Ireland Notebook

DATACLEANING IRELAND

Load the libraries

```
# lib = where to save the library  (zonder = default)

package_list <- c("data.table", "tidyverse", "naniar", "stringr", "readr",  "dplyr", "magrittr", "readx
                  )

for (pkg in package_list) {
  if (pkg %in% rownames(.packages()) == FALSE)
  {library(pkg, character.only = TRUE)}

}
```

##Function to apply SHA-256 hashing

```
sha256_hash <- function(data) {
  openssl::sha256(data)
}
```

Data loading

```
barometer_dt_raw_2021 <- readxl::read_excel("../Data/Ireland/Jade_2021_Final_Anonymised_data_Only_2023-
barometer_dt_raw_2022 <- readxl::read_excel("../Data/Ireland/Jade_2022_Final_Anonymised_data_Only_2023-
```

Add new dataset

```
barometer_dt_combined <- rbind(barometer_dt_raw_2021, barometer_dt_raw_2022)
```

Filter data

```
barometer_dt_filter <- barometer_dt_combined %>%
  dplyr::filter(SYSTEM %in% c('Respiratory', 'NA')
                )

barometer_dt_filter2 <- barometer_dt_filter %>%
  dplyr::filter(ALIQUOTMATRIXTYPE %in% c('Pleural Fluid', 'Tissue swab', 'Tonsil',  'Lymph Node - Multi
                )
barometer_dt_filter3 <- barometer_dt_filter2 %>%
  dplyr::filter(TEST %in% c("PI3V PCR", "PCR M. haemolytica - ARVL", "Maldi ToF",
                            "Mycoplasma bovis (PCR)", "PCR H. somni - ARVL",
                            "PCR P. multocida - ARVL", "Miscellaneous Test",
                            "Routine Culture", "PCR M. bovis - ARVL", "BRSV PCR",
                            "Culture Growth", "Next Generation Sequencing",
```

```
                              "PCR BoCoV", "Mycoplasma bovis (PCR)")
)
```

Data manipulation

```
barometer_dt <- barometer_dt_filter3 %>%
  dplyr::rename(
    Filenumber = SDGa,
    Samplenumber = SAMPLEa,
    Farm_ID = HERD_NOa,
    Date = DELIVERY_DATE,
    breed = Herd.Type
    ) %>%
  dplyr::mutate(
    Country ='Ireland',
    Lab_reference ='5',
    Sample_type = case_when(
      SUBCLASS == 'Carcass' ~ 'Autopsy',
      ALIQUOTMATRIXTYPE %in% c('Carcass', 'Lung', 'Thymus',
                               'Lymph Node - Multiple', 'Tissue-Pool',
                               'Lymph Node', 'Tissue (VTM)',
                               'Part Carcass') ~ 'Autopsy',
      ALIQUOTMATRIXTYPE %in% c('Swab', 'Nasal Swab', 'Pooled swab',
                               'Nasal Fluid') ~ 'Swab',
      ALIQUOTMATRIXTYPE %in% c('Trachea', 'Thoracic Fluid', 'Culture',
                               'Fluid', 'Misc.', 'Pleural Fluid') ~ 'Unknown',
      TRUE ~ 'Missing'
    ),
    Diagnostic_test = case_when(
      TEST %in% c("PI3V PCR", "PCR M. haemolytica - ARVL",
                  "Mycoplasma bovis (PCR)", "PCR H. somni - ARVL",
                  "PCR M. bovis - ARVL", "BRSV PCR", "PCR BoCoV",
                  "Mycoplasma bovis (PCR)", "PCR P. multocida - ARVL") ~ 'PCR',
      TEST %in% c("Routine Culture", "Culture Growth") ~ "Culture",
      TEST == 'Maldi ToF' ~ 'MALDI-TOF',
      TEST == "Next Generation Sequencing" ~ 'NGS',
      TEST == "Miscellaneous Test" ~ 'Unknown',
      TRUE ~ 'Missing'
    ),
    Breed = case_when(
      breed == "BEEF" ~ 'Beef',  ## could add 'SUCKLER' here??
      breed == "DAIRY" ~'Dairy',
      breed %in% c("OTHER") ~ 'Unknown',
      TRUE ~ 'Unknown'
    ),
    Province = case_when(     ## data per county or region??
      Region == "LEINSTER" ~ 'Leinster',
      Region == "MUNSTER" ~'Munster',
      Region == "ULSTER" ~'Ulster',
      Region == "CONNAUGHT" ~ 'Connaught',
      Region == "Unavailable/Incomplete" ~ 'Unknown',
      TRUE ~ 'Missing'
    ),
```

```
    Pathogen = case_when(
      TEST == "PCR P. multocida - ARVL" ~ 'PM',
      TEST == "PCR M. haemolytica - ARVL" ~ 'MH',
      TEST == "PCR H. somni - ARVL" ~ 'HS',
      TEST %in% c("Mycoplasma bovis (PCR)", "PCR M. bovis - ARVL") ~ 'MB',
      TEST == "PI3V PCR" ~ 'PI3',
      TEST == "PCR BoCoV" ~ 'BCV',
      TEST == "BRSV PCR" ~ 'BRSV',
      TRUE ~ 'Missing'
    ),
  )%>%
  dplyr::select(
    Filenumber,
    Samplenumber,
    Diagnostic_test,
    Country,
    Lab_reference,
    Sample_type,
    Breed,
    Pathogen,
    Date,
    Province,
    RESULT,
    RESULTNAME,
    AGENT,
    Farm_ID
  ) %>%
  dplyr::distinct() %>%
  dplyr::mutate(
    Filenumber = sha256_hash(as.character(Filenumber)),
    Samplenumber = sha256_hash(as.character(Samplenumber))
    )
```

Toevoegen extra rijen voor cultuur (& MALDI & NGS?)

```
barometer_dt$HS <- ifelse(barometer_dt$Diagnostic_test == "Culture", 0, NA)
barometer_dt$MH <- ifelse(barometer_dt$Diagnostic_test == "Culture", 0, NA)
barometer_dt$PM <- ifelse(barometer_dt$Diagnostic_test == "Culture", 0, NA)

barometer_dt_culture_wide <- barometer_dt %>%
  tidyr::pivot_longer(
    cols = c('PM', 'MH', 'HS'),
    names_to = 'Pathogen_culture',
    values_to = 'Result_culture'
  )

barometer_dt_culture_wide$Pathogen <-
  ifelse(barometer_dt_culture_wide$Pathogen == "Missing",
         barometer_dt_culture_wide$Pathogen_culture,
         barometer_dt_culture_wide$Pathogen)

## nu kan result-culture overruled worden door iets van RESULT/AGENT/etc.
```

```
barometer_dt <- barometer_dt_filter3 %>%
  dplyr::rename(
    Filenumber = SDGa,
    Samplenumber = SAMPLEa,
    Farm_ID = HERD_NOa,
    Date = DELIVERY_DATE,
    breed = Herd.Type
    ) %>%
  dplyr::mutate(
    Country ='Ireland',
    Lab_reference ='5',
    Sample_type = case_when(
      SUBCLASS == 'Carcass' ~ 'Autopsy',
      ALIQUOTMATRIXTYPE %in% c('Carcass', 'Lung', 'Thymus',
                               'Lymph Node - Multiple', 'Tissue-Pool',
                               'Lymph Node', 'Tissue (VTM)',
                               'Part Carcass') ~ 'Autopsy',
      ALIQUOTMATRIXTYPE %in% c('Swab', 'Nasal Swab', 'Pooled swab',
                               'Nasal Fluid') ~ 'Swab',
      ALIQUOTMATRIXTYPE %in% c('Trachea', 'Thoracic Fluid', 'Culture',
                               'Fluid', 'Misc.', 'Pleural Fluid') ~ 'Unknown',
      TRUE ~ 'Missing'
    ),
    Diagnostic_test = case_when(
      TEST %in% c("PI3V PCR", "PCR M. haemolytica - ARVL",
                  "Mycoplasma bovis (PCR)", "PCR H. somni - ARVL",
                  "PCR M. bovis - ARVL", "BRSV PCR", "PCR BoCoV",
                  "Mycoplasma bovis (PCR)", "PCR P. multocida - ARVL") ~ 'PCR',
      TEST %in% c("Routine Culture", "Culture Growth") ~ "Culture",
      TEST == 'Maldi ToF' ~ 'MALDI-TOF',
      TEST == "Next Generation Sequencing" ~ 'NGS',
      TEST == "Miscellaneous Test" ~ 'Unknown',
      TRUE ~ 'Missing'
    ),
    Breed = case_when(
      breed == "BEEF" ~ 'Beef',  ## could add 'SUCKLER' here??
      breed == "DAIRY" ~'Dairy',
      breed %in% c("OTHER") ~ 'Unknown',
      TRUE ~ 'Unknown'
    ),
    Province = case_when(     ## data per county or region??
      Region == "LEINSTER" ~ 'Leinster',
      Region == "MUNSTER" ~'Munster',
      Region == "ULSTER" ~'Ulster',
      Region == "CONNAUGHT" ~ 'Connaught',
      Region == "Unavailable/Incomplete" ~ 'Unknown',
      TRUE ~ 'Missing'
    ),
    Pathogen = case_when(
      TEST == "PCR P. multocida - ARVL" ~ 'PM',
      TEST == "PCR M. haemolytica - ARVL" ~ 'MH',
      TEST == "PCR H. somni - ARVL" ~ 'HS',
      TEST %in% c("Mycoplasma bovis (PCR)", "PCR M. bovis - ARVL") ~ 'MB',
```

```
      TEST == "PI3V PCR" ~ 'PI3',
      TEST == "PCR BoCoV" ~ 'BCV',
      TEST == "BRSV PCR" ~ 'BRSV',
      TRUE ~ 'Missing'
    ),
    Result = case_when(
      RESULT %in% c("Positive", "Weak Positive", "Mycoplasma bovis PCR Positive",
                    "Strong Positive") ~ 1,
      RESULT %in% c("No Pathogen detected", "Negative", "sterile",
                    "No Significant Growth", "No CT",
                    "Mycoplasma bovis PCR Negative",
                    "Mixed Non-Significant Bacterial Growth",
                    "No Significant Growth @48hrs", "No Growth",
                    "No Pathogen detectedn", "No RNA detected",
                    "No DNA detected") ~ 0,
      RESULT %in% c("Inconclusive", "Mixed Bacterial Growth", "Mixed Growth",
                      "Very Mixed Growth") ~ as.numeric(NA),
      Diagnostic_test == 'Culture' & is.na(RESULT) ~ 0,
      Diagnostic_test == 'Culture' & !is.na(RESULT) ~ 1,
      Diagnostic_test == 'MALDI-TOF' & is.na(RESULT) ~ 0,
      Diagnostic_test == 'MALDI-TOF' & (RESULT > 1.7) ~ 1,
      Diagnostic_test == 'NGS' & is.na(AGENT) ~ 0,
      Diagnostic_test == 'NGS' & !is.na(RESULT) ~ 1,
      TRUE ~ as.numeric(NA)
    )
  )%>%
  dplyr::select(
    Filenumber,
    Samplenumber,
    Diagnostic_test,
    Country,
    Lab_reference,
    Sample_type,
    Breed,
    Pathogen,
    Result,
    Date,
    Province,
    Farm_ID
  ) %>%
  dplyr::distinct()%>%

  dplyr::mutate(
    Filenumber = sha256_hash(as.character(Filenumber)),
    Samplenumber = sha256_hash(as.character(Samplenumber)),
    Farm_ID = sha256_hash(as.character(Farm_ID))
    )
```

Stuk van die diagnostic tests moet nog anders worden geschreven..

Diagnostic_test == 'Culture' && RESULT == "Pasteurella multocida" ||

"Mannheimia haemolytica" || "Histophilus somnus" || "Histophilus somni" ||

"Histophilus somnii",

multiply row ??

```
df <- data.frame(barometer_dt, Diagnostic_test = 'Culture')

new_df <- rbind(rep(df, times = 4))

new_df$Pathogen <- c('PM', 'MH', 'HS', 'MB')
```

Intermediate table for culture/NGS results ??

```
df_samples_culture <- data.frame(
    Result = c('Missing', 'Missing', 'Missing', 'Missing'),
    Diagnostic_test = c('Culture','Culture','Culture', 'Culture'),
    Pathogen = c("PM","MH","HS", "MB")
)

df_samples_NGS <- data.frame(
  Result = c('Missing', 'Missing', 'Missing', 'Missing'),
  Diagnostic_test = c('NGS','NGS','NGS', 'NGS'),
  Pathogen = c("PM","MH","HS", "MB")
)

barometer_dt2 <- barometer_dt %>%
  dplyr::left_join(df_samples_culture, by = c('Diagnostic_test','Pathogen')) %>%
  dplyr::left_join(df_samples_NGS, by = c('Diagnostic_test', 'Pathogen')

)
```

dplyr::filter(

Pathogen_identification %in% c(“Pasteurella multocida”,“Mannheimia haemolytica”,“Histophilus somni”,“Mycoplasma bovis”)

RESULT? ## ““Maldi ToF”, ““,”Miscellaneous Test”, “Routine Culture”

“Culture Growth”, “Next Generation Sequencing”,

Floor date to 1st of month

```
barometer_dt$Floored_date <- lubridate::floor_date(barometer_dt$Date, "month")
```

Aggregate data based on farm_ID & month

```
barometer_groupby <- barometer_dt %>%
  group_by(Lab_reference, Country, Breed, Floored_date, Province,
           Farm_ID, Diagnostic_test, Sample_type, Pathogen) %>%
  summarise(across(c(Result), max), .groups = "drop")
```

Save file (long version)

```
write.csv(barometer_groupby, "../Data/CleanedData/barometer_Ireland.csv", row.names=TRUE)
```

Convert to wide version

```
barometer_wide <- barometer_groupby %>%
  tidyr::pivot_wider(names_from = c(Pathogen), values_from = Result)
```