Stroke\_prediction

Yuraza\_Qadri

2023-04-09

#Installed Required library  
  
require(tidyverse)

## Loading required package: tidyverse

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.1 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.2 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.1   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

require(tidymodels)

## Loading required package: tidymodels  
## ── Attaching packages ────────────────────────────────────── tidymodels 1.0.0 ──  
## ✔ broom 1.0.4 ✔ rsample 1.1.1  
## ✔ dials 1.2.0 ✔ tune 1.1.0  
## ✔ infer 1.0.4 ✔ workflows 1.1.3  
## ✔ modeldata 1.1.0 ✔ workflowsets 1.0.1  
## ✔ parsnip 1.0.4 ✔ yardstick 1.1.0  
## ✔ recipes 1.0.5   
## ── Conflicts ───────────────────────────────────────── tidymodels\_conflicts() ──  
## ✖ scales::discard() masks purrr::discard()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ recipes::fixed() masks stringr::fixed()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ yardstick::spec() masks readr::spec()  
## ✖ recipes::step() masks stats::step()  
## • Use tidymodels\_prefer() to resolve common conflicts.

library(dplyr)  
library(skimr)  
library(gapminder)  
require(ggplot2)  
library(ggridges)

# Load the data  
stroke\_data <- read.csv("C:/Users/Lenovo/Downloads/Milliman Task-1/stroke\_data.csv")  
  
# View the first few rows of the data  
head(stroke\_data)

## id gender age married hypertension heart\_disease occupation residence  
## 1 1 Male 3 No 0 0 A Rural  
## 2 2 Male 58 Yes 1 0 B Urban  
## 3 3 Female 8 No 0 0 B Urban  
## 4 4 Female 70 Yes 0 0 B Rural  
## 5 5 Male 14 No 0 0 C Rural  
## 6 6 Female 47 Yes 0 0 B Urban  
## metric\_1 metric\_2 metric\_3 metric\_4 metric\_5 smoking\_status stroke  
## 1 95.12 18.0 1 99.35 95.12 0  
## 2 87.96 39.2 1 99.70 87.96 never smoked 0  
## 3 110.89 17.6 0 96.35 110.89 0  
## 4 69.04 35.9 0 95.52 69.04 formerly smoked 0  
## 5 161.28 19.1 1 95.10 161.28 0  
## 6 210.95 50.1 0 97.63 210.95 0

# Get the summary statistics for the data  
summary(stroke\_data)

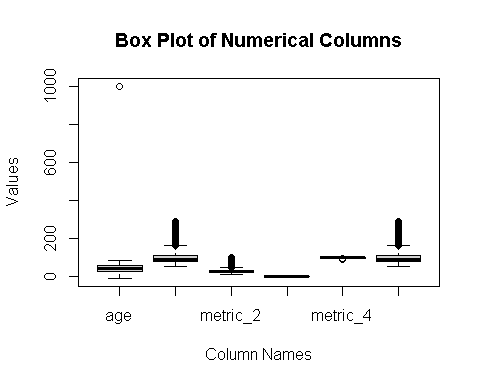
## id gender age married   
## Min. : 1 Length:43400 Min. : -10.00 Length:43400   
## 1st Qu.:10851 Class :character 1st Qu.: 24.00 Class :character   
## Median :21701 Mode :character Median : 44.00 Mode :character   
## Mean :21701 Mean : 42.26   
## 3rd Qu.:32550 3rd Qu.: 60.00   
## Max. :43400 Max. :1000.00   
##   
## hypertension heart\_disease occupation residence   
## Min. :0.00000 Min. :0.00000 Length:43400 Length:43400   
## 1st Qu.:0.00000 1st Qu.:0.00000 Class :character Class :character   
## Median :0.00000 Median :0.00000 Mode :character Mode :character   
## Mean :0.09357 Mean :0.04751   
## 3rd Qu.:0.00000 3rd Qu.:0.00000   
## Max. :1.00000 Max. :1.00000   
##   
## metric\_1 metric\_2 metric\_3 metric\_4   
## Min. : 55.00 Min. :10.10 Min. :0.0000 Min. : 87.42   
## 1st Qu.: 77.54 1st Qu.:23.20 1st Qu.:0.0000 1st Qu.: 96.59   
## Median : 91.58 Median :27.70 Median :0.0000 Median : 97.61   
## Mean :104.48 Mean :28.61 Mean :0.2899 Mean : 97.53   
## 3rd Qu.:112.07 3rd Qu.:32.90 3rd Qu.:1.0000 3rd Qu.: 98.70   
## Max. :291.05 Max. :97.60 Max. :1.0000 Max. :100.00   
## NA's :1462   
## metric\_5 smoking\_status stroke   
## Min. : 55.00 Length:43400 Min. :0.00000   
## 1st Qu.: 77.54 Class :character 1st Qu.:0.00000   
## Median : 91.58 Mode :character Median :0.00000   
## Mean :104.48 Mean :0.01804   
## 3rd Qu.:112.07 3rd Qu.:0.00000   
## Max. :291.05 Max. :1.00000   
##

# Check the Glimpse of the data  
glimpse(stroke\_data)

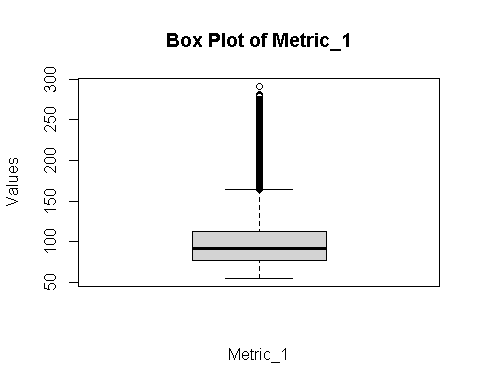
## Rows: 43,400  
## Columns: 15  
## $ id <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, …  
## $ gender <chr> "Male", "Male", "Female", "Female", "Male", "Female", "…  
## $ age <dbl> 3, 58, 8, 70, 14, 47, 52, 75, 32, 74, 79, 79, 37, 37, 4…  
## $ married <chr> "No", "Yes", "No", "Yes", "No", "Yes", "Yes", "Yes", "Y…  
## $ hypertension <int> 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0…  
## $ heart\_disease <int> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0…  
## $ occupation <chr> "A", "B", "B", "B", "C", "B", "B", "D", "B", "D", "E", …  
## $ residence <chr> "Rural", "Urban", "Urban", "Rural", "Rural", "Urban", "…  
## $ metric\_1 <dbl> 95.12, 87.96, 110.89, 69.04, 161.28, 210.95, 77.59, 243…  
## $ metric\_2 <dbl> 18.0, 39.2, 17.6, 35.9, 19.1, 50.1, 17.7, 27.0, 32.3, 5…  
## $ metric\_3 <int> 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…  
## $ metric\_4 <dbl> 99.35, 99.70, 96.35, 95.52, 95.10, 97.63, 96.46, 98.47,…  
## $ metric\_5 <dbl> 95.12, 87.96, 110.89, 69.04, 161.28, 210.95, 77.59, 243…  
## $ smoking\_status <chr> "", "never smoked", "", "formerly smoked", "", "", "for…  
## $ stroke <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…

#Removing ID variable because it contains unique values.  
stroke\_data=stroke\_data%>%select(-id)

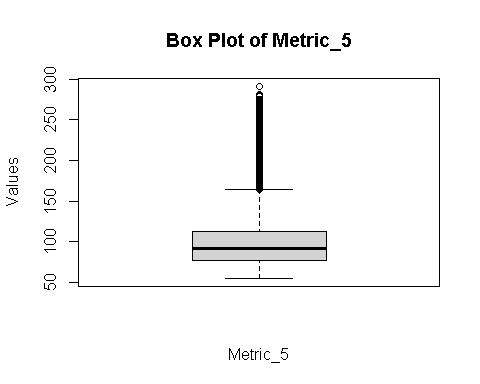
# Step-1 Data Cleaning  
  
# I have check Column names in Excel and it looks good.  
  
#Covert age variable from double to integer   
stroke\_data$age <- as.integer(stroke\_data$age)  
  
# Select the numerical columns for box plot  
num\_cols <- c("age", "metric\_1", "metric\_2", "metric\_3", "metric\_4", "metric\_5")  
  
# Create the box plot  
boxplot(stroke\_data[, num\_cols],   
 main = "Box Plot of Numerical Columns",  
 xlab = "Column Names",  
 ylab = "Values")



# Create box plot for metric\_1 as there are some outliers  
boxplot(stroke\_data$metric\_1,   
 main = "Box Plot of Metric\_1",  
 xlab = "Metric\_1",  
 ylab = "Values")



# Create box plot for metric\_5 as there are some outliers  
boxplot(stroke\_data$metric\_5,   
 main = "Box Plot of Metric\_5",  
 xlab = "Metric\_5",  
 ylab = "Values")



# But after plotting the box plot separately Metric 1 is glucose in blood and metric 5 is Blood pressure, henced the values can be kept.

#Using Skim we can see the missingness or each columns in the dataset and we can see metric 2 which is BMI has 1426 missing columns we cannot remove all the columns as the number of columns are high so we need to check should we change it with mean or median.  
#for that we have to check the skewness of data.  
skim(stroke\_data)

Data summary

|  |  |
| --- | --- |
| Name | stroke\_data |
| Number of rows | 43400 |
| Number of columns | 14 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| character | 5 |
| numeric | 9 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

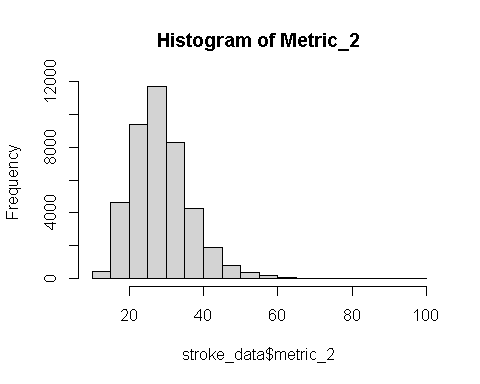
**Variable type: character**

| skim\_variable | n\_missing | complete\_rate | min | max | empty | n\_unique | whitespace |
| --- | --- | --- | --- | --- | --- | --- | --- |
| gender | 0 | 1 | 4 | 6 | 0 | 3 | 0 |
| married | 0 | 1 | 2 | 3 | 0 | 2 | 0 |
| occupation | 0 | 1 | 1 | 1 | 0 | 5 | 0 |
| residence | 0 | 1 | 5 | 5 | 0 | 2 | 0 |
| smoking\_status | 0 | 1 | 0 | 15 | 13292 | 4 | 0 |

**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| age | 0 | 1.00 | 42.25 | 23.46 | -10.00 | 24.00 | 44.00 | 60.00 | 1000.00 | ▇▁▁▁▁ |
| hypertension | 0 | 1.00 | 0.09 | 0.29 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | ▇▁▁▁▁ |
| heart\_disease | 0 | 1.00 | 0.05 | 0.21 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | ▇▁▁▁▁ |
| metric\_1 | 0 | 1.00 | 104.48 | 43.11 | 55.00 | 77.54 | 91.58 | 112.07 | 291.05 | ▇▂▁▁▁ |
| metric\_2 | 1462 | 0.97 | 28.61 | 7.77 | 10.10 | 23.20 | 27.70 | 32.90 | 97.60 | ▇▇▁▁▁ |
| metric\_3 | 0 | 1.00 | 0.29 | 0.45 | 0.00 | 0.00 | 0.00 | 1.00 | 1.00 | ▇▁▁▁▃ |
| metric\_4 | 0 | 1.00 | 97.53 | 1.47 | 87.42 | 96.59 | 97.61 | 98.70 | 100.00 | ▁▁▁▆▇ |
| metric\_5 | 0 | 1.00 | 104.48 | 43.11 | 55.00 | 77.54 | 91.58 | 112.07 | 291.05 | ▇▂▁▁▁ |
| stroke | 0 | 1.00 | 0.02 | 0.13 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | ▇▁▁▁▁ |

# As the data is normally distributed we can replace it with mean.  
# Create a histogram of metric\_2  
hist(stroke\_data$metric\_2, main = "Histogram of Metric\_2")



# here we are replacing the missing value in metric 2 with mean.  
  
# Calculate the mean of metric\_2  
mean\_metric\_2 <- mean(stroke\_data$metric\_2, na.rm = TRUE)  
  
# Replace missing values in metric\_2 with the mean  
stroke\_data$metric\_2[is.na(stroke\_data$metric\_2)] <- mean\_metric\_2

#here we can see that there are no missing values  
skim(stroke\_data)

Data summary

|  |  |
| --- | --- |
| Name | stroke\_data |
| Number of rows | 43400 |
| Number of columns | 14 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| character | 5 |
| numeric | 9 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: character**

| skim\_variable | n\_missing | complete\_rate | min | max | empty | n\_unique | whitespace |
| --- | --- | --- | --- | --- | --- | --- | --- |
| gender | 0 | 1 | 4 | 6 | 0 | 3 | 0 |
| married | 0 | 1 | 2 | 3 | 0 | 2 | 0 |
| occupation | 0 | 1 | 1 | 1 | 0 | 5 | 0 |
| residence | 0 | 1 | 5 | 5 | 0 | 2 | 0 |
| smoking\_status | 0 | 1 | 0 | 15 | 13292 | 4 | 0 |

**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| age | 0 | 1 | 42.25 | 23.46 | -10.00 | 24.00 | 44.00 | 60.00 | 1000.00 | ▇▁▁▁▁ |
| hypertension | 0 | 1 | 0.09 | 0.29 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | ▇▁▁▁▁ |
| heart\_disease | 0 | 1 | 0.05 | 0.21 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | ▇▁▁▁▁ |
| metric\_1 | 0 | 1 | 104.48 | 43.11 | 55.00 | 77.54 | 91.58 | 112.07 | 291.05 | ▇▂▁▁▁ |
| metric\_2 | 0 | 1 | 28.61 | 7.64 | 10.10 | 23.40 | 28.10 | 32.60 | 97.60 | ▇▇▁▁▁ |
| metric\_3 | 0 | 1 | 0.29 | 0.45 | 0.00 | 0.00 | 0.00 | 1.00 | 1.00 | ▇▁▁▁▃ |
| metric\_4 | 0 | 1 | 97.53 | 1.47 | 87.42 | 96.59 | 97.61 | 98.70 | 100.00 | ▁▁▁▆▇ |
| metric\_5 | 0 | 1 | 104.48 | 43.11 | 55.00 | 77.54 | 91.58 | 112.07 | 291.05 | ▇▂▁▁▁ |
| stroke | 0 | 1 | 0.02 | 0.13 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | ▇▁▁▁▁ |

# step-2 Data Splitting  
set.seed(321)  
# Create a split object  
stroke\_split <- initial\_split(stroke\_data, prop = 0.80,   
 strata = stroke)  
  
# Build training data set  
stroke\_training <- stroke\_split %>%   
 training()  
  
# Build testing data set  
stroke\_testing <- stroke\_split %>%   
 testing()

#Step-3 Feature Engineering  
  
#Creating Recipe  
  
#Creating Recipe step\_YeoJohnson: This step applies the Yeo-Johnson transformation to all numeric features to transform them into a more normally distributed form.  
#step\_dummy: This step creates dummy variables for all nominal features to enable the use of these categorical features in machine learning models.  
#step\_nzv: This step removes predictors that have no or very low variance across the samples. This is done to remove features that are not informative or that might cause overfitting.  
  
stroke\_training$stroke <- as.factor(stroke\_training$stroke)  
stroke\_testing$stroke <- as.factor(stroke\_testing$stroke)  
  
Stroke\_main<-recipe(stroke~ age, hypertension, heart\_disease, occupation, residence ,metric\_1 ,metric\_2, metric\_3, metric\_4 ,metric\_5 ,smoking\_status, stroke\_training ,data=stroke\_training, family='binomial')%>%  
 step\_YeoJohnson(all\_numeric(), - all\_outcomes()) %>% ##  
 step\_dummy(all\_nominal(), -all\_outcomes()) %>%  
 step\_nzv(all\_predictors())

#Step-4 Model fitting

# Here we are preparing and baking the recipe and setting the model and creating workflow and fitting the model.  
Stroke\_main %>%   
 prep() %>%   
 bake(new\_data = stroke\_testing)

## # A tibble: 8,680 × 2  
## age stroke  
## <dbl> <fct>   
## 1 2.39 0   
## 2 20.3 0   
## 3 27.2 0   
## 4 27.9 0   
## 5 18.2 0   
## 6 28.8 0   
## 7 14.0 0   
## 8 25.0 0   
## 9 2.39 0   
## 10 8.42 0   
## # ℹ 8,670 more rows

lr\_mod<-logistic\_reg()%>%  
 set\_engine('glm')  
  
  
stroke\_workflow<-workflow()%>%  
 add\_model(lr\_mod)%>%  
 add\_recipe(Stroke\_main)  
  
stroke\_fit <- stroke\_workflow %>%   
 fit(data=stroke\_testing)

#Step-5 Testing model

# Here we comparing our predicted values with testing dataset.  
# Here we are testing our model with test data .  
# Here We train our model with test data set the 20% of actual data set.  
  
predict(stroke\_fit, stroke\_testing)%>%  
 bind\_cols(stroke\_testing%>%select(stroke))

## # A tibble: 8,680 × 2  
## .pred\_class stroke  
## <fct> <fct>   
## 1 0 0   
## 2 0 0   
## 3 0 0   
## 4 0 0   
## 5 0 0   
## 6 0 0   
## 7 0 0   
## 8 0 0   
## 9 0 0   
## 10 0 0   
## # ℹ 8,670 more rows

#Here we are predicting person who have stroke which is 1 and person who doesn't have stroke which is 0   
  
stroke\_predictions<-predict(stroke\_fit, stroke\_testing, type ="prob")%>%  
 bind\_cols(stroke\_testing%>%select(stroke))  
stroke\_predictions

## # A tibble: 8,680 × 3  
## .pred\_0 .pred\_1 stroke  
## <dbl> <dbl> <fct>   
## 1 1.00 0.000100 0   
## 2 0.994 0.00553 0   
## 3 0.968 0.0319 0   
## 4 0.963 0.0374 0   
## 5 0.997 0.00327 0   
## 6 0.953 0.0473 0   
## 7 0.999 0.00120 0   
## 8 0.982 0.0181 0   
## 9 1.00 0.000100 0   
## 10 1.00 0.000338 0   
## # ℹ 8,670 more rows

tidy(stroke\_fit)

## # A tibble: 2 × 5  
## term estimate std.error statistic p.value  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 (Intercept) -9.61 0.529 -18.2 9.95e-74  
## 2 age 0.153 0.0123 12.4 2.63e-35

# Confussion Matrix is poltted below:   
  
# As you can see there are no type 2 errors which are false negative and there are 156 false positive errors which means   
  
stroke\_predictions1 <-predict(stroke\_fit, stroke\_testing)%>%  
 bind\_cols(stroke\_testing%>%select(stroke))  
  
stroke\_predictions1%>%table()

## stroke  
## .pred\_class 0 1  
## 0 8524 156  
## 1 0 0

#Here we are checking the accuracy which is 98% accurate model  
  
#install.packages("caret")  
library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following objects are masked from 'package:yardstick':  
##   
## precision, recall, sensitivity, specificity

## The following object is masked from 'package:purrr':  
##   
## lift

predictions = predict(stroke\_fit, stroke\_testing)%>%  
 bind\_cols(stroke\_testing%>%select(stroke))  
predictions%>%conf\_mat(truth=stroke, .pred\_class)

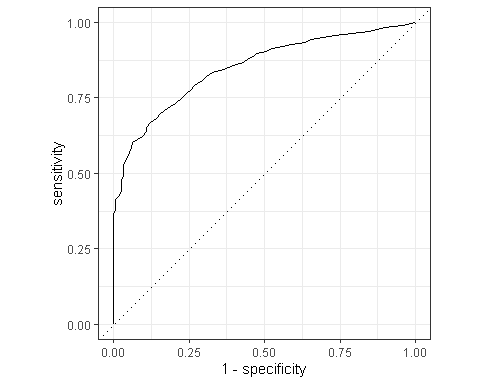
## Truth  
## Prediction 0 1  
## 0 8524 156  
## 1 0 0

caret::confusionMatrix(predictions$.pred\_class,predictions$stroke)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 8524 156  
## 1 0 0  
##   
**## Accuracy : 0.982**   
## 95% CI : (0.979, 0.9847)  
## No Information Rate : 0.982   
## P-Value [Acc > NIR] : 0.5213   
##   
## Kappa : 0   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 1.000   
## Specificity : 0.000   
## Pos Pred Value : 0.982   
## Neg Pred Value : NaN   
## Prevalence : 0.982   
## Detection Rate : 0.982   
## Detection Prevalence : 1.000   
## Balanced Accuracy : 0.500   
##   
## 'Positive' Class : 0   
##

#Step-6 Evaluating model   
  
# As much as area is there under the line the model is more accurate.  
  
stroke\_predictions %>%  
 roc\_curve(  
 truth = stroke,   
 .pred\_0,  
 event\_level = "first"  
 )%>%  
 autoplot()

## Warning: Returning more (or less) than 1 row per `summarise()` group was deprecated in  
## dplyr 1.1.0.  
## ℹ Please use `reframe()` instead.  
## ℹ When switching from `summarise()` to `reframe()`, remember that `reframe()`  
## always returns an ungrouped data frame and adjust accordingly.  
## ℹ The deprecated feature was likely used in the yardstick package.  
## Please report the issue at <https://github.com/tidymodels/yardstick/issues>.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.



#Plotting the area in number which is 85%.  
  
stroke\_predictions %>%  
 roc\_auc(  
 truth = stroke,   
 .pred\_0,  
 event\_level = "first")

## # A tibble: 1 × 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 roc\_auc binary 0.850

#Trying to improve the above model.

# Define the cross-validation scheme  
cv <- vfold\_cv(stroke\_training, v = 10, strata = stroke)  
  
# Define the model specification  
lr\_spec <- logistic\_reg() %>%  
 set\_engine("glm") %>%  
 set\_mode("classification")  
  
#create New Recipe  
  
Stroke\_main2<-recipe(stroke~ hypertension, heart\_disease,metric\_1 ,metric\_2, metric\_3, metric\_4 ,metric\_5 ,data=stroke\_training, family='binomial')%>%  
 step\_YeoJohnson(all\_numeric(), - all\_outcomes()) %>% ##  
 step\_dummy(all\_nominal(), -all\_outcomes()) %>%  
 step\_nzv(all\_predictors())  
  
  
# Define the workflow  
lr\_wf <- workflow() %>%  
 add\_model(lr\_spec) %>%  
 add\_recipe(Stroke\_main2)  
  
# Fit and evaluate the model using cross-validation  
  
stroke\_fit2 <- lr\_wf %>%  
 fit\_resamples(resamples = cv) %>%  
 collect\_metrics()  
  
stroke\_fit2

## # A tibble: 2 × 6  
## .metric .estimator mean n std\_err .config   
## <chr> <chr> <dbl> <int> <dbl> <chr>   
## 1 accuracy binary 0.982 10 0.000914 Preprocessor1\_Model1  
## 2 roc\_auc binary 0.578 10 0.00496 Preprocessor1\_Model1