Identification of key risk factors associated with severe hypoglycemia in older adults with Type 1 Diabetes

Nikki L. B. Freeman

Table of Contents

This notebook contains the main steps in the analysis for “Identification of key risk factors associated with severe hypoglycemia in older adults with Type 1 Diabetes”. Function files are located in the same directory as this notebook.

# Packages, functions, and data

The getRawDataTables function reads in the individual data tables from the raw data set.

# Packages needed for the analysis  
library(tidyverse)  
library(mice)  
library(caret)  
library(randomForest)  
library(glmnet)  
library(xtable)  
library(table1)  
  
# Functions  
source("1\_createAnalysisRaw.R")  
source("2\_dataPreparation2.R")  
source("3\_imputation.R")  
source("4\_randomForest.R")  
source("5\_regLogisticReg.R")  
  
# Load the raw data  
dataTables <- getRawDataTables("../")

# Data cleaning and preparation

The collectVariables function extracts from each data table, the variables of interest for the analysis. For some variables, minor data cleaning is completed, including

* Adding an inclusion/exclusion variable: two participants are excluded due to not having any demographic data available
* Adjudicating duplicate lab results: if one entry is NULL and the other is not, take the non-NULL entry; otherwise, average the entries.
* Scoring the diabetes numeracy test
* Scoring the geriatric depression test
* Identifying beta blockers from the medication list and creating a binary beta blocker variable
* Scoring the hypoglycemia unawareness survey

ThemakeAnalysisDataWithMissing does additional data cleaning and perparation to yield a dataframe with all of the observed data that will be used for the analysis. The cleaning steps include:

* Re-coding education into categories (highest educational attainment <= HS, Some college or college degree, or advanced degree)
* Re-coding insurance into a binary variable for having insurance (public, private, single service) or not
* Calculating BMI and categorizing into BMI classes (BMI < 18.5 as underweight, 18.5 BMI < 25 as normal weight, 15 BMI < 30 as overweight, and BMI 30 as obese)
* Re-coding total insulin units as <40, 40 and <60, and 60
* Averaging the two frailty walk scores to yield a single frailty walk score for each individual
* Hopkins Verbal learning test is the sum of the total correct from the first 3 trials, the delayed recall is the value of the 4th trial

# Collect the variables of interest from the various tables and put into a single analysisData\_raw dataframe  
# Some variables are missing their scores or are re-coded  
analysisData\_raw <- collectVariables(dataTables)   
# Write the analysis data to a csv  
write\_csv(x = analysisData\_raw, file = "../2\_pipeline/1\_analysis\_raw.csv")  
  
# Create analytic data set for analysis  
analysisWithMissing <- makeAnalysisDataWithMissing(analysisData\_raw = analysisData\_raw)   
# Write the analysis data set to a csv  
write\_csv(x = analysisWithMissing, file = "../2\_pipeline/2\_analysisDataWithMissing.csv")

# Imputation and preparation of testing and training sets

The getImputationDataSets function takes the observed data for analysis and creates 5 imputed dated sets. The full observed data set for imputation. The createTrainTest function takes the imputed dated and observed data to generate a test set composed of only complete cases and 5 imputation sets that are used for training the models (effectively, these are folds).

set.seed(5678)  
imputedData <- getImputationDataSets(analysisWithMissing = analysisWithMissing)  
# trainTest <- createTrainTest(analysisWithMissing = analysisWithMissing, imputedData = imputedData)

# Model Fitting

## Generate the common test set

# Create a common test set across models  
completeCases <- analysisWithMissing[complete.cases(analysisWithMissing),]  
testSet <- caret::createDataPartition(y = completeCases$BCaseControlStatus, p = 0.40)  
testSetIDs <- completeCases[testSet[[1]],]$PtID

## Predictor selection and train/test construction

### Model 1: Demographic and clinical variables

predictors\_model1 <- c("PtID", "BCaseControlStatus", "exclude",   
 "Gender", "nonHispWhite", "educationCat", "insurance",   
 "bmiCat", "insDoseCat", "glucoseMonitoringCat", "HBA1C",   
 "detectableCPEP", "betaBlocker", "abnormalCreatinine",   
 "frailty", "hypoUnaware")  
imputedData\_model1 <- imputedData %>%  
 select(all\_of(predictors\_model1), ".imp", ".id")  
  
trainTest\_model1 <- createTrainTest(analysisWithMissing = analysisWithMissing,   
 imputedData = imputedData\_model1,  
 predictors = predictors\_model1,   
 testSetIDs = testSetIDs)

### Model 2: Demographic and clinical variables (Model 1) + behavioral and lifestyle factors

predictors\_model2 <- c(predictors\_model1,  
 "DaysWkEx", "LiveAlone", "hypoFear", "DukeSocTotDSSI",  
 "FuncActTotTestScore")  
imputedData\_model2 <- imputedData %>%  
 dplyr::select(all\_of(predictors\_model2), ".imp", ".id")  
  
trainTest\_model2 <- createTrainTest(analysisWithMissing = analysisWithMissing,  
 imputedData = imputedData\_model2,  
 predictors = predictors\_model2,  
 testSetIDs = testSetIDs)

### Model 3: Demographic and clinical variables (Model 1) + behavioral and lifestyle factors (Model 2) + neurocognitive factors

predictors\_model3 <- c(predictors\_model2,   
 "MoCATotal", "SymbDigWTotCorr", "SymbDigOTotCorr",  
 "TrailMakATotTime", "TrailMakBTotTime", "GrPegDomTotTime")  
imputedData\_model3 <- imputedData %>%  
 dplyr::select(all\_of(predictors\_model3), ".imp", ".id")  
  
trainTest\_model3 <- createTrainTest(analysisWithMissing = analysisWithMissing,  
 imputedData = imputedData\_model3,  
 predictors = predictors\_model3,  
 testSetIDs = testSetIDs)

### Model 4: Demographic and clinical variables (Model 1) + behavior and lifestyle factors (Model 2) + neurocognitive factors (Model 3) + CGM metrics

predictors\_model4 <- c(predictors\_model3,  
 "bgLess70\_pct", "pctCV")  
  
imputedData\_model4 <- imputedData %>%  
 dplyr::select(all\_of(predictors\_model4), ".imp", ".id")  
  
trainTest\_model4 <- createTrainTest(analysisWithMissing = analysisWithMissing,  
 imputedData = imputedData\_model4,  
 predictors = predictors\_model4,  
 testSetIDs = testSetIDs)

## Random Forest

## Model 1: Demographic and clinical variables ----------------------------------------------------------------  
# Fit the random forest to the training data  
rfObjs\_model1 <- map(trainTest\_model1$trainingSets\_list, rfHelper)  
# Predict on the test set  
rfPredictions\_model1 <- map(rfObjs\_model1, predict, trainTest\_model1$testingData %>%  
 dplyr::select(-c(BCaseControlStatus, PtID)))  
# Extract variable importance  
rfImportance\_model1 <- map(rfObjs\_model1, rfImportanceHelper)  
# Performance metrics  
rfMetrics\_model1 <- map(rfPredictions\_model1, confusionMatrix, reference = factor(trainTest\_model1$testingData$BCaseControlStatus))   
rfPlotMetrics\_model1 <- data.frame(reduce(map(1:5, ~rfMetrics\_model1[[.]]$byClass), rbind)) %>%   
 mutate(Method = "RF")   
  
## Model 2: Demographic and clinical variables (Model 1) + behavioral and lifestyle factors --------------------  
# Fit the random forest to the training data  
rfObjs\_model2 <- map(trainTest\_model2$trainingSets\_list, rfHelper)  
# Predict on the test set  
rfPredictions\_model2 <- map(rfObjs\_model2, predict, trainTest\_model2$testingData %>%  
 dplyr::select(-c(BCaseControlStatus, PtID)))  
# Extract variable importance  
rfImportance\_model2 <- map(rfObjs\_model2, rfImportanceHelper)  
# Performance metrics  
rfMetrics\_model2 <- map(rfPredictions\_model2, confusionMatrix, reference = factor(trainTest\_model2$testingData$BCaseControlStatus))   
rfPlotMetrics\_model2 <- data.frame(reduce(map(1:5, ~rfMetrics\_model2[[.]]$byClass), rbind)) %>%   
 mutate(Method = "RF")   
  
## Model 3: Demographic and clinical variables (Model 1) + behavioral and lifestyle factors (Model 2) + neurocognitive factors ----------  
# Fit the random forest to the training data  
rfObjs\_model3 <- map(trainTest\_model3$trainingSets\_list, rfHelper)  
# Predict on the test set  
rfPredictions\_model3 <- map(rfObjs\_model3, predict, trainTest\_model3$testingData %>%  
 dplyr::select(-c(BCaseControlStatus, PtID)))  
# Extract variable importance  
rfImportance\_model3 <- map(rfObjs\_model3, rfImportanceHelper)  
# Performance metrics  
rfMetrics\_model3 <- map(rfPredictions\_model3, confusionMatrix, reference = factor(trainTest\_model3$testingData$BCaseControlStatus))   
rfPlotMetrics\_model3 <- data.frame(reduce(map(1:5, ~rfMetrics\_model3[[.]]$byClass), rbind)) %>%   
 mutate(Method = "RF")   
  
## Model 4: Demographic and clinical variables (Model 1) + behavior and lifestyle factors (Model 2) + neurocognitive factors (Model 3) + CGM metrics -----------------------------------------------------------------------------------------------------------------------------------------  
# Fit the random forest to the training data  
rfObjs\_model4 <- map(trainTest\_model4$trainingSets\_list, rfHelper)  
# Predict on the test set  
rfPredictions\_model4 <- map(rfObjs\_model4, predict, trainTest\_model4$testingData %>%  
 dplyr::select(-c(BCaseControlStatus, PtID)))  
# Extract variable importance  
rfImportance\_model4 <- map(rfObjs\_model4, rfImportanceHelper)  
# Performance metrics  
rfMetrics\_model4 <- map(rfPredictions\_model4, confusionMatrix, reference = factor(trainTest\_model4$testingData$BCaseControlStatus))   
rfPlotMetrics\_model4 <- data.frame(reduce(map(1:5, ~rfMetrics\_model4[[.]]$byClass), rbind)) %>%   
 mutate(Method = "RF")

## L1-regularized Logistic Regression

## Model 1: Demographic and clinical variables ----------------------------------------------------------------  
# Fit the model on the training data  
regLogisticRegObj\_model1 <- map(trainTest\_model1$trainingSets\_list, regLogisticReg\_helper\_model1)   
# Predict on the test data  
regLogisticRegPredictions\_model1 <- map(regLogisticRegObj\_model1, predict, convertTestCategoricalToBinary\_model1(trainTest\_model1$testingData))   
# Extract variable importance  
regLogisticRegImportance\_model1 <- map(regLogisticRegObj\_model1, varImp)   
# Get performance metrics  
regLogMetrics\_model1 <- map(regLogisticRegPredictions\_model1, confusionMatrix, reference = factor(trainTest\_model1$testingData$BCaseControlStatus))  
regLogPlotMetrics\_model1 <- data.frame(reduce(map(1:5, ~regLogMetrics\_model1[[.]]$byClass), rbind)) %>%   
 mutate(Method = "LR")   
  
## Model 2: Demographic and clinical variables (Model 1) + behavioral and lifestyle factors --------------------  
# Fit the model on the training data  
regLogisticRegObj\_model2 <- map(trainTest\_model2$trainingSets\_list, regLogisticReg\_helper\_model2)   
# Predict on the test data  
regLogisticRegPredictions\_model2 <- map(regLogisticRegObj\_model2, predict, convertTestCategoricalToBinary\_model2(trainTest\_model2$testingData))   
# Extract variable importance  
regLogisticRegImportance\_model2 <- map(regLogisticRegObj\_model2, varImp)   
# Get performance metrics  
regLogMetrics\_model2 <- map(regLogisticRegPredictions\_model2, confusionMatrix, reference = factor(trainTest\_model2$testingData$BCaseControlStatus))  
regLogPlotMetrics\_model2 <- data.frame(reduce(map(1:5, ~regLogMetrics\_model2[[.]]$byClass), rbind)) %>%   
 mutate(Method = "LR")   
  
## Model 3: Demographic and clinical variables (Model 1) + behavioral and lifestyle factors (Model 2) + neurocognitive factors ----------  
# Fit the model on the training data  
regLogisticRegObj\_model3 <- map(trainTest\_model3$trainingSets\_list, regLogisticReg\_helper\_model2)   
# Predict on the test data  
regLogisticRegPredictions\_model3 <- map(regLogisticRegObj\_model3, predict, convertTestCategoricalToBinary\_model2(trainTest\_model3$testingData))   
# Extract variable importance  
regLogisticRegImportance\_model3 <- map(regLogisticRegObj\_model3, varImp)   
# Get performance metrics  
regLogMetrics\_model3 <- map(regLogisticRegPredictions\_model3, confusionMatrix, reference = factor(trainTest\_model3$testingData$BCaseControlStatus))  
regLogPlotMetrics\_model3 <- data.frame(reduce(map(1:5, ~regLogMetrics\_model3[[.]]$byClass), rbind)) %>%   
 mutate(Method = "LR")   
  
## Model 4: Demographic and clinical variables (Model 1) + behavior and lifestyle factors (Model 2) + neurocognitive factors (Model 3) + CGM metrics -----------------------------------------------------------------------------------------------------------------------------------------  
regLogisticRegObj\_model4 <- map(trainTest\_model4$trainingSets\_list, regLogisticReg\_helper\_model2)   
# Predict on the test data  
regLogisticRegPredictions\_model4 <- map(regLogisticRegObj\_model4, predict, convertTestCategoricalToBinary\_model2(trainTest\_model4$testingData))   
# Extract variable importance  
regLogisticRegImportance\_model4 <- map(regLogisticRegObj\_model4, varImp)   
# Get performance metrics  
regLogMetrics\_model4 <- map(regLogisticRegPredictions\_model4, confusionMatrix, reference = factor(trainTest\_model4$testingData$BCaseControlStatus))  
regLogPlotMetrics\_model4 <- data.frame(reduce(map(1:5, ~regLogMetrics\_model4[[.]]$byClass), rbind)) %>%   
 mutate(Method = "LR")

## Performance

rfPlotMetrics\_model1 %>% summarise\_all(mean) %>%  
 mutate\_all(round, 3)

## Warning in mean.default(Method): argument is not numeric or logical: returning  
## NA

## Sensitivity Specificity Pos.Pred.Value Neg.Pred.Value Precision Recall F1  
## 1 0.727 0.62 0.679 0.673 0.679 0.727 0.702  
## Prevalence Detection.Rate Detection.Prevalence Balanced.Accuracy Method  
## 1 0.524 0.381 0.562 0.674 NA

rfPlotMetrics\_model2 %>% summarise\_all(mean) %>%  
 mutate\_all(round, 3)

## Warning in mean.default(Method): argument is not numeric or logical: returning  
## NA

## Sensitivity Specificity Pos.Pred.Value Neg.Pred.Value Precision Recall F1  
## 1 0.727 0.43 0.584 0.589 0.584 0.727 0.648  
## Prevalence Detection.Rate Detection.Prevalence Balanced.Accuracy Method  
## 1 0.524 0.381 0.652 0.579 NA

rfPlotMetrics\_model3 %>% summarise\_all(mean) %>%  
 mutate\_all(round, 3)

## Warning in mean.default(Method): argument is not numeric or logical: returning  
## NA

## Sensitivity Specificity Pos.Pred.Value Neg.Pred.Value Precision Recall F1  
## 1 0.827 0.64 0.716 0.772 0.716 0.827 0.768  
## Prevalence Detection.Rate Detection.Prevalence Balanced.Accuracy Method  
## 1 0.524 0.433 0.605 0.734 NA

rfPlotMetrics\_model4 %>% summarise\_all(mean) %>%  
 mutate\_all(round, 3)

## Warning in mean.default(Method): argument is not numeric or logical: returning  
## NA

## Sensitivity Specificity Pos.Pred.Value Neg.Pred.Value Precision Recall F1  
## 1 0.855 0.64 0.724 0.799 0.724 0.855 0.784  
## Prevalence Detection.Rate Detection.Prevalence Balanced.Accuracy Method  
## 1 0.524 0.448 0.619 0.747 NA

regLogPlotMetrics\_model1 %>% summarise\_all(mean) %>%  
 mutate\_all(round, 3)

## Warning in mean.default(Method): argument is not numeric or logical: returning  
## NA

## Sensitivity Specificity Pos.Pred.Value Neg.Pred.Value Precision Recall F1  
## 1 0.773 0.65 0.708 0.722 0.708 0.773 0.739  
## Prevalence Detection.Rate Detection.Prevalence Balanced.Accuracy Method  
## 1 0.524 0.405 0.571 0.711 NA

regLogPlotMetrics\_model2 %>% summarise\_all(mean) %>%  
 mutate\_all(round, 3)

## Warning in mean.default(Method): argument is not numeric or logical: returning  
## NA

## Sensitivity Specificity Pos.Pred.Value Neg.Pred.Value Precision Recall F1  
## 1 0.764 0.63 0.695 0.708 0.695 0.764 0.727  
## Prevalence Detection.Rate Detection.Prevalence Balanced.Accuracy Method  
## 1 0.524 0.4 0.576 0.697 NA

regLogPlotMetrics\_model3 %>% summarise\_all(mean) %>%  
 mutate\_all(round, 3)

## Warning in mean.default(Method): argument is not numeric or logical: returning  
## NA

## Sensitivity Specificity Pos.Pred.Value Neg.Pred.Value Precision Recall F1  
## 1 0.755 0.67 0.718 0.713 0.718 0.755 0.735  
## Prevalence Detection.Rate Detection.Prevalence Balanced.Accuracy Method  
## 1 0.524 0.395 0.552 0.712 NA

regLogPlotMetrics\_model4 %>% summarise\_all(mean) %>%  
 mutate\_all(round, 3)

## Warning in mean.default(Method): argument is not numeric or logical: returning  
## NA

## Sensitivity Specificity Pos.Pred.Value Neg.Pred.Value Precision Recall F1  
## 1 0.755 0.64 0.699 0.704 0.699 0.755 0.725  
## Prevalence Detection.Rate Detection.Prevalence Balanced.Accuracy Method  
## 1 0.524 0.395 0.567 0.697 NA

## Individual level predictions

head(trainTest\_model1$testingData)

## # A tibble: 6 × 15  
## # Rowwise:   
## PtID BCaseContr…¹ Gender nonHi…² educa…³ insur…⁴ bmiCat insDo…⁵ gluco…⁶ HBA1C  
## <dbl> <dbl> <chr> <dbl> <fct> <fct> <fct> <fct> <fct> <dbl>  
## 1 53 0 M 1 Advanc… Only g… obese greate… 5-6 6.6  
## 2 95 1 F 1 College Only g… under… greate… 4 6.2  
## 3 22 0 F 1 College Govern… overw… 40to60 4 8.4  
## 4 90 0 M 1 College Govern… obese 40to60 4 6.6  
## 5 64 0 F 1 HS Only g… obese less40 1-3 8.9  
## 6 5 0 M 1 Advanc… Only c… obese greate… 1-3 9.1  
## # … with 5 more variables: detectableCPEP <dbl>, betaBlocker <dbl>,  
## # abnormalCreatinine <dbl>, frailty <dbl>, hypoUnaware <fct>, and abbreviated  
## # variable names ¹​BCaseControlStatus, ²​nonHispWhite, ³​educationCat,  
## # ⁴​insurance, ⁵​insDoseCat, ⁶​glucoseMonitoringCat

individualDF1 <- trainTest\_model1$testingData %>% ungroup() %>% slice(1) %>%  
 mutate(Gender = "M",  
 nonHispWhite = 1,  
 educationCat = factor("HS", levels = c("College", "HS", "Advanced degree")),  
 insurance = factor("Only government", levels = c("Only government", "Only commercial", "Government and commercial", "None")),   
 bmiCat = factor("underweight or normal weight", levels = c("obese", "overweight", "underweight or normal weight")),  
 insDoseCat = factor("less40", levels = c("less40", "greater60", "40to60")),  
 glucoseMonitoringCat = factor(">=10", levels = c("0", "1-3", "4", "5-6", "7-9", ">=10")),  
 HBA1C = 5.0,  
 detectableCPEP = 1,   
 betaBlocker = 1,   
 abnormalCreatinine = 1,   
 frailty = 10,   
 hypoUnaware = factor("hypo aware", levels = c("reduced awareness", "hypo aware", "unaware"))) %>%  
 select(-c(PtID, BCaseControlStatus))  
reduce(map(rfObjs\_model1, predict, individualDF1), c)

## 1 1 1 1 1   
## 0 1 1 1 1   
## Levels: 0 1

individualDF2 <- trainTest\_model1$testingData %>% ungroup() %>% slice(1) %>%  
 mutate(Gender = "F",  
 nonHispWhite = 1,  
 educationCat = factor("HS", levels = c("College", "HS", "Advanced degree")),  
 insurance = factor("Only government", levels = c("Only government", "Only commercial", "Government and commercial", "None")),   
 bmiCat = factor("overweight", levels = c("obese", "overweight", "underweight or normal weight")),  
 insDoseCat = factor("40to60", levels = c("less40", "greater60", "40to60")),  
 glucoseMonitoringCat = factor("4", levels = c("0", "1-3", "4", "5-6", "7-9", ">=10")),  
 HBA1C = 7.5,  
 detectableCPEP = 1,   
 betaBlocker = 1,   
 abnormalCreatinine = 1,   
 frailty = 3,   
 hypoUnaware = factor("hypo aware", levels = c("reduced awareness", "hypo aware", "unaware"))) %>%  
 select(-c(PtID, BCaseControlStatus))  
reduce(map(rfObjs\_model1, predict, individualDF2), c)

## 1 1 1 1 1   
## 1 1 1 1 1   
## Levels: 0 1

individualDF3 <- trainTest\_model1$testingData %>% ungroup() %>% slice(1) %>%  
 mutate(Gender = "F",  
 nonHispWhite = 1,  
 educationCat = factor("Advanced degree", levels = c("College", "HS", "Advanced degree")),  
 insurance = factor("Only government", levels = c("Only government", "Only commercial", "Government and commercial", "None")),   
 bmiCat = factor("underweight or normal weight", levels = c("obese", "overweight", "underweight or normal weight")),  
 insDoseCat = factor("40to60", levels = c("less40", "greater60", "40to60")),  
 glucoseMonitoringCat = factor("7-9", levels = c("0", "1-3", "4", "5-6", "7-9", ">=10")),  
 HBA1C = 5.0,  
 detectableCPEP = 0,   
 betaBlocker = 0,   
 abnormalCreatinine = 0,   
 frailty = 1,   
 hypoUnaware = factor("reduced awareness", levels = c("reduced awareness", "hypo aware", "unaware"))) %>%  
 select(-c(PtID, BCaseControlStatus))  
reduce(map(rfObjs\_model1, predict, individualDF3), c)

## 1 1 1 1 1   
## 1 1 1 1 1   
## Levels: 0 1

## Tree visualization

getTree(rfObjs\_model1[[1]], k = 1, labelVar = TRUE)

## left daughter right daughter split var split point status  
## 1 2 3 educationCat 1.00 1  
## 2 4 5 glucoseMonitoringCat 2.00 1  
## 3 6 7 glucoseMonitoringCat 15.00 1  
## 4 0 0 <NA> 0.00 -1  
## 5 8 9 frailty 3.25 1  
## 6 10 11 hypoUnaware 2.00 1  
## 7 12 13 insurance 3.00 1  
## 8 14 15 glucoseMonitoringCat 4.00 1  
## 9 16 17 insDoseCat 2.00 1  
## 10 18 19 educationCat 2.00 1  
## 11 20 21 insurance 1.00 1  
## 12 0 0 <NA> 0.00 -1  
## 13 0 0 <NA> 0.00 -1  
## 14 22 23 insDoseCat 2.00 1  
## 15 24 25 bmiCat 3.00 1  
## 16 0 0 <NA> 0.00 -1  
## 17 26 27 hypoUnaware 2.00 1  
## 18 28 29 glucoseMonitoringCat 7.00 1  
## 19 30 31 HBA1C 7.70 1  
## 20 32 33 frailty 3.75 1  
## 21 34 35 detectableCPEP 0.50 1  
## 22 0 0 <NA> 0.00 -1  
## 23 36 37 Gender 1.50 1  
## 24 0 0 <NA> 0.00 -1  
## 25 38 39 glucoseMonitoringCat 1.00 1  
## 26 40 41 HBA1C 8.75 1  
## 27 42 43 insurance 7.00 1  
## 28 0 0 <NA> 0.00 -1  
## 29 0 0 <NA> 0.00 -1  
## 30 0 0 <NA> 0.00 -1  
## 31 44 45 glucoseMonitoringCat 2.00 1  
## 32 46 47 HBA1C 7.45 1  
## 33 0 0 <NA> 0.00 -1  
## 34 48 49 HBA1C 7.35 1  
## 35 0 0 <NA> 0.00 -1  
## 36 50 51 frailty 2.25 1  
## 37 52 53 detectableCPEP 0.50 1  
## 38 54 55 detectableCPEP 0.50 1  
## 39 56 57 insurance 2.00 1  
## 40 58 59 insurance 1.00 1  
## 41 60 61 frailty 5.25 1  
## 42 62 63 insDoseCat 1.00 1  
## 43 0 0 <NA> 0.00 -1  
## 44 0 0 <NA> 0.00 -1  
## 45 0 0 <NA> 0.00 -1  
## 46 0 0 <NA> 0.00 -1  
## 47 0 0 <NA> 0.00 -1  
## 48 0 0 <NA> 0.00 -1  
## 49 64 65 HBA1C 9.00 1  
## 50 66 67 HBA1C 8.55 1  
## 51 0 0 <NA> 0.00 -1  
## 52 0 0 <NA> 0.00 -1  
## 53 68 69 betaBlocker 0.50 1  
## 54 70 71 HBA1C 6.65 1  
## 55 0 0 <NA> 0.00 -1  
## 56 0 0 <NA> 0.00 -1  
## 57 0 0 <NA> 0.00 -1  
## 58 72 73 HBA1C 6.50 1  
## 59 0 0 <NA> 0.00 -1  
## 60 0 0 <NA> 0.00 -1  
## 61 0 0 <NA> 0.00 -1  
## 62 0 0 <NA> 0.00 -1  
## 63 74 75 Gender 1.50 1  
## 64 76 77 frailty 3.25 1  
## 65 0 0 <NA> 0.00 -1  
## 66 0 0 <NA> 0.00 -1  
## 67 0 0 <NA> 0.00 -1  
## 68 0 0 <NA> 0.00 -1  
## 69 0 0 <NA> 0.00 -1  
## 70 0 0 <NA> 0.00 -1  
## 71 0 0 <NA> 0.00 -1  
## 72 0 0 <NA> 0.00 -1  
## 73 0 0 <NA> 0.00 -1  
## 74 0 0 <NA> 0.00 -1  
## 75 0 0 <NA> 0.00 -1  
## 76 0 0 <NA> 0.00 -1  
## 77 0 0 <NA> 0.00 -1  
## prediction  
## 1 <NA>  
## 2 <NA>  
## 3 <NA>  
## 4 0  
## 5 <NA>  
## 6 <NA>  
## 7 <NA>  
## 8 <NA>  
## 9 <NA>  
## 10 <NA>  
## 11 <NA>  
## 12 1  
## 13 1  
## 14 <NA>  
## 15 <NA>  
## 16 1  
## 17 <NA>  
## 18 <NA>  
## 19 <NA>  
## 20 <NA>  
## 21 <NA>  
## 22 1  
## 23 <NA>  
## 24 0  
## 25 <NA>  
## 26 <NA>  
## 27 <NA>  
## 28 0  
## 29 1  
## 30 0  
## 31 <NA>  
## 32 <NA>  
## 33 1  
## 34 <NA>  
## 35 1  
## 36 <NA>  
## 37 <NA>  
## 38 <NA>  
## 39 <NA>  
## 40 <NA>  
## 41 <NA>  
## 42 <NA>  
## 43 0  
## 44 0  
## 45 1  
## 46 1  
## 47 0  
## 48 1  
## 49 <NA>  
## 50 <NA>  
## 51 1  
## 52 0  
## 53 <NA>  
## 54 <NA>  
## 55 0  
## 56 1  
## 57 0  
## 58 <NA>  
## 59 0  
## 60 1  
## 61 0  
## 62 1  
## 63 <NA>  
## 64 <NA>  
## 65 0  
## 66 0  
## 67 1  
## 68 1  
## 69 0  
## 70 1  
## 71 0  
## 72 1  
## 73 0  
## 74 1  
## 75 0  
## 76 0  
## 77 1

table1df <- analysisWithMissing %>%  
 filter(str\_detect(exclude, "No")) %>%  
 mutate(Gender = factor(Gender, levels = c("F", "M"), labels = c("Female", "Male")),  
 `Non-Hispanic White` = factor(nonHispWhite, levels = c(1, 0), labels = c("Yes", "No")),  
 `Exercise (days per week)` = DaysWkEx,  
 `Lives alone` = LiveAlone,  
 `Insulin delivery method` = InsDeliveryMethod,  
 # `Units Insulin (bolus or long acting)` = UnitsInsBasalOrLongAct,  
 `Number boluses or injections short acting per day` = NumPumpBolusOrShortAct,  
 `Hospitalized DKA in last year` = factor(hospWithDKA, levels = c(0, 1),  
 labels = c("No", "Yes")),  
 HBA1C = as.numeric(HBA1C),  
 # `Glucose` = GLU,  
 # `CGM %Glucose <70 (day)` = bgLess70\_pct\_day,  
 # `CGM %Glucose <70 (night)` = bgLess70\_pct\_night,  
 # `CGM %Glucose >180 (day)` = bgGreater180\_pct\_day,  
 # `CGM %Glucose >180 (night)` = bgGreater180\_pct\_night,  
 # `Mean Glucose (day)` = meanGlucose\_day,  
 # `Mean Glucose (night)` = meanGlucose\_night,  
 # `%CV (day)` = pctCV\_day,  
 # `%CV (night)` = pctCV\_night,  
 `Beta blocker use` = factor(betaBlocker, levels = c(1, 0), labels = c("Yes", "no")),  
 `Symbolic Digits Written test` = SymbDigWTotCorr,  
 `Symbolic Digits Oral test` = SymbDigOTotCorr,  
 # `Hopkins Verbal Learning test - Total` = hopkinsTotal,  
 # `Hopkins Verbal Learning test - Recall` = hopkinsRecall,  
 `Trail Making test A` = TrailMakATotTime,  
 `Trail Making test B` = TrailMakBTotTime,  
 `Grooved Peg Board test (dominant hand)` = GrPegDomTotTime,  
 # `Functional Activity score` = FuncActTotTestScore,  
 `Duke Social Support scale` = DukeSocTotDSSI,  
 # `Geriatric Depression score` = geriDepressionScore,  
 `Hyperglycemia fear score` = hyperFear,  
 `Hypoglycemia unawareness` = factor(hypoUnaware, levels = c("hypo aware", "reduced awareness", "unaware"),  
 labels = c("Aware", "Reduced awareness", "Unaware")),  
 # `Diabetes Numeracy score` = diabNumeracyScore,  
 `Montreal Cognitive Assessment score` = MoCATotal,  
 `Education` = factor(educationCat, levels = c("HS", "College", "Advanced degree"),  
 labels = c("<= HS", "Any college", "Advanced degree")),  
 `Insurance` = factor(insurance, levels = c("Government and commercial", "Only commercial", "Only government", "None"),  
 labels = c("Government and commercial", "Only commercial", "Only government", "None")),  
 `Annual Income` = factor(annualIncomeCat, levels = c("<35k", "35-50k", "50-100k", ">100k"),  
 labels = c("<$35,000", "$35,000 to < $50,000", "$50,000 to < $100,000", ">= $100,000")),  
 # `Alcohol use (days per month)` = DaysMoDrinkAlc,  
 # `At least 1 day/month binge drinking` = factor(bingeAlc, levels = c(0, 1),  
 # labels = c("No", "Yes")),  
 `Insulin dose` = factor(insDoseCat, levels = c("less40", "40to60", "greater60"),  
 labels = c("<40", "40-60", ">60")),  
 `Home blood glucose monitoring (times/day)` = factor(glucoseMonitoringCat,  
 levels = c("0", "1-3", "4", "5-6", "7-9", ">=10"),  
 labels = c("0", "1-3", "4", "5-6", "7-9", ">=10")),  
 `Detectable C-peptide` = factor(detectableCPEP, levels = c(0, 1),  
 labels = c("<0.017", ">=0.017")),  
 `Abnormal creatinine` = factor(abnormalCreatinine, levels = c(0, 1),  
 labels = c("<=1.1 females/<=1.2 males", ">1.1 females/>1.2 males")),  
 `Average frailty walk time` = frailty,  
 `BMI category` = factor(bmiCat, levels = c("underweight or normal weight", "overweight", "obese"),  
 labels = c("Underweight or normal weight", "Overweight", "Obese")))  
  
table1\_out <- table1(~ Gender + `Non-Hispanic White` +  
 `Education` + `Annual Income` + `Insurance` + `BMI category` +  
 `Exercise (days per week)` +  
 `Lives alone` +  
 # `Alcohol use (days per month)` +  
 # `At least 1 day/month binge drinking` +  
 `Insulin delivery method` +  
 `Insulin dose` +  
 `Number boluses or injections short acting per day` +  
 `Home blood glucose monitoring (times/day)` +  
 `Hospitalized DKA in last year` + HBA1C + `Detectable C-peptide` +  
 # Glucose +  
 # `CGM %Glucose <70 (day)` +  
 # `CGM %Glucose <70 (night)` +  
 # `CGM %Glucose >180 (day)` +  
 # `CGM %Glucose >180 (night)` +  
 # `Mean Glucose (day)` +  
 # `Mean Glucose (night)` +  
 # `%CV (day)` +  
 # `%CV (night)` +  
 `Abnormal creatinine` + `Beta blocker use` +  
 `Symbolic Digits Written test` + `Symbolic Digits Oral test` +  
 # `Hopkins Verbal Learning test - Total` + `Hopkins Verbal Learning test - Recall` +  
 `Trail Making test A` + `Trail Making test B` +  
 `Grooved Peg Board test (dominant hand)` +  
 # `Functional Activity score` +  
 `Duke Social Support scale` +  
 # `Geriatric Depression score` +  
 `Hypoglycemia unawareness` +  
 `Hyperglycemia fear score` +  
 # `Diabetes Numeracy score` +  
 `Montreal Cognitive Assessment score` +  
 `Average frailty walk time`  
 | BCaseControlStatus, data = table1df)  
# Knitted table  
table1\_out

## Get nicer `table1` .docx output by simply installing the `flextable` package

##   Case  
## 1 (N=95)  
## 2 Gender   
## 3   Female 48 (50.5%)  
## 4   Male 47 (49.5%)  
## 5 Non-Hispanic White   
## 6   Yes 89 (93.7%)  
## 7   No 6 (6.3%)  
## 8 Education   
## 9   <= HS 13 (13.7%)  
## 10   Any college 54 (56.8%)  
## 11   Advanced degree 27 (28.4%)  
## 12   Missing 1 (1.1%)  
## 13 Annual Income   
## 14   <$35,000 22 (23.2%)  
## 15   $35,000 to < $50,000 10 (10.5%)  
## 16   $50,000 to < $100,000 29 (30.5%)  
## 17   >= $100,000 26 (27.4%)  
## 18   Missing 8 (8.4%)  
## 19 Insurance   
## 20   Government and commercial 34 (35.8%)  
## 21   Only commercial 24 (25.3%)  
## 22   Only government 35 (36.8%)  
## 23   None 1 (1.1%)  
## 24   Missing 1 (1.1%)  
## 25 BMI category   
## 26   Underweight or normal weight 34 (35.8%)  
## 27   Overweight 35 (36.8%)  
## 28   Obese 24 (25.3%)  
## 29   Missing 2 (2.1%)  
## 30 Exercise (days per week)   
## 31   Mean (SD) 5.09 (2.10)  
## 32   Median [Min, Max] 6.00 [0, 7.00]  
## 33   Missing 2 (2.1%)  
## 34 Lives alone   
## 35   No 72 (75.8%)  
## 36   Yes 23 (24.2%)  
## 37 Insulin delivery method   
## 38   Injections 38 (40.0%)  
## 39   Pump 57 (60.0%)  
## 40 Insulin dose   
## 41   <40 43 (45.3%)  
## 42   40-60 21 (22.1%)  
## 43   >60 19 (20.0%)  
## 44   Missing 12 (12.6%)  
## 45 Number boluses or injections short acting per day   
## 46   Mean (SD) 5.00 (4.32)  
## 47   Median [Min, Max] 4.00 [2.00, 25.0]  
## 48   Missing 19 (20.0%)  
## 49 Home blood glucose monitoring (times/day)   
## 50   0 1 (1.1%)  
## 51   1-3 5 (5.3%)  
## 52   4 20 (21.1%)  
## 53   5-6 37 (38.9%)  
## 54   7-9 19 (20.0%)  
## 55   >=10 13 (13.7%)  
## 56 Hospitalized DKA in last year   
## 57   No 87 (91.6%)  
## 58   Yes 7 (7.4%)  
## 59   Missing 1 (1.1%)  
## 60 HBA1C   
## 61   Mean (SD) 7.75 (1.36)  
## 62   Median [Min, Max] 7.70 [3.30, 11.0]  
## 63 Detectable C-peptide   
## 64   <0.017 75 (78.9%)  
## 65   >=0.017 18 (18.9%)  
## 66   Missing 2 (2.1%)  
## 67 Abnormal creatinine   
## 68   <=1.1 females/<=1.2 males 76 (80.0%)  
## 69   >1.1 females/>1.2 males 17 (17.9%)  
## 70   Missing 2 (2.1%)  
## 71 Beta blocker use   
## 72   Yes 39 (41.1%)  
## 73   no 55 (57.9%)  
## 74   Missing 1 (1.1%)  
## 75 Symbolic Digits Written test   
## 76   Mean (SD) 36.7 (10.6)  
## 77   Median [Min, Max] 35.0 [12.0, 66.0]  
## 78   Missing 4 (4.2%)  
## 79 Symbolic Digits Oral test   
## 80   Mean (SD) 42.8 (11.6)  
## 81   Median [Min, Max] 41.0 [16.0, 74.0]  
## 82   Missing 4 (4.2%)  
## 83 Trail Making test A   
## 84   Mean (SD) 39.1 (12.8)  
## 85   Median [Min, Max] 39.0 [15.0, 82.0]  
## 86 Trail Making test B   
## 87   Mean (SD) 113 (54.0)  
## 88   Median [Min, Max] 102 [39.0, 300]  
## 89   Missing 2 (2.1%)  
## 90 Grooved Peg Board test (dominant hand)   
## 91   Mean (SD) 108 (40.7)  
## 92   Median [Min, Max] 97.0 [64.0, 261]  
## 93   Missing 1 (1.1%)  
## 94 Duke Social Support scale   
## 95   Mean (SD) 27.6 (3.60)  
## 96   Median [Min, Max] 28.0 [15.0, 33.0]  
## 97   Missing 1 (1.1%)  
## 98 Hypoglycemia unawareness   
## 99   Aware 25 (26.3%)  
## 100   Reduced awareness 62 (65.3%)  
## 101   Unaware 6 (6.3%)  
## 102   Missing 2 (2.1%)  
## 103 Hyperglycemia fear score   
## 104   Mean (SD) 14.5 (3.53)  
## 105   Median [Min, Max] 14.0 [5.00, 24.0]  
## 106   Missing 2 (2.1%)  
## 107 Montreal Cognitive Assessment score   
## 108   Mean (SD) 25.3 (3.12)  
## 109   Median [Min, Max] 26.0 [13.0, 31.0]  
## 110   Missing 1 (1.1%)  
## 111 Average frailty walk time   
## 112   Mean (SD) 3.49 (1.05)  
## 113   Median [Min, Max] 3.25 [2.00, 7.50]  
## 114   Missing 1 (1.1%)  
## Control Overall  
## 1 (N=97) (N=192)  
## 2   
## 3 43 (44.3%) 91 (47.4%)  
## 4 54 (55.7%) 101 (52.6%)  
## 5   
## 6 88 (90.7%) 177 (92.2%)  
## 7 9 (9.3%) 15 (7.8%)  
## 8   
## 9 9 (9.3%) 22 (11.5%)  
## 10 64 (66.0%) 118 (61.5%)  
## 11 23 (23.7%) 50 (26.0%)  
## 12 1 (1.0%) 2 (1.0%)  
## 13   
## 14 20 (20.6%) 42 (21.9%)  
## 15 9 (9.3%) 19 (9.9%)  
## 16 33 (34.0%) 62 (32.3%)  
## 17 20 (20.6%) 46 (24.0%)  
## 18 15 (15.5%) 23 (12.0%)  
## 19   
## 20 34 (35.1%) 68 (35.4%)  
## 21 34 (35.1%) 58 (30.2%)  
## 22 27 (27.8%) 62 (32.3%)  
## 23 2 (2.1%) 3 (1.6%)  
## 24 0 (0%) 1 (0.5%)  
## 25   
## 26 35 (36.1%) 69 (35.9%)  
## 27 37 (38.1%) 72 (37.5%)  
## 28 23 (23.7%) 47 (24.5%)  
## 29 2 (2.1%) 4 (2.1%)  
## 30   
## 31 4.98 (1.96) 5.03 (2.03)  
## 32 5.00 [0, 7.00] 5.00 [0, 7.00]  
## 33 0 (0%) 2 (1.0%)  
## 34   
## 35 75 (77.3%) 147 (76.6%)  
## 36 22 (22.7%) 45 (23.4%)  
## 37   
## 38 39 (40.2%) 77 (40.1%)  
## 39 58 (59.8%) 115 (59.9%)  
## 40   
## 41 42 (43.3%) 85 (44.3%)  
## 42 33 (34.0%) 54 (28.1%)  
## 43 17 (17.5%) 36 (18.8%)  
## 44 5 (5.2%) 17 (8.9%)  
## 45   
## 46 4.68 (4.31) 4.83 (4.31)  
## 47 4.00 [2.00, 37.0] 4.00 [2.00, 37.0]  
## 48 12 (12.4%) 31 (16.1%)  
## 49   
## 50 0 (0%) 1 (0.5%)  
## 51 17 (17.5%) 22 (11.5%)  
## 52 23 (23.7%) 43 (22.4%)  
## 53 31 (32.0%) 68 (35.4%)  
## 54 21 (21.6%) 40 (20.8%)  
## 55 5 (5.2%) 18 (9.4%)  
## 56   
## 57 95 (97.9%) 182 (94.8%)  
## 58 2 (2.1%) 9 (4.7%)  
## 59 0 (0%) 1 (0.5%)  
## 60   
## 61 7.66 (1.11) 7.71 (1.24)  
## 62 7.70 [5.40, 11.5] 7.70 [3.30, 11.5]  
## 63   
## 64 73 (75.3%) 148 (77.1%)  
## 65 24 (24.7%) 42 (21.9%)  
## 66 0 (0%) 2 (1.0%)  
## 67   
## 68 89 (91.8%) 165 (85.9%)  
## 69 8 (8.2%) 25 (13.0%)  
## 70 0 (0%) 2 (1.0%)  
## 71   
## 72 19 (19.6%) 58 (30.2%)  
## 73 77 (79.4%) 132 (68.8%)  
## 74 1 (1.0%) 2 (1.0%)  
## 75   
## 76 42.0 (10.4) 39.4 (10.8)  
## 77 43.0 [17.0, 71.0] 38.0 [12.0, 71.0]  
## 78 4 (4.1%) 8 (4.2%)  
## 79   
## 80 47.0 (11.1) 44.9 (11.5)  
## 81 47.0 [19.0, 74.0] 44.0 [16.0, 74.0]  
## 82 5 (5.2%) 9 (4.7%)  
## 83   
## 84 36.5 (15.0) 37.8 (14.0)  
## 85 34.0 [16.0, 120] 36.0 [15.0, 120]  
## 86   
## 87 93.5 (40.7) 103 (48.7)  
## 88 84.0 [38.0, 257] 91.5 [38.0, 300]  
## 89 4 (4.1%) 6 (3.1%)  
## 90   
## 91 97.8 (38.6) 103 (39.9)  
## 92 86.5 [59.0, 278] 92.0 [59.0, 278]  
## 93 1 (1.0%) 2 (1.0%)  
## 94   
## 95 28.3 (3.12) 28.0 (3.38)  
## 96 29.0 [14.0, 33.0] 29.0 [14.0, 33.0]  
## 97 0 (0%) 1 (0.5%)  
## 98   
## 99 76 (78.4%) 101 (52.6%)  
## 100 19 (19.6%) 81 (42.2%)  
## 101 1 (1.0%) 7 (3.6%)  
## 102 1 (1.0%) 3 (1.6%)  
## 103   
## 104 13.8 (3.89) 14.1 (3.72)  
## 105 14.0 [5.00, 22.0] 14.0 [5.00, 24.0]  
## 106 0 (0%) 2 (1.0%)  
## 107   
## 108 26.1 (2.86) 25.7 (3.01)  
## 109 26.0 [18.0, 30.0] 26.0 [13.0, 31.0]  
## 110 0 (0%) 1 (0.5%)  
## 111   
## 112 3.13 (0.755) 3.31 (0.929)  
## 113 3.00 [2.00, 6.50] 3.00 [2.00, 7.50]  
## 114 1 (1.0%) 2 (1.0%)

# Create the latex table  
print(xtable(as.data.frame(table1\_out)), include.rownames = FALSE)

## % latex table generated in R 4.2.3 by xtable 1.8-4 package  
## % Sun Mar 26 10:46:28 2023  
## \begin{table}[ht]  
## \centering  
## \begin{tabular}{llll}  
## \hline  
##   & Case & Control & Overall \\   
## \hline  
## & (N=95) & (N=97) & (N=192) \\   
## Gender & & & \\   
##   Female & 48 (50.5\%) & 43 (44.3\%) & 91 (47.4\%) \\   
##   Male & 47 (49.5\%) & 54 (55.7\%) & 101 (52.6\%) \\   
## Non-Hispanic White & & & \\   
##   Yes & 89 (93.7\%) & 88 (90.7\%) & 177 (92.2\%) \\   
##   No & 6 (6.3\%) & 9 (9.3\%) & 15 (7.8\%) \\   
## Education & & & \\   
##   $<$= HS & 13 (13.7\%) & 9 (9.3\%) & 22 (11.5\%) \\   
##   Any college & 54 (56.8\%) & 64 (66.0\%) & 118 (61.5\%) \\   
##   Advanced degree & 27 (28.4\%) & 23 (23.7\%) & 50 (26.0\%) \\   
##   Missing & 1 (1.1\%) & 1 (1.0\%) & 2 (1.0\%) \\   
## Annual Income & & & \\   
##   $<$\$35,000 & 22 (23.2\%) & 20 (20.6\%) & 42 (21.9\%) \\   
##   \$35,000 to $<$ \$50,000 & 10 (10.5\%) & 9 (9.3\%) & 19 (9.9\%) \\   
##   \$50,000 to $<$ \$100,000 & 29 (30.5\%) & 33 (34.0\%) & 62 (32.3\%) \\   
##   $>$= \$100,000 & 26 (27.4\%) & 20 (20.6\%) & 46 (24.0\%) \\   
##   Missing & 8 (8.4\%) & 15 (15.5\%) & 23 (12.0\%) \\   
## Insurance & & & \\   
##   Government and commercial & 34 (35.8\%) & 34 (35.1\%) & 68 (35.4\%) \\   
##   Only commercial & 24 (25.3\%) & 34 (35.1\%) & 58 (30.2\%) \\   
##   Only government & 35 (36.8\%) & 27 (27.8\%) & 62 (32.3\%) \\   
##   None & 1 (1.1\%) & 2 (2.1\%) & 3 (1.6\%) \\   
##   Missing & 1 (1.1\%) & 0 (0\%) & 1 (0.5\%) \\   
## BMI category & & & \\   
##   Underweight or normal weight & 34 (35.8\%) & 35 (36.1\%) & 69 (35.9\%) \\   
##   Overweight & 35 (36.8\%) & 37 (38.1\%) & 72 (37.5\%) \\   
##   Obese & 24 (25.3\%) & 23 (23.7\%) & 47 (24.5\%) \\   
##   Missing & 2 (2.1\%) & 2 (2.1\%) & 4 (2.1\%) \\   
## Exercise (days per week) & & & \\   
##   Mean (SD) & 5.09 (2.10) & 4.98 (1.96) & 5.03 (2.03) \\   
##   Median [Min, Max] & 6.00 [0, 7.00] & 5.00 [0, 7.00] & 5.00 [0, 7.00] \\   
##   Missing & 2 (2.1\%) & 0 (0\%) & 2 (1.0\%) \\   
## Lives alone & & & \\   
##   No & 72 (75.8\%) & 75 (77.3\%) & 147 (76.6\%) \\   
##   Yes & 23 (24.2\%) & 22 (22.7\%) & 45 (23.4\%) \\   
## Insulin delivery method & & & \\   
##   Injections & 38 (40.0\%) & 39 (40.2\%) & 77 (40.1\%) \\   
##   Pump & 57 (60.0\%) & 58 (59.8\%) & 115 (59.9\%) \\   
## Insulin dose & & & \\   
##   $<$40 & 43 (45.3\%) & 42 (43.3\%) & 85 (44.3\%) \\   
##   40-60 & 21 (22.1\%) & 33 (34.0\%) & 54 (28.1\%) \\   
##   $>$60 & 19 (20.0\%) & 17 (17.5\%) & 36 (18.8\%) \\   
##   Missing & 12 (12.6\%) & 5 (5.2\%) & 17 (8.9\%) \\   
## Number boluses or injections short acting per day & & & \\   
##   Mean (SD) & 5.00 (4.32) & 4.68 (4.31) & 4.83 (4.31) \\   
##   Median [Min, Max] & 4.00 [2.00, 25.0] & 4.00 [2.00, 37.0] & 4.00 [2.00, 37.0] \\   
##   Missing & 19 (20.0\%) & 12 (12.4\%) & 31 (16.1\%) \\   
## Home blood glucose monitoring (times/day) & & & \\   
##   0 & 1 (1.1\%) & 0 (0\%) & 1 (0.5\%) \\   
##   1-3 & 5 (5.3\%) & 17 (17.5\%) & 22 (11.5\%) \\   
##   4 & 20 (21.1\%) & 23 (23.7\%) & 43 (22.4\%) \\   
##   5-6 & 37 (38.9\%) & 31 (32.0\%) & 68 (35.4\%) \\   
##   7-9 & 19 (20.0\%) & 21 (21.6\%) & 40 (20.8\%) \\   
##   $>$=10 & 13 (13.7\%) & 5 (5.2\%) & 18 (9.4\%) \\   
## Hospitalized DKA in last year & & & \\   
##   No & 87 (91.6\%) & 95 (97.9\%) & 182 (94.8\%) \\   
##   Yes & 7 (7.4\%) & 2 (2.1\%) & 9 (4.7\%) \\   
##   Missing & 1 (1.1\%) & 0 (0\%) & 1 (0.5\%) \\   
## HBA1C & & & \\   
##   Mean (SD) & 7.75 (1.36) & 7.66 (1.11) & 7.71 (1.24) \\   
##   Median [Min, Max] & 7.70 [3.30, 11.0] & 7.70 [5.40, 11.5] & 7.70 [3.30, 11.5] \\   
## Detectable C-peptide & & & \\   
##   $<$0.017 & 75 (78.9\%) & 73 (75.3\%) & 148 (77.1\%) \\   
##   $>$=0.017 & 18 (18.9\%) & 24 (24.7\%) & 42 (21.9\%) \\   
##   Missing & 2 (2.1\%) & 0 (0\%) & 2 (1.0\%) \\   
## Abnormal creatinine & & & \\   
##   $<$=1.1 females/$<$=1.2 males & 76 (80.0\%) & 89 (91.8\%) & 165 (85.9\%) \\   
##   $>$1.1 females/$>$1.2 males & 17 (17.9\%) & 8 (8.2\%) & 25 (13.0\%) \\   
##   Missing & 2 (2.1\%) & 0 (0\%) & 2 (1.0\%) \\   
## Beta blocker use & & & \\   
##   Yes & 39 (41.1\%) & 19 (19.6\%) & 58 (30.2\%) \\   
##   no & 55 (57.9\%) & 77 (79.4\%) & 132 (68.8\%) \\   
##   Missing & 1 (1.1\%) & 1 (1.0\%) & 2 (1.0\%) \\   
## Symbolic Digits Written test & & & \\   
##   Mean (SD) & 36.7 (10.6) & 42.0 (10.4) & 39.4 (10.8) \\   
##   Median [Min, Max] & 35.0 [12.0, 66.0] & 43.0 [17.0, 71.0] & 38.0 [12.0, 71.0] \\   
##   Missing & 4 (4.2\%) & 4 (4.1\%) & 8 (4.2\%) \\   
## Symbolic Digits Oral test & & & \\   
##   Mean (SD) & 42.8 (11.6) & 47.0 (11.1) & 44.9 (11.5) \\   
##   Median [Min, Max] & 41.0 [16.0, 74.0] & 47.0 [19.0, 74.0] & 44.0 [16.0, 74.0] \\   
##   Missing & 4 (4.2\%) & 5 (5.2\%) & 9 (4.7\%) \\   
## Trail Making test A & & & \\   
##   Mean (SD) & 39.1 (12.8) & 36.5 (15.0) & 37.8 (14.0) \\   
##   Median [Min, Max] & 39.0 [15.0, 82.0] & 34.0 [16.0, 120] & 36.0 [15.0, 120] \\   
## Trail Making test B & & & \\   
##   Mean (SD) & 113 (54.0) & 93.5 (40.7) & 103 (48.7) \\   
##   Median [Min, Max] & 102 [39.0, 300] & 84.0 [38.0, 257] & 91.5 [38.0, 300] \\   
##   Missing & 2 (2.1\%) & 4 (4.1\%) & 6 (3.1\%) \\   
## Grooved Peg Board test (dominant hand) & & & \\   
##   Mean (SD) & 108 (40.7) & 97.8 (38.6) & 103 (39.9) \\   
##   Median [Min, Max] & 97.0 [64.0, 261] & 86.5 [59.0, 278] & 92.0 [59.0, 278] \\   
##   Missing & 1 (1.1\%) & 1 (1.0\%) & 2 (1.0\%) \\   
## Duke Social Support scale & & & \\   
##   Mean (SD) & 27.6 (3.60) & 28.3 (3.12) & 28.0 (3.38) \\   
##   Median [Min, Max] & 28.0 [15.0, 33.0] & 29.0 [14.0, 33.0] & 29.0 [14.0, 33.0] \\   
##   Missing & 1 (1.1\%) & 0 (0\%) & 1 (0.5\%) \\   
## Hypoglycemia unawareness & & & \\   
##   Aware & 25 (26.3\%) & 76 (78.4\%) & 101 (52.6\%) \\   
##   Reduced awareness & 62 (65.3\%) & 19 (19.6\%) & 81 (42.2\%) \\   
##   Unaware & 6 (6.3\%) & 1 (1.0\%) & 7 (3.6\%) \\   
##   Missing & 2 (2.1\%) & 1 (1.0\%) & 3 (1.6\%) \\   
## Hyperglycemia fear score & & & \\   
##   Mean (SD) & 14.5 (3.53) & 13.8 (3.89) & 14.1 (3.72) \\   
##   Median [Min, Max] & 14.0 [5.00, 24.0] & 14.0 [5.00, 22.0] & 14.0 [5.00, 24.0] \\   
##   Missing & 2 (2.1\%) & 0 (0\%) & 2 (1.0\%) \\   
## Montreal Cognitive Assessment score & & & \\   
##   Mean (SD) & 25.3 (3.12) & 26.1 (2.86) & 25.7 (3.01) \\   
##   Median [Min, Max] & 26.0 [13.0, 31.0] & 26.0 [18.0, 30.0] & 26.0 [13.0, 31.0] \\   
##   Missing & 1 (1.1\%) & 0 (0\%) & 1 (0.5\%) \\   
## Average frailty walk time & & & \\   
##   Mean (SD) & 3.49 (1.05) & 3.13 (0.755) & 3.31 (0.929) \\   
##   Median [Min, Max] & 3.25 [2.00, 7.50] & 3.00 [2.00, 6.50] & 3.00 [2.00, 7.50] \\   
##   Missing & 1 (1.1\%) & 1 (1.0\%) & 2 (1.0\%) \\   
## \hline  
## \end{tabular}  
## \end{table}