PathoSense Notebook

DATACLEANING PATHOSENSE

Load the libraries

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

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

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 <- read.csv("../Data/PathoSense/AllBovineRespiratory_NegativesIncluded.csv")
```

Adding of rows for pathogens

```
barometer_dt <- barometer_dt_raw %>%
  dplyr::mutate(
    HS = ifelse(str_detect(pathogens, "Histophilus somni"), 1, 0),
    MH = ifelse(str_detect(pathogens, "Mannheimia haemolytica"), 1, 0),
    PM = ifelse(str_detect(pathogens, "Pasteurella multocida"), 1, 0),
    BCV = ifelse(str_detect(pathogens, "Bovine coronavirus"), 1, 0),
    MB = ifelse(str_detect(pathogens, "Mycoplasmopsis bovis"), 1, 0),
    PI3 = ifelse(str_detect(pathogens, "Bovine respirovirus 3"), 1, 0),
    BRSV = ifelse(str_detect(pathogens, "Bovine orthopneumovirus"), 1, 0),
  )
```

Data manipulation

```
barometer_dt <- barometer_dt_raw %>%
  dplyr::rename(
    Filenumber = sample_id,
    #Samplenumber = sample_id,
    Farm_ID = farm_id,
    #Project = project,
    Date = created
    ) %>%
  dplyr::mutate(
    Lab_reference ='4',
    Diagnostic_test = 'NPS',
    Breed = 'Unknown',
    Province = NA,
    Country = case_when(
      country == 'BE' ~ 'Belgium',
      country == 'NL' ~ 'The Netherlands'
    ),
    Sample_type = case_when(
      type == 'balFluid' ~ 'BAL',
      type == 'noseSwab' ~ 'Swab',
      TRUE ~ 'Other'
    ),
    HS = ifelse(str_detect(pathogens, "Histophilus somni"), 1, 0),
    MH = ifelse(str_detect(pathogens, "Mannheimia haemolytica"), 1, 0),
    PM = ifelse(str_detect(pathogens, "Pasteurella multocida"), 1, 0),
    BCV = ifelse(str_detect(pathogens, "Bovine coronavirus"), 1, 0),
    MB = ifelse(str_detect(pathogens, "Mycoplasmopsis bovis"), 1, 0),
    PI3 = ifelse(str_detect(pathogens, "Bovine respirovirus 3"), 1, 0),
    BRSV = ifelse(str_detect(pathogens, "Bovine orthopneumovirus"), 1, 0)

  )%>%
  dplyr::select(
    Filenumber,
    Lab_reference,
    Country,
    Breed,
    Province,
    Farm_ID,
    Diagnostic_test,
    Sample_type,
    PM,
    MH,
    HS,
    MB,
    BRSV,
    PI3,
    BCV,
    Date
  ) %>%
  dplyr::distinct()%>%
dplyr::mutate(
    Filenumber = sha256_hash(as.character(Filenumber)),
    Farm_ID = sha256_hash(as.character(Farm_ID))
    )
```

Floor date to 1st of month

```
barometer_dt$Date <- lubridate::ymd_hms(barometer_dt$Date)
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
  summarise(across(c(PM, MH, HS, MB, BRSV, PI3, BCV), max))

# If all are NA, than NA, if not (else): max in group, while ignoring NA
```

Convert to long

```
barometer_long <- barometer_groupby %>%
  tidyr::pivot_longer(
    cols = c('PM', 'MH', 'HS', 'MB', 'BRSV', 'PI3', 'BCV'),
    names_to = 'Pathogen',
    values_to = 'Result',
  )
```

Save file (long version)

```
write.csv(barometer_long, "../Data/CleanedData/barometer_PathoSense.csv", row.names=TRUE)
```