

# Resistenzen.Rmd : Vorbereitung Daten

12.03.2022

## Bibliotheken laden, Hilfsfunktion

```
library(xlsx)           # Um Excel files einzulesen
library(stringr)        # String-verarbeitung

debug <- F              # kein debug printout
debug <- T              # debug printout
Log <- function(string) {
  if(debug){print(string)}
}
```

## Die 2 Excel-Files einlesen

```
# alle Spalten einlesen, insb. Farm ID, WM group. Farm 30 fehlt, aber ich rechne nie mit Zeilennummern :
codes <- read.xlsx("coded_data_questionnaire_pilot12.xlsx", sheetName="data")

codes <- head(codes,59) # die letzten 2 Zeilen sind Quatsch
#View(codes)

Resistenzen <- read.xlsx("MIC_E. coli environment_L Windhofer_final.xls", sheetName="Abfrage")[6:22] # wichtige
Resistenzen[2] <- NULL # die ist nicht wichtig
#View(Resistenzen)    # 240 Zeilen
```

## Farm IDs extrahieren:

```
col1_ <- str_replace(Resistenzen[[1]], "-", "") # evtl. Bindestrich weg

Resistenzen[[1]] <- substr(col1_, 1,nchar(col1_)-2) # dann sind die letzten 2 Zeichen überflüssig
names(Resistenzen)[1] <- "Farm.ID" # Diese Spalte enthält jetzt nur noch die Farm IDs

#View(Resistenzen) # 240 Zeilen
```

## Farm 30 ausschliessen

```
Resistenzen <- Resistenzen[Resistenzen["Farm.ID"] != 30,]
ResRow <- nrow(Resistenzen)
#View(Resistenzen) # 236 = 240 - 4 Proben der Farm 30
#Resistenzen[116,] Farm 29
#Resistenzen[117,] Farm 31
```

## Spalten für die unabhängigen Variablen anfügen:

Abkürzung	Bedeutung	Variablentyp	Code/Werte	Code
WM	Waste Milk	binär	1=Waste Milk	2=No Waste Milk

Abkürzung	Bedeutung	Variablentyp	Code/Werte	Code
OLS	Q9 Other LiveStock	binär	0=No	1=Yes
IAC	Q12 Ill Animals in Calving box	binär, viele NA	0=No	1=Yes
HSC	Q20 Husbandry System Calves	6-wertig nominal	0=stable w\o outlet 2=outdoors 4=1+2	1=stable w\ outlet 3=0+1 5=0+2
MY	Q6 meanMY/cow	numerisch		
SCC	Q7 mean SCC/11mo	numerisch		
CBC	Q13a calvingbox_clean	numerisch, viele NA		
DIA	Q17 IN_diarrhea<30d	6-wertig ordinal	0-5	

*# Start mit leeren Spalten:*

```
Resistenzen["WM.group" ] <- vector(mode="character", length=ResRow)
Resistenzen["OLS.group"] <- vector(mode="character", length=ResRow)
Resistenzen["IAC.group"] <- vector(mode="character", length=ResRow)
```

*### Neue binäre hier dazufügen ###*

```
Resistenzen["HSC.group"] <- vector(mode="character", length=ResRow)
```

```
Resistenzen["MY.group" ] <- vector(mode="character", length=ResRow)
Resistenzen["SCC.group"] <- vector(mode="character", length=ResRow)
Resistenzen["CBC.group"] <- vector(mode="character", length=ResRow)
```

```
Resistenzen["DIA.group"] <- vector(mode="character", length=ResRow)
```

```
for (i in c(1:ResRow)) {                                # Schleife über alle Einträge
  Farm_ID <- Resistenzen[i,"Farm.ID"]
```

```
  Resistenzen[i,"WM.group" ] <- codes[codes["Farm.ID"] == Farm_ID,"WM.group" ]
  Resistenzen[i,"OLS.group"] <- codes[codes["Farm.ID"] == Farm_ID,"Q9.other_livestock" ]
  Resistenzen[i,"IAC.group"] <- codes[codes["Farm.ID"] == Farm_ID,"Q12.illanimals_in_calvingbox" ]
  ### Neue binäre hier dazufügen ###
```

```
  Resistenzen[i,"HSC.group"] <- codes[codes["Farm.ID"] == Farm_ID,"Q20.husbandry_system_calves" ]
  Resistenzen[i,"MY.group" ] <- codes[codes["Farm.ID"] == Farm_ID,"Q6.meanMY.cow" ]
  Resistenzen[i,"SCC.group"] <- codes[codes["Farm.ID"] == Farm_ID,"Q7.mean.SCC.11mo" ]
  Resistenzen[i,"CBC.group"] <- codes[codes["Farm.ID"] == Farm_ID,"Q13a.calvingbox_clean" ]
```

```
  Resistenzen[i,"DIA.group"] <- codes[codes["Farm.ID"] == Farm_ID,"Q17.IN_diarrhea.30d" ]
```

```
}
```

```
if(debug){
  View(Resistenzen)      # 240 Zeilen (ohne Farm 30)
}
```

## Resistenzen.xlsx rausschreiben

```
write.csv(Resistenzen,"Resistenzen.csv")
```