install.packages("finalfit")

library("finalfit")

Valve<-subset(Readmit8,Readmit8$VALVE== "1. Yes"& CABG=="0. No" & MAJOR\_AORTIC=="0. No" & OTHER\_CARDIAC\_PROCS=="0. No")

#1.Count the number of readmission

table(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_1))

summary(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_2))

summary(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_3))

summary(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_4))

summary(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_5))

summary(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_6))

summary(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_7))

summary(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_8))

summary(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_9))

summary(is.na(Valve$DIAG\_4\_CONCAT\_ALL\_10))

#2. Number of 30 days readmission

Valve$datefromdischarge\_1<- difftime(Valve$ADMIDATE\_1,Valve$DATE\_DISCHARGE\_OR\_HOSP\_DEATH , units = c("days"))

table(sum(Valve$datefromdischarge\_1 < 30, na.rm = TRUE))

summary(as.numeric(Valve$datefromdischarge\_1))

#number of admission per month/year

yymm <- substr(Valve$ADMIDATE\_1, 1, 7)

yymm\_counts <- table(yymm)

print(yymm\_counts)

#number of discharge per month/year

Valve1 <- Valve %>%

mutate(

def = format(DATE\_DISCHARGE\_OR\_HOSP\_DEATH, "%Y-%m"))

table(Valve1$def)

#number of non readmission = discharge -yes admission

#causes of admission

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_1 = substr(DIAG\_4\_CONCAT\_ALL\_1, 1, 5))

frequency\_table1 <- table(Valve$DIAG\_4\_CONCAT\_01\_1)

sorted\_table1 <- sort(frequency\_table1, decreasing = TRUE)

print(head(sorted\_table1,20))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_2 = substr(DIAG\_4\_CONCAT\_ALL\_2, 1, 5))

frequency\_table2 <- table(Valve$DIAG\_4\_CONCAT\_01\_2)

sorted\_table2 <- sort(frequency\_table2, decreasing = TRUE)

print(head(sorted\_table2,20))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_3 = substr(DIAG\_4\_CONCAT\_ALL\_3, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_4 = substr(DIAG\_4\_CONCAT\_ALL\_4, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_5 = substr(DIAG\_4\_CONCAT\_ALL\_5, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_6 = substr(DIAG\_4\_CONCAT\_ALL\_6, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_7 = substr(DIAG\_4\_CONCAT\_ALL\_7, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_8 = substr(DIAG\_4\_CONCAT\_ALL\_8, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_9 = substr(DIAG\_4\_CONCAT\_ALL\_9, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_10 = substr(DIAG\_4\_CONCAT\_ALL\_10, 1, 5))

Valve1<-select(Valve,DIAG\_4\_CONCAT\_01\_1,DIAG\_4\_CONCAT\_01\_2,DIAG\_4\_CONCAT\_01\_3,DIAG\_4\_CONCAT\_01\_4,DIAG\_4\_CONCAT\_01\_5,DIAG\_4\_CONCAT\_01\_6,DIAG\_4\_CONCAT\_01\_7,DIAG\_4\_CONCAT\_01\_8,DIAG\_4\_CONCAT\_01\_9,DIAG\_4\_CONCAT\_01\_10)

# Combine all columns into a single vector

long\_df <- Valve1 %>%

pivot\_longer(cols = everything(), names\_to = "DIAG\_4\_CONCAT\_01", values\_to = "code")

# Count occurrences of each code

code\_counts <- long\_df %>%

count(code, sort = TRUE)

print(code\_counts,25)

#Secondary diagnosis

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_1))

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_2))

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_3))

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_4))

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_5))

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_6))

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_7))

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_8))

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_9))

length(grep("I61",Valve$DIAG\_4\_CONCAT\_ALL\_10))

##procedure

## Counting procedure

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_1))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_2))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_3))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_4))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_5))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_6))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_7))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_8))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_9))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_10))

#table for pre op characteristics

#Age

Valve$date1\_parsed <-as.Date(paste0(Valve$MONTH\_YEAR\_OF\_BIRTH,"-01"))

Valve$date2\_parsed <-as.POSIXct(Valve$DATE\_AND\_TIME\_OF\_OPERATION,format="%Y-%m-%d %H:%M:%S",tz="UTC")

Valve$date2\_parsed <-as.Date(Valve$date2\_parsed)

Valve$Age<- difftime(Valve$date2\_parsed,Valve$date1\_parsed , units = "weeks")

Valve$Age <- Valve$Age/52

#BMI

Valve <- Valve %>%

mutate(HEIGHT = ifelse(HEIGHT == 7179, 179, HEIGHT))

table(Valve$HEIGHT)

Valve$HEIGHT[which(is.na(Valve$HEIGHT))] = median(Valve$HEIGHT, na.rm=TRUE)

Valve$WEIGHT[which(is.na(Valve$WEIGHT))] = median(Valve$WEIGHT, na.rm=TRUE)

Valve$BMI <- Valve$WEIGHT / (Valve$HEIGHT/100 \* Valve$HEIGHT/100)

Valve$BMI[which(is.na(Valve$BMI))] = median(Valve$BMI, na.rm=TRUE)

median\_value <- median(Valve$BMI[!is.infinite(Valve$BMI)], na.rm = TRUE)

# Replace infinite values with the median

Valve$BMI[is.infinite(Valve$BMI)] <- median\_value

#Risk factors predicting 30 days and 12 month re-admission

Valve$ReadmissionYN <-ifelse(is.na(Valve$ADMIDATE\_1),0,1)

#Clean data Method 1

# Convert to factor if not already

Valve$OPERATIVE\_URGENCY <- as.factor(Valve$OPERATIVE\_URGENCY)

# Set new levels

levels(Valve$OPERATIVE\_URGENCY) <- c("1", "2", "3", "4")

# Check the result

table(Valve$OPERATIVE\_URGENCY)

Valve$CARDIOGENIC\_SHOCK\_PRE\_OP <- as.factor(Valve$CARDIOGENIC\_SHOCK\_PRE\_OP)

# Set new levels

levels(Valve$CARDIOGENIC\_SHOCK\_PRE\_OP) <- c("0", "1")

# Check the result

table(Valve$CARDIOGENIC\_SHOCK\_PRE\_OP)

Valve$ANGINA\_STATUS\_PRE\_SURGERY <- as.factor(Valve$ANGINA\_STATUS\_PRE\_SURGERY)

# Set new levels

levels(Valve$ANGINA\_STATUS\_PRE\_SURGERY) <- c("0", "1", "2", "3", "4")

# Check the result

table(Valve$ANGINA\_STATUS\_PRE\_SURGERY)

Valve$DYSPNOEA\_STATUS\_PRE\_SURGERY <- as.factor(Valve$DYSPNOEA\_STATUS\_PRE\_SURGERY)

# Set new levels

levels(Valve$DYSPNOEA\_STATUS\_PRE\_SURGERY) <- c("1", "2", "3", "4")

# Check the result

table(Valve$DYSPNOEA\_STATUS\_PRE\_SURGERY)

Valve$EJECTION\_FRACTION\_CATEGORY <- as.factor(Valve$EJECTION\_FRACTION\_CATEGORY)

# Set new levels

levels(Valve$EJECTION\_FRACTION\_CATEGORY) <- c("1", "2","2", "3","3", "4")

# Check the result

table(Valve$EJECTION\_FRACTION\_CATEGORY)

Valve$EXTRACARDIAC\_ARTERIOPATHY <- as.factor(Valve$EXTRACARDIAC\_ARTERIOPATHY)

# Set new levels

levels(Valve$EXTRACARDIAC\_ARTERIOPATHY) <- c("0", "1")

# Check the result

table(Valve$EXTRACARDIAC\_ARTERIOPATHY)

Valve$DIABETES\_MANAGEMENT <- as.factor(Valve$DIABETES\_MANAGEMENT)

# Set new levels

levels(Valve$DIABETES\_MANAGEMENT) <- c("0", "1","2","3")

# Check the result

table(Valve$DIABETES\_MANAGEMENT)

Valve$HISTORY\_OF\_HYPERTENSION <- as.factor(Valve$HISTORY\_OF\_HYPERTENSION)

# Set new levels

levels(Valve$HISTORY\_OF\_HYPERTENSION) <- c("0", "1","9")

# Check the result

table(Valve$HISTORY\_OF\_HYPERTENSION)

Valve$HISTORY\_OF\_NEUROLOGICAL\_DYSFN <- as.factor(Valve$HISTORY\_OF\_NEUROLOGICAL\_DYSFN)

# Set new levels

levels(Valve$HISTORY\_OF\_NEUROLOGICAL\_DYSFN) <- c("0", "1")

# Check the result

table(Valve$HISTORY\_OF\_NEUROLOGICAL\_DYSFN)

Valve$HISTORY\_OF\_PULMONARY\_DISEASE <- as.factor(Valve$HISTORY\_OF\_PULMONARY\_DISEASE)

# Set new levels

levels(Valve$HISTORY\_OF\_PULMONARY\_DISEASE) <- c("0","0","1", "1")

# Check the result

table(Valve$HISTORY\_OF\_PULMONARY\_DISEASE)

Valve$CIGARETTE\_SMOKING\_HISTORY <- as.factor(Valve$CIGARETTE\_SMOKING\_HISTORY)

# Set new levels

levels(Valve$CIGARETTE\_SMOKING\_HISTORY) <- c("0", "1","2")

# Check the result

table(Valve$CIGARETTE\_SMOKING\_HISTORY)

Valve$RENAL\_FUNCTION\_DIALYSIS <- as.factor(Valve$RENAL\_FUNCTION\_DIALYSIS)

# Set new levels

levels(Valve$RENAL\_FUNCTION\_DIALYSIS) <- c("0", "1","2","3","3")

# Check the result

table(Valve$RENAL\_FUNCTION\_DIALYSIS)

Valve$SEX <- as.factor(Valve$SEX)

# Set new levels

levels(Valve$SEX) <- c("1","2")

# Check the result

table(Valve$SEX)

Valve$INTERVAL\_SURGERY\_AND\_LAST\_MI <- as.factor(Valve$INTERVAL\_SURGERY\_AND\_LAST\_MI)

# Set new levels

levels(Valve$INTERVAL\_SURGERY\_AND\_LAST\_MI) <- c("0","1","2","3","4","5")

# Check the result

table(Valve$INTERVAL\_SURGERY\_AND\_LAST\_MI)

Valve$DSWI <- as.factor(Valve$DSWI)

# Set new levels

levels(Valve$DSWI) <- c("0", "1")

# Check the result

table(Valve$DSWI)

Valve$NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP <- as.factor(Valve$NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP)

# Set new levels

levels(Valve$NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP) <- c("0", "1")

# Check the result

table(Valve$NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP)

Valve$NEW\_POST\_OP\_NEUROLOGICAL\_DYSF <- as.factor(Valve$NEW\_POST\_OP\_NEUROLOGICAL\_DYSF)

# Set new levels

levels(Valve$NEW\_POST\_OP\_NEUROLOGICAL\_DYSF) <- c("0","2","1","1","2","2","2","2","2","2")

# Check the result

table(Valve$NEW\_POST\_OP\_NEUROLOGICAL\_DYSF)

#Predict

library(stats)

Valve$ReadmissionYN <-factor(Valve$ReadmissionYN)

Valve$Age <-as.numeric(Valve$Age)

Valve$BMI <-as.numeric(Valve$BMI)

Valve$SEX <-factor(Valve$SEX)

Valve$OPERATIVE\_URGENCY <- factor(Valve$OPERATIVE\_URGENCY)

Valve$ANGINA\_STATUS\_PRE\_SURGERY <-factor(Valve$ANGINA\_STATUS\_PRE\_SURGERY)

Valve$CARDIOGENIC\_SHOCK\_PRE\_OP <-factor(Valve$CARDIOGENIC\_SHOCK\_PRE\_OP )

Valve$DYSPNOEA\_STATUS\_PRE\_SURGERY <-factor(Valve$DYSPNOEA\_STATUS\_PRE\_SURGERY )

Valve$DIABETES\_MANAGEMENT <-factor(Valve$DIABETES\_MANAGEMENT )

Valve$EJECTION\_FRACTION\_CATEGORY <-factor(Valve$EJECTION\_FRACTION\_CATEGORY )

Valve$EXTRACARDIAC\_ARTERIOPATHY <-factor(Valve$EXTRACARDIAC\_ARTERIOPATHY )

Valve$HISTORY\_OF\_HYPERTENSION <-factor(Valve$HISTORY\_OF\_HYPERTENSION )

Valve$HISTORY\_OF\_NEUROLOGICAL\_DYSFN <-factor(Valve$HISTORY\_OF\_NEUROLOGICAL\_DYSFN )

Valve$HISTORY\_OF\_PULMONARY\_DISEASE <-factor(Valve$HISTORY\_OF\_PULMONARY\_DISEASE )

Valve$CIGARETTE\_SMOKING\_HISTORY <-factor(Valve$CIGARETTE\_SMOKING\_HISTORY )

Valve$RENAL\_FUNCTION\_DIALYSIS <-factor(Valve$RENAL\_FUNCTION\_DIALYSIS)

Valve$INTERVAL\_SURGERY\_AND\_LAST\_MI <-factor(Valve$INTERVAL\_SURGERY\_AND\_LAST\_MI)

Valve$NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP <-factor(Valve$NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP )

Valve$DSWI <-factor(Valve$DSWI )

Valve$NEW\_POST\_OP\_NEUROLOGICAL\_DYSF <-factor(Valve$NEW\_POST\_OP\_NEUROLOGICAL\_DYSF)

Predictreadmission <- glm(ReadmissionYN ~ Age + BMI + SEX+OPERATIVE\_URGENCY+ANGINA\_STATUS\_PRE\_SURGERY+CARDIOGENIC\_SHOCK\_PRE\_OP+DYSPNOEA\_STATUS\_PRE\_SURGERY+DIABETES\_MANAGEMENT+

EJECTION\_FRACTION\_CATEGORY+EXTRACARDIAC\_ARTERIOPATHY+HISTORY\_OF\_HYPERTENSION+HISTORY\_OF\_NEUROLOGICAL\_DYSFN+HISTORY\_OF\_PULMONARY\_DISEASE+CIGARETTE\_SMOKING\_HISTORY+RENAL\_FUNCTION\_DIALYSIS

+INTERVAL\_SURGERY\_AND\_LAST\_MI+NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP+DSWI+NEW\_POST\_OP\_NEUROLOGICAL\_DYSF

,data=Valve,family="binomial")

class(Predictreadmission)

summary(Predictreadmission)

explanatory <- c( "Age" , "BMI" , "SEX" , "OPERATIVE\_URGENCY" ,

"ANGINA\_STATUS\_PRE\_SURGERY" , "CARDIOGENIC\_SHOCK\_PRE\_OP" , "DYSPNOEA\_STATUS\_PRE\_SURGERY" ,

"DIABETES\_MANAGEMENT" , "EJECTION\_FRACTION\_CATEGORY" , "EXTRACARDIAC\_ARTERIOPATHY",

"HISTORY\_OF\_HYPERTENSION" , "HISTORY\_OF\_NEUROLOGICAL\_DYSFN",

"HISTORY\_OF\_PULMONARY\_DISEASE" , "CIGARETTE\_SMOKING\_HISTORY" ,

"RENAL\_FUNCTION\_DIALYSIS" , "INTERVAL\_SURGERY\_AND\_LAST\_MI","NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP")

dependent <- "ReadmissionYN"

table2 <- Valve %>%

finalfit(dependent, explanatory,

dependent\_label\_prefix = "")

table2

#gtsummary compare readmission vs no readmission

Preop <- select(Valve, Age , BMI , SEX , OPERATIVE\_URGENCY ,

ANGINA\_STATUS\_PRE\_SURGERY , CARDIOGENIC\_SHOCK\_PRE\_OP , DYSPNOEA\_STATUS\_PRE\_SURGERY ,

DIABETES\_MANAGEMENT , EJECTION\_FRACTION\_CATEGORY , EXTRACARDIAC\_ARTERIOPATHY,

HISTORY\_OF\_HYPERTENSION , HISTORY\_OF\_NEUROLOGICAL\_DYSFN,

HISTORY\_OF\_PULMONARY\_DISEASE , CIGARETTE\_SMOKING\_HISTORY ,

RENAL\_FUNCTION\_DIALYSIS , INTERVAL\_SURGERY\_AND\_LAST\_MI,ReadmissionYN)

###########################################################################################################Survival

library(lubridate)

library(tidyverse)

library(ggsurvfit)

Valve1 <-Valve

Valve1$REG\_DATE\_OF\_DEATH <- as.Date(as.character(Valve1$REG\_DATE\_OF\_DEATH), format = "%Y%m%d")

summary(Valve1$REG\_DATE\_OF\_DEATH)

Valve1$DATE\_AND\_TIME\_OF\_OPERATION <- as.Date(Valve1$DATE\_AND\_TIME\_OF\_OPERATION)

summary(Valve1$DATE\_AND\_TIME\_OF\_OPERATION)

Valve1 <- Valve1 %>%

filter(is.na(REG\_DATE\_OF\_DEATH) | REG\_DATE\_OF\_DEATH >= DATE\_AND\_TIME\_OF\_OPERATION)

#filter 7

Valve1 <-

Valve1 %>%

mutate(

os\_months = as.duration(DATE\_AND\_TIME\_OF\_OPERATION %--% REG\_DATE\_OF\_DEATH) / dmonths(1)

)

summary(Valve1$os\_months)

Valve1$Status <- NA

Valve1 <- Valve1 %>%

mutate(os\_months = replace\_na(os\_months, 0)) %>%

mutate(Status = if\_else(os\_months > 0, 1, 0))

Valve1$REG\_DATE\_OF\_DEATH[is.na(Valve1$REG\_DATE\_OF\_DEATH)] <- "2023-06-28"

Valve1 <- Valve1 %>%

filter(is.na(REG\_DATE\_OF\_DEATH) | REG\_DATE\_OF\_DEATH >= DATE\_AND\_TIME\_OF\_OPERATION)

Valve1 <-

Valve1 %>%

mutate(

os\_months = as.duration(DATE\_AND\_TIME\_OF\_OPERATION %--% REG\_DATE\_OF\_DEATH) / dmonths(1)

)

survfit2(Surv(os\_months, Status) ~ Valve1$ReadmissionYN, data = Valve1) %>%

ggsurvfit() +

labs(

y = "Percentage Survival",

title = "12 month survival in patients with/without readmission",

x= "Time,Months",

) +

scale\_color\_manual(values = c('brown1', 'cyan'),

labels = c('No Readmission', 'Readmission')) +

scale\_fill\_manual(values = c('brown1', 'cyan'),

labels = c('No Readmission', 'Readmission')) +

add\_pvalue(caption = "Log-rank {p.value}")+

add\_legend\_title("Admission")+

add\_risktable(risktable\_stats = "{n.risk} ({cum.event})")

######################################################################################################

Valve2 <- Valve1 %>%

filter(is.na(REG\_DATE\_OF\_DEATH) | REG\_DATE\_OF\_DEATH >= DATE\_AND\_TIME\_OF\_OPERATION)

Valve2 <- subset(Valve2, PATIENT\_STATUS\_AT\_DISCHARGE == "0. Alive")

#1.Count the number of readmission

table(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_1))

summary(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_2))

summary(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_3))

summary(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_4))

summary(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_5))

summary(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_6))

summary(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_7))

summary(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_8))

summary(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_9))

summary(is.na(Valve2$DIAG\_4\_CONCAT\_ALL\_10))

#2. Number of 365 days readmission

Valve2$DATE\_AND\_TIME\_OF\_OPERATION <- as.Date(Valve2$DATE\_AND\_TIME\_OF\_OPERATION, format = "%Y-%m-%d")

Valve2$ADMIDATE\_1 <- as.Date(Valve2$ADMIDATE\_1, format = "%Y-%m-%d")

Valve2$REG\_DATE\_OF\_DEATH <- as.Date(Valve2$REG\_DATE\_OF\_DEATH, format = "%Y-%m-%d")

# Filter patients who had their operation in 2013

patients\_2013 <- Valve2 %>% filter(format(DATE\_AND\_TIME\_OF\_OPERATION, "%Y") == "2013")

# Calculate the difference between operation date and readmission date in days

patients\_2013 <- patients\_2013 %>%

mutate(DaysToReadmission = as.numeric(ADMIDATE\_1 - DATE\_AND\_TIME\_OF\_OPERATION))

# Calculate the cutoff date for 12 months after the operation

patients\_2013 <- patients\_2013 %>%

mutate(TwelveMonthCutoff = DATE\_AND\_TIME\_OF\_OPERATION + 365)

# Exclude patients who died before they could be readmitted (within 12 months)

patients\_eligible\_for\_readmission <- patients\_2013 %>%

filter(Status == 0 | (Status == 1 & REG\_DATE\_OF\_DEATH > TwelveMonthCutoff))

# Filter patients who had a readmission within 12 months after operation

readmitted\_within\_12\_months <- patients\_eligible\_for\_readmission %>%

filter(!is.na(ADMIDATE\_1) & DaysToReadmission <= 365)

### Calculate the readmission rate

Valve2 <- Valve2 %>%

mutate(

Cutoff\_30days = DATE\_AND\_TIME\_OF\_OPERATION + 30,

Cutoff\_3months = DATE\_AND\_TIME\_OF\_OPERATION + 90,

Cutoff\_6months = DATE\_AND\_TIME\_OF\_OPERATION + 180,

Cutoff\_12months = DATE\_AND\_TIME\_OF\_OPERATION + 365

)

calculate\_rates <- function(df, cutoff\_date, time\_period) {

# Filter patients who were readmitted within the given time period

readmitted <- df %>%

filter(!is.na(ADMIDATE\_1) & ADMIDATE\_1 <= !!sym(cutoff\_date) &

(Status == 0 | (Status == 1 & ADMIDATE\_1 <= REG\_DATE\_OF\_DEATH)))

# Filter patients who died within the given time period

mortality <- df %>%

filter(Status == 1 & REG\_DATE\_OF\_DEATH <= !!sym(cutoff\_date))

# Calculate the number of patients readmitted and those who died

num\_readmitted <- nrow(readmitted)

num\_died <- nrow(mortality)

# Calculate the number of patients who had an operation and were eligible for readmission

num\_eligible <- nrow(df)

# Calculate the readmission rate

readmission\_rate <- num\_readmitted / num\_eligible

# Calculate mortality rate

mortality\_rate <- num\_died / num\_eligible

# Print results

cat("For", time\_period, ":\n")

cat("Number of patients readmitted within", time\_period, ":", num\_readmitted, "\n")

cat("Number of patients who died within", time\_period, ":", num\_died, "\n")

cat("Number of patients eligible for readmission:", num\_eligible, "\n")

cat("Readmission rate within", time\_period, ":", readmission\_rate, "\n")

cat("Mortality rate within", time\_period, ":", mortality\_rate, "\n\n")

}

# Calculate and print readmission and mortality rates for each time period

calculate\_rates(Valve2, "Cutoff\_30days", "30 days")

calculate\_rates(Valve2, "Cutoff\_3months", "3 months")

calculate\_rates(Valve2, "Cutoff\_6months", "6 months")

calculate\_rates(Valve2, "Cutoff\_12months", "12 months")

#####causes of admission

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_1 = substr(DIAG\_4\_CONCAT\_ALL\_1, 1, 5))

frequency\_table1 <- table(Valve$DIAG\_4\_CONCAT\_01\_1)

sorted\_table1 <- sort(frequency\_table1, decreasing = TRUE)

print(head(sorted\_table1,20))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_2 = substr(DIAG\_4\_CONCAT\_ALL\_2, 1, 5))

frequency\_table2 <- table(Valve2$DIAG\_4\_CONCAT\_01\_2)

sorted\_table2 <- sort(frequency\_table2, decreasing = TRUE)

print(head(sorted\_table2,20))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_3 = substr(DIAG\_4\_CONCAT\_ALL\_3, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_4 = substr(DIAG\_4\_CONCAT\_ALL\_4, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_5 = substr(DIAG\_4\_CONCAT\_ALL\_5, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_6 = substr(DIAG\_4\_CONCAT\_ALL\_6, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_7 = substr(DIAG\_4\_CONCAT\_ALL\_7, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_8 = substr(DIAG\_4\_CONCAT\_ALL\_8, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_9 = substr(DIAG\_4\_CONCAT\_ALL\_9, 1, 5))

Valve <- Valve %>% mutate(DIAG\_4\_CONCAT\_01\_10 =substr(DIAG\_4\_CONCAT\_ALL\_10, 1, 5))

Valve3<-select(Valve,DIAG\_4\_CONCAT\_01\_1,DIAG\_4\_CONCAT\_01\_2,DIAG\_4\_CONCAT\_01\_3,DIAG\_4\_CONCAT\_01\_4,DIAG\_4\_CONCAT\_01\_5,DIAG\_4\_CONCAT\_01\_6,DIAG\_4\_CONCAT\_01\_7,DIAG\_4\_CONCAT\_01\_8,DIAG\_4\_CONCAT\_01\_9,DIAG\_4\_CONCAT\_01\_10)

# Combine all columns into a single vector

long\_df <- Valve3 %>%

pivot\_longer(cols = everything(), names\_to = "DIAG\_4\_CONCAT\_01", values\_to = "code")

# Count occurrences of each code

code\_counts <- long\_df %>%

count(code, sort = TRUE)

print(code\_counts,25)

#find out how many primary diagnosis of a specific diagnosis

specific\_code <- 'I489,'

abc <- code\_counts %>%

filter(code == specific\_code) %>%

pull(n)

print(abc)

#Secondary diagnosis

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_1))

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_2))

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_3))

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_4))

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_5))

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_6))

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_7))

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_8))

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_9))

length(grep("I61",Valve2$DIAG\_4\_CONCAT\_ALL\_10))

##procedure

## Counting procedure

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_1))

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_2))

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_3))

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_4))

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_5))

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_6))

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_7))

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_8))

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_9))

length(grep("K59",Valve2$OPERTN\_4\_CONCAT\_ALL\_10))

#Risk factors predicting 30 days and 12 month re-admission

Valve$ReadmissionYN <-ifelse(is.na(Valve$ADMIDATE\_1),0,1)

Valve2$ReadmissionYN <-ifelse(is.na(Valve2$ADMIDATE\_1),0,1)

#Predict

library(stats)

names(Valve2)[names(Valve2) == "3\_96\_SURGICAL\_INCISION"] <- "SURGICAL\_INCISION"

Predictreadmission <- glm(ReadmissionYN ~ Age + BMI + SEX+OPERATIVE\_URGENCY+ANGINA\_STATUS\_PRE\_SURGERY+CARDIOGENIC\_SHOCK\_PRE\_OP+DYSPNOEA\_STATUS\_PRE\_SURGERY+DIABETES\_MANAGEMENT+

EJECTION\_FRACTION\_CATEGORY+EXTRACARDIAC\_ARTERIOPATHY+HISTORY\_OF\_HYPERTENSION+HISTORY\_OF\_NEUROLOGICAL\_DYSFN+HISTORY\_OF\_PULMONARY\_DISEASE+CIGARETTE\_SMOKING\_HISTORY+RENAL\_FUNCTION\_DIALYSIS

+INTERVAL\_SURGERY\_AND\_LAST\_MI+NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP+DSWI+NEW\_POST\_OP\_NEUROLOGICAL\_DYSF+CUMULATIVE\_BYPASS\_TIME+CUMULATIVE\_CROSS\_CLAMP\_TIME+

VALVES\_REPLACED\_REPAIRED+NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP+SURGICAL\_INCISION+RETURN\_TO\_THEATRE+NEW\_POST\_OP\_NEUROLOGICAL\_DYSF+DSWI

,data=Valve2,family="binomial")

class(Predictreadmission)

summary(Predictreadmission)

#gtsummary compare readmission vs no readmission

Preop <- select(Valve, Age , BMI , SEX , OPERATIVE\_URGENCY ,

ANGINA\_STATUS\_PRE\_SURGERY , CARDIOGENIC\_SHOCK\_PRE\_OP , DYSPNOEA\_STATUS\_PRE\_SURGERY ,

DIABETES\_MANAGEMENT , EJECTION\_FRACTION\_CATEGORY , EXTRACARDIAC\_ARTERIOPATHY,

HISTORY\_OF\_HYPERTENSION , HISTORY\_OF\_NEUROLOGICAL\_DYSFN,

HISTORY\_OF\_PULMONARY\_DISEASE , CIGARETTE\_SMOKING\_HISTORY ,

RENAL\_FUNCTION\_DIALYSIS , INTERVAL\_SURGERY\_AND\_LAST\_MI,ReadmissionYN)

## Post op

names(Valve)[names(Valve) == "3\_96\_SURGICAL\_INCISION"] <- "SURGICAL\_INCISION"

postop <- c( "CUMULATIVE\_BYPASS\_TIME","CUMULATIVE\_CROSS\_CLAMP\_TIME","VALVES\_REPLACED\_REPAIRED","PATIENT\_STATUS\_AT\_DISCHARGE","NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP","SURGICAL\_INCISION","RETURN\_TO\_THEATRE","NEW\_POST\_OP\_NEUROLOGICAL\_DYSF","DSWI")

dependent <- "ReadmissionYN"

table3 <- Valve %>%

finalfit(dependent, postop,

dependent\_label\_prefix = "")

table3

Valve2$CUMULATIVE\_BYPASS\_TIME <-as.numeric(Valve2$CUMULATIVE\_BYPASS\_TIME)

Valve2$CUMULATIVE\_CROSS\_CLAMP\_TIME <-as.numeric(Valve2$CUMULATIVE\_CROSS\_CLAMP\_TIME)

Valve2$VALVES\_REPLACED\_REPAIRED <-as.factor(Valve2$VALVES\_REPLACED\_REPAIRED)

# Set new levels

levels(Valve2$PATIENT\_STATUS\_AT\_DISCHARGE) <- c("0", "1")

# Check the result

table(Valve2$PATIENT\_STATUS\_AT\_DISCHARGE)

Valve2$"SURGICAL\_INCISION" <- as.factor(Valve2$"SURGICAL\_INCISION")

# Set new levels

levels(Valve2$"SURGICAL\_INCISION") <- c("1", "1","1","1","1","1","1","2","2","3","3","3","3","3","3","3","3","3","3","3","3","3","3","4","4","4","4","4")

# Check the result

table(Valve2$"SURGICAL\_INCISION")

Valve2$RETURN\_TO\_THEATRE <- as.factor(Valve2$RETURN\_TO\_THEATRE)

# Set new levels

levels(Valve2$RETURN\_TO\_THEATRE) <- c("0","0","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1","1")

# Check the result

table(Valve2$RETURN\_TO\_THEATRE)

##gtsummary

library(gtsummary)

Preop <- select(Valve2, Age , BMI , SEX , OPERATIVE\_URGENCY ,

ANGINA\_STATUS\_PRE\_SURGERY , CARDIOGENIC\_SHOCK\_PRE\_OP , DYSPNOEA\_STATUS\_PRE\_SURGERY ,

DIABETES\_MANAGEMENT , EJECTION\_FRACTION\_CATEGORY , EXTRACARDIAC\_ARTERIOPATHY,

HISTORY\_OF\_HYPERTENSION , HISTORY\_OF\_NEUROLOGICAL\_DYSFN,

HISTORY\_OF\_PULMONARY\_DISEASE , CIGARETTE\_SMOKING\_HISTORY ,

RENAL\_FUNCTION\_DIALYSIS , INTERVAL\_SURGERY\_AND\_LAST\_MI,ReadmissionYN)

Postop <- select(Valve2,"CUMULATIVE\_BYPASS\_TIME","CUMULATIVE\_CROSS\_CLAMP\_TIME","VALVES\_REPLACED\_REPAIRED","NEW\_HAEMOFILT\_OR\_DIAL\_POST\_OP",

"SURGICAL\_INCISION","RETURN\_TO\_THEATRE","NEW\_POST\_OP\_NEUROLOGICAL\_DYSF","DSWI",ReadmissionYN)

Preop %>%

tbl\_summary(by = ReadmissionYN, statistic = list(all\_continuous() ~ "{mean} ({sd})", all\_categorical() ~ "{n} / {N} ({p}%)"), digits = all\_continuous() ~ 2,) %>%

add\_p(pvalue\_fun = ~style\_pvalue(.x, digits = 2)) %>%

add\_overall() %>%

add\_n() %>%

modify\_header(label ~ "\*\*Variable\*\*") %>%

modify\_spanning\_header(c("stat\_1", "stat\_2") ~ "\*\*Treatment Received\*\*") %>%

modify\_caption("\*\*Table 1. Patient Characteristics\*\*") %>%

bold\_labels()

Postop %>%

tbl\_summary(by = ReadmissionYN, statistic = list(all\_continuous() ~ "{mean} ({sd})", all\_categorical() ~ "{n} / {N} ({p}%)"), digits = all\_continuous() ~ 2,) %>%

add\_p(pvalue\_fun = ~style\_pvalue(.x, digits = 2)) %>%

add\_overall() %>%

add\_n() %>%

modify\_header(label ~ "\*\*Variable\*\*") %>%

modify\_spanning\_header(c("stat\_1", "stat\_2") ~ "\*\*Treatment Received\*\*") %>%

modify\_caption("\*\*Table 1. Patient Characteristics\*\*") %>%

bold\_labels()

###############################################################COVID time period#########################################################################

Prelockdown <- Valve1[Valve1$DATE\_AND\_TIME\_OF\_OPERATION >= "2010-01-01" & Valve1$DATE\_AND\_TIME\_OF\_OPERATION <= "2020-03-22", ]

table(Prelockdown$ReadmissionYN)

Firstlockdown <- Valve1[Valve1$DATE\_AND\_TIME\_OF\_OPERATION >= "2020-03-23" & Valve1$DATE\_AND\_TIME\_OF\_OPERATION <= "2020-06-23", ]

table(Firstlockdown$ReadmissionYN)

Firstrelaxation <- Valve1[Valve1$DATE\_AND\_TIME\_OF\_OPERATION >= "2020-06-24" & Valve1$DATE\_AND\_TIME\_OF\_OPERATION <= "2020-11-04", ]

table(Firstrelaxation$ReadmissionYN)

Secondlockdown <- Valve1[Valve1$DATE\_AND\_TIME\_OF\_OPERATION >= "2020-11-05" & Valve1$DATE\_AND\_TIME\_OF\_OPERATION <= "2020-12-02", ]

table(Secondlockdown$ReadmissionYN)

Firstrelaxation <- Valve1[Valve1$DATE\_AND\_TIME\_OF\_OPERATION >= "2020-06-24" & Valve1$DATE\_AND\_TIME\_OF\_OPERATION <= "2020-11-04", ]

table(Firstrelaxation$ReadmissionYN)

Primary and secondary analysis

library(dplyr)

library(stringr)

library(tidyr)

library(purrr)

Valve5<-select(Valve2,DIAG\_4\_CONCAT\_ALL\_1,DIAG\_4\_CONCAT\_ALL\_2,DIAG\_4\_CONCAT\_ALL\_3,DIAG\_4\_CONCAT\_ALL\_4,DIAG\_4\_CONCAT\_ALL\_5,DIAG\_4\_CONCAT\_ALL\_6,DIAG\_4\_CONCAT\_ALL\_7,DIAG\_4\_CONCAT\_ALL\_8,DIAG\_4\_CONCAT\_ALL\_9,DIAG\_4\_CONCAT\_ALL\_10)

# Combine all columns into a single vector

long\_df1 <- Valve5 %>%

pivot\_longer(cols = everything(), names\_to = "DIAG\_4\_CONCAT\_ALL\_1", values\_to = "code")

# Count occurrences of each code

code\_counts <- long\_df1 %>%

count(code, sort = TRUE)

print(code\_counts,25)

# Step 1: Separate Primary and Secondary Diagnoses

Valve6 <- long\_df1 %>%

mutate(

Primary\_Diag = str\_split(code, ",|\\s") %>% map\_chr(~ .[1]), # Extract primary diagnosis from DIAG\_4\_CONCAT\_ALL\_1

Secondary\_Diag = str\_split(code, ",|\\s") %>% map(~ .[-1]) # Extract secondary diagnoses from DIAG\_4\_CONCAT\_ALL\_1

)

# Step 2: Count the frequency of each Primary Diagnosis

primary\_diagnosis\_counts <- Valve6 %>%

group\_by(Primary\_Diag) %>%

summarise(count = n()) %>%

arrange(desc(count)) # Sort by the most common diagnoses

# Step 3: Get the Top 100 Most Common Primary Diagnoses

top\_100\_primary\_diagnoses <- primary\_diagnosis\_counts %>%

slice\_head(n = 101)

# View the Top 100 Primary Diagnoses

print(top\_100\_primary\_diagnoses)

# Step 4: Unnest Secondary Diagnoses for separate analysis

Valve\_secondary <- Valve6 %>%

unnest(Secondary\_Diag) # Each secondary diagnosis in its own row

# Step 5: Count the frequency of each Secondary Diagnosis

secondary\_diagnosis\_counts <- Valve\_secondary %>%

group\_by(Secondary\_Diag) %>%

summarise(count = n()) %>%

arrange(desc(count))

# Step 6: Get the Top 100 Most Common Secondary Diagnoses (optional)

top\_100\_secondary\_diagnoses <- secondary\_diagnosis\_counts %>%

slice\_head(n = 100)

# View the top 100 secondary diagnoses (if needed)

print(top\_100\_secondary\_diagnoses)

matching\_rows <- grep("I61", primary\_diagnosis\_counts$Primary\_Diag)

total\_count <- sum(primary\_diagnosis\_counts$count[matching\_rows])   
total\_count

length(grep("I61",Valve\_secondary$Secondary\_Diag))

#####calculate cardiovascular related primary diagnosis

#Remove first row in primary diagnosis

primary\_diagnosis\_counts <- primary\_diagnosis\_counts[-1, ]

#calculate the number of row with I and no I

#consist of I (10318)

matching\_rows <- grep("I", primary\_diagnosis\_counts$Primary\_Diag)

total\_count <- sum(primary\_diagnosis\_counts$count[matching\_rows])   
total\_count

#calculate sum of column and minus 10318 (n=42430, 42430-10318=32112)

total\_sum <- sum(primary\_diagnosis\_counts$count, na.rm = TRUE)

#Cardiovascular secondary diagnosis (Total row 429212)  
length(grep("I",Valve\_secondary$Secondary\_Diag))

#107151 (Non cardiovascular secondary diagnosis = 429212-107151=322061)

Procedure

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_1))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_2))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_3))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_4))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_5))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_6))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_7))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_8))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_9))

length(grep("K59",Valve$OPERTN\_4\_CONCAT\_ALL\_10))