` <dbl>, `Arroz de feijão` <dbl>, `Arroz de frango` <dbl>,
## #   `Arroz de frango com feijão e chouriço` <dbl>, `Arroz de frango
## #   malandrinho à moda de Monção` <dbl>, `Arroz de gambas` <dbl>, `Arroz
## #   de lulas` <dbl>, `Arroz de manteiga` <dbl>, `Arroz de marisco` <dbl>,
## #   `Arroz de pato` <dbl>, `Arroz de peixe` <dbl>, `Arroz de peixe com
## #   ervilhas` <dbl>, `Arroz de polvo com azeite` <dbl>, `Arroz de polvo
## #   com tomate` <dbl>, `Arroz de polvo com tomate e vinho` <dbl>, `Arroz
## #   de tamboril` <dbl>, `Arroz de tamboril malandrinho` <dbl>, `Arroz de
## #   tomate com azeite` <dbl>, `Arroz de tomate com margarina` <dbl>,
## #   `Arroz de tomate malandrinho` <dbl>, `Arroz doce` <dbl>, `Arroz
```

```
## #   integral cru` <dbl>, `Atum conserva em óleo` <dbl>, `Atum de
## #   cebolada` <dbl>, `Atum fresco cru` <dbl>, `Atum fresco
## #   grelhado` <dbl>, `Avelã, miolo` <dbl>, `Azeite (4 marcas)` <dbl>,
## #   Azeitona <dbl>, `Bacalhau à Brás` <dbl>, `Bacalhau à Brás com
## #   azeite` <dbl>, `Bacalhau à Brás com azeite e azeitonas` <dbl>,
## #   `Bacalhau à Gomes de Sá` <dbl>, `Bacalhau à Gomes de Sá, com
## #   azeite` <dbl>, `Bacalhau assado no forno com azeite` <dbl>, `Bacalhau
## #   com natas` <dbl>, `Bacalhau com natas, com queijo ralado` <dbl>,
## #   `Bacalhau cozido` <dbl>, `Bacalhau fresco cozido` <dbl>, `Bacalhau
## #   fresco cru` <dbl>, `Bacalhau grelhado` <dbl>, `Bacalhau seco e
## #   salgado, demolhado cru` <dbl>, Bacon <dbl>, ...
```

Save the wide version to use it later in the Shiny app

```
save(nutri_wide, file = "data/nutri_wide.RData")
```

Selecting/Matching specific variables

```
x <- c("Abacate", "Abóbora", "Açorda")

nutri_wide %>% select(Nutrient, str_subset(names(.), x)) %>% head()
```

```
## Warning in stri_subset_regex(string, pattern, omit_na = TRUE, opts_regex
## = opts(pattern)): longer object length is not a multiple of shorter object
## length
```

```
## # A tibble: 6 x 5
##                               Nutrient Abacate `Abóbora cristalizada`
##                                  <chr>   <dbl>                  <dbl>
## 1                a-tocoferol (TOCPHA)     2.1                    0.0
## 2           Ácido linoleico (F18:2CN6)     1.1                    0.0
## 3 Ácidos gordos monoinsaturados (FAMS)     6.5                    0.0
## 4  Ácidos gordos polinsaturados (FAPU)     1.2                    0.0
## 5      Ácidos gordos saturados (FASAT)     2.2                    0.1
## 6          Ácidos gordos trans (FATRS)     0.0                    0.0
## # ... with 2 more variables: Açorda <dbl>, `Açorda de marisco` <dbl>
```

Unite variables unit with Food to enable later spread of nutrient values into individual variables

```
nutri_new <- nutri_clean %>%
        mutate(Quantity_unit = str_detect(Quantity, "g"),
               Quantity_unit = ifelse(Quantity_unit == TRUE, "g", "mL"),
               Quantity = str_replace_all(Quantity, "[^[\\d]]", ""),
               Value = sprintf("%6.f", Value),
               Value = as.numeric(Value),
               Quantity = as.numeric(Quantity)) %>%
        group_by(foodItem) %>%
        unite(Food, foodItem, Quantity_unit, sep = " (") %>%
        mutate(Food = str_c(Food, ")")) %>%
        select(FoodID = foodID, Food, FoodGroup = foodGroup, Quantity, NutrientID, Nutrient, Unit, Valu
```

```
## Warning in evalq(as.numeric(Value), <environment>): NAs introduced by
## coercion
```

```
head(nutri_new)
```

```
## # A tibble: 6 x 8
##   FoodID               Food
```

```
##    <dbl>              <chr>
## 1    804 Açúcar amarelo (g)
## 2    804 Açúcar amarelo (g)
## 3    804 Açúcar amarelo (g)
## 4    804 Açúcar amarelo (g)
## 5    804 Açúcar amarelo (g)
## 6    804 Açúcar amarelo (g)
## # ... with 6 more variables: FoodGroup <chr>, Quantity <dbl>,
## #   NutrientID <int>, Nutrient <chr>, Unit <chr>, Value <dbl>
```

```
save(nutri_new, file = "data/nutri_new.RData", envir = .GlobalEnv)
```

Example of using `nutri_new` in Shiny for user input represented by `y`

```
y <- "Energia_kcal (ENERCC)"
nutri_comp <- nutri_new %>%
                        select(Food, Quantity, Nutrient, Unit, Value) %>%
                        unite(Nutrient, Nutrient, Unit, sep = " (") %>%
                        mutate(Nutrient = str_c(Nutrient, ")")) %>%
                        spread(Nutrient, Value) %>%
                        select(Food, Quantity,
                               str_subset(names(.),
                                          str_c(str_match(y,
                                                          "^[\\w-\\w+\\s]+"),
                                                          collapse = "|")))
head(nutri_comp)
```

```
## # A tibble: 6 x 3
##                       Food Quantity `Energia_kcal (ENERCC) (quilocaloria)`
##                      <chr>    <dbl>                                  <dbl>
## 1              Abacate (g)      100                                    114
## 2 Abóbora cristalizada (g)      100                                    293
## 3         Abóbora crua (g)      100                                     11
## 4       Abrótea cozida (g)      100                                     78
## 5         Abrótea crua (g)      100                                     70
## 6               Açorda (g)      100                                    103
```

Generate other dataset versions to work more easily with Shiny

```
# New  dataset to show Nutrient with units
nutri_choice <- nutri_new %>%
        unite(Nutrient, Nutrient, Unit, sep = " (") %>%
        mutate(Nutrient = str_c(Nutrient, ")"))

save(nutri_choice, file = "data/nutri_choice.RData", envir = .GlobalEnv)

## New dataset to show  less food observations (reduce memory usage in the ui selectInput)
food_wide <- nutri_choice %>%
        select(FoodID, Food, Nutrient, Value) %>%
        spread(Nutrient, Value)

save(food_wide, file = "data/food_wide.RData", envir = .GlobalEnv)
```

End: 12 December 2017 (14:03:30)
