# takes an input and converts the PR-0000000000001-type IDs into patient\_time\_IDs

# removes PR-

# splits admissions as 0.01, 0.02 etc for repeat admissions

# adds the arrival time hour

# addes columns for all variables

create\_patient\_time\_ID <- function(patient\_list) {

counter <- 0

patient\_time\_ID <- matrix(NA, nrow(patient\_list), 1)

for (y in 1:nrow(patient\_time\_ID)) {

patient\_time\_ID[y,1] <- substr(patient\_list[y,1], 4, 11)

}

patient\_time\_ID[1, 1] <- paste(patient\_time\_ID[1, 1], ".01", sep="")

for (y in 2:nrow(patient\_time\_ID)) {

if (substr(patient\_list[y,1], 4, 11) == substr(patient\_list[y-1,1], 4, 11)) {

counter <- counter + 1}

else {

counter <- 0

}

if (counter < 10) {patient\_time\_ID[y, 1] <- paste(patient\_time\_ID[y, 1], ".0", counter+1, sep="")}

else {patient\_time\_ID[y, 1] <- paste(patient\_time\_ID[y, 1], ".", counter+1, sep="")}

}

for (y in 1:nrow(patient\_time\_ID)) {

patient\_time\_ID[y,1] <- paste(patient\_time\_ID[y, 1], substr(patient\_list[y,2], 1, 16), sep=" ")

}

patient\_list <- cbind(patient\_time\_ID, patient\_list)

variable\_key <- read.delim("variable\_key.csv", header=TRUE, sep=",", quote="", stringsAsFactors = FALSE)

for (i in 1:nrow(variable\_key)){

new\_column <- matrix(NA, nrow(patient\_list), 1)

colnames(new\_column)[1] <- variable\_key[i, "Variable"]

patient\_list <- cbind(patient\_list, new\_column)

}

return (patient\_list)

}

# converts lists of results into table as per instructions in variable\_key.csv

variable\_key <- read.delim("variable\_key.csv", header=TRUE, sep=",", quote="", stringsAsFactors = FALSE)

file\_locs <- unique(variable\_key[,2], incomparables = FALSE, fromLast = FALSE, nmax = NA)

all\_data <- read.delim("all\_PICU\_patients.csv", header=TRUE, sep=",", quote="", stringsAsFactors = FALSE)

all\_data <- subset(all\_data, select = -1)

for (loc in 1:length(file\_locs)) {

some\_data <- read.delim(file\_locs[loc], header=TRUE, sep=",", quote="", stringsAsFactors = FALSE) #cycles through all relevant files loading one at a time

for (y in 1:nrow(some\_data)) {

for (x in 7:(ncol(some\_data)-1)) {

for (entry in 1:nrow(variable\_key)) {

#locate entry of interest

if (some\_data[y, x] == variable\_key[entry, "Code"]) {

#extract the figure (last column always)

this\_value <- some\_data[y, ncol(some\_data)]

this\_variable <- variable\_key[entry, "Variable"]

#insert in correct location or create location in all\_data (must be after admission and before discharge)

entry\_entered = FALSE

for (n in 1:nrow(all\_PICU\_patients)) {

if (all\_PICU\_patients["project\_id"] == some\_data["project\_id"]) { #if right patients

if (substr(all\_PICU\_patients["patient\_time\_id"],13,28) == substr(some\_data["start\_datetime"]),1,16) #if right time

all\_PICU\_patients[n, this\_variable] <- this\_value #enter value in results table

entry\_entered <- TRUE

}

}

}

if (entry\_entered = FALSE) {

#check entry is within the time limits of that patient's admission

#copy the most recent row from all\_PICU\_patients to the next hour, insert entry there

}

}

}

}

}

}

# generate a list of all the patients who have had a PICU stay

# accesses 11846/ward\_stays.csv for this

# note this effectively excludes covid times when the PICU patients went to Dolphin

PICU\_list\_create <- function(ward\_stay\_file) {

#load data

ward\_stay\_data <- read.delim(ward\_stay\_file, header=TRUE, sep=",", quote="", stringsAsFactors = FALSE)

#PICU\_patients <- matrix(c(0, 0, 0), nrow=1, ncol=3, byrow = TRUE)

PICU\_patients <- c()

#search through data for presence of PICU in the ward\_code column

for (i in 1:nrow(ward\_stay\_data)) {

if (ward\_stay\_data[i, "ward\_code"] == "PICU") {

PICU\_patients <- rbind(PICU\_patients, ward\_stay\_data[i, 2:4]) #formerly ward\_stay\_data[i, "project\_id"]

}

}

PICU\_patients <- PICU\_patients[order(PICU\_patients[,"project\_id"], decreasing = FALSE),]

return (PICU\_patients)

}

#This to wrangle GOSH data from various large lists into a single data table

#get a list of all the PICU ward stays

source("PICU\_list\_create.r") #define extractor fuction

all\_PICU\_patients <- PICU\_list\_create("11868/ward\_stays.csv")

source("create\_patient\_time\_ID.r")

all\_PICU\_patients <- create\_patient\_time\_ID(all\_PICU\_patients)

write.csv(all\_PICU\_patients, "all\_PICU\_patients.csv", row.names=TRUE)

Variable,File,Code

Weight,11871/flowsheet\_rows.csv,82447

HR,11871/flowsheet\_rows.csv,67261

RR,11871/flowsheet\_rows.csv,67466

MAP,11871/flowsheet\_rows.csv,61924

SATs,11871/flowsheet\_rows.csv,40721

Temperature,11871/flowsheet\_rows.csv,74513

Elective/non-elective,11871/flowsheet\_rows.csv,66304

Blood gas pH,11870/lab\_components.csv,"pH POC"

Blood gas CO2,11870/lab\_components.csv,"paCO2 POC"

Blood gas O2,11870/lab\_components.csv,"paP2 POC"

Blood gas lactate,11870/lab\_components.csv,"Lactate POC"

Blood gas sodium,11870/lab\_components.csv,"Sodium POC"

Blood gas potassuim,11870/lab\_components.csv,"Potassium POC"

Blood gas calcium,11870/lab\_components.csv,"Calcium POC"

Boood gas chloride,11870/lab\_components.csv,"Chloride POC"

Blood gas glucose,11870/lab\_components.csv,"glucose POC"