Pima Indian women

Janica Wrosch, Bence Klujber

14 Dezember 2019

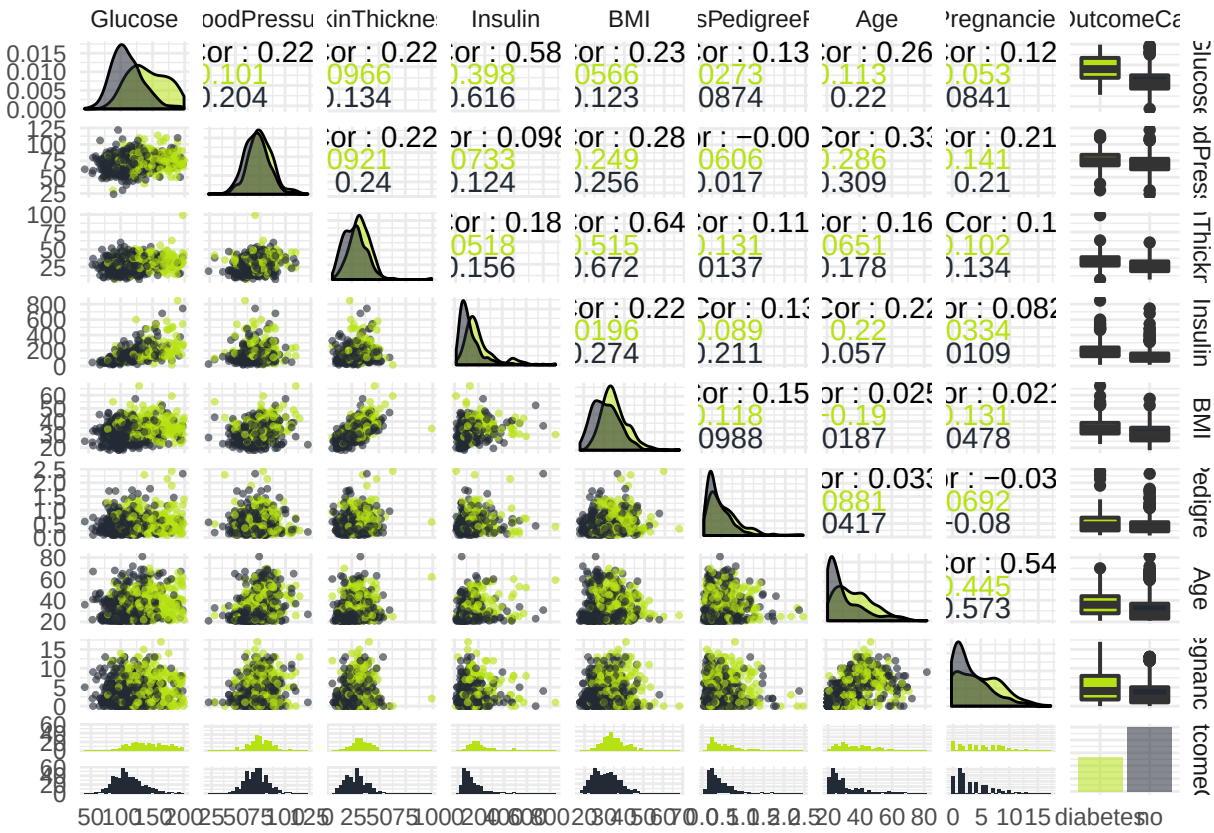
R Session Pima Indian Dataset

We were using statistical analysis for the Indian Dataset to find out if there are any correlations between the Outcome of being diabetical or not and our 9 health attributes measured from every women.

```
#Convert Outcome to categorical
data$OutcomeCat[data$Outcome == 0] <- 'no'
data$OutcomeCat[data$Outcome == 1] <- 'diabetes'
```

Graphical Overview over the Data

Following plots depict data distribution and shows scatterplots for each attribute compared to the others.

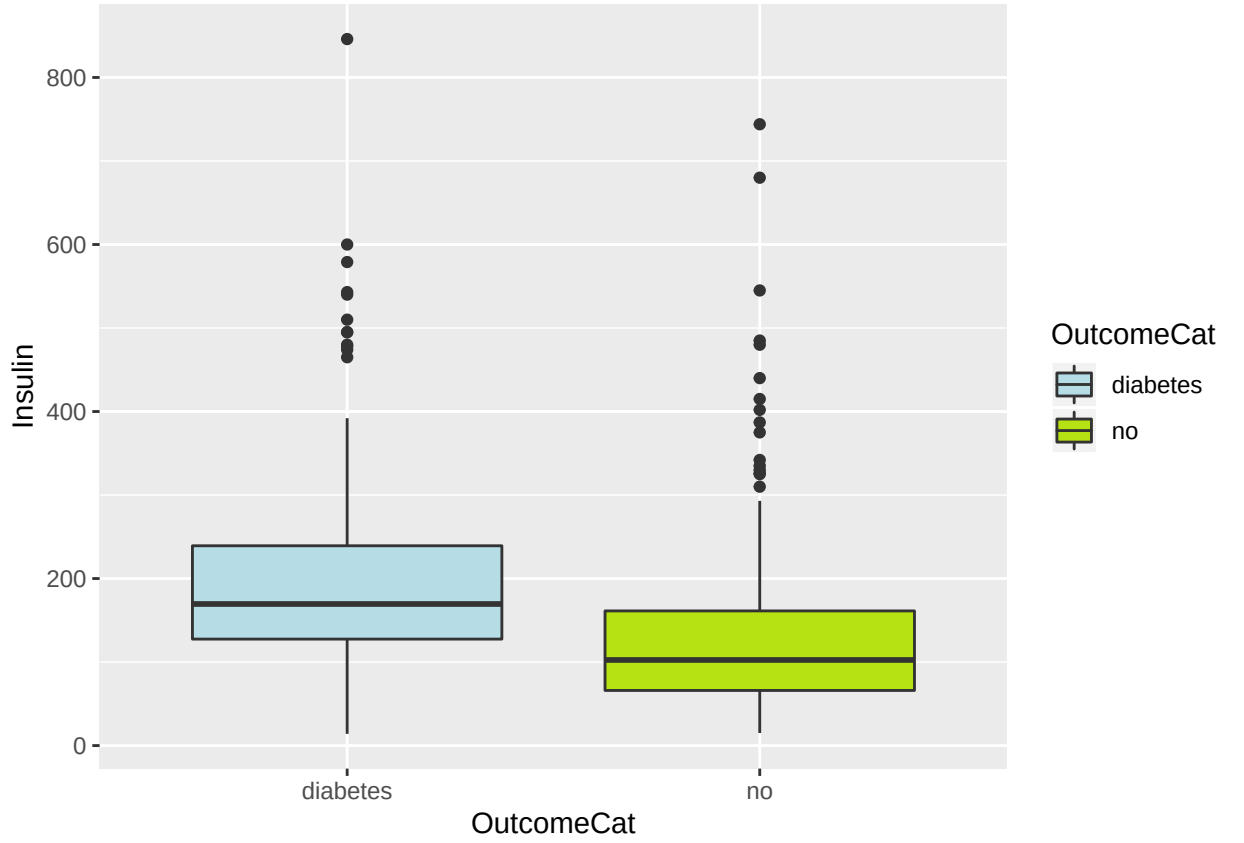


Interesting Plots

Following are the most interesting, more readable plots that give insights in the data.

```
#boxplots

ggplot(data, aes(x = OutcomeCat, y = Insulin))+
```

```
  geom_boxplot(aes(fill = OutcomeCat))+
  scale_fill_manual(values=c("no" = "#B6E213", "diabetes" = "#B6DCE6"))
```




```
diabetes_ppl <- subset(data, Outcome == 1)
healthy_ppl <- subset(data, Outcome == 0)

var.test(x = diabetes_ppl$Insulin, y = healthy_ppl$Insulin, alternative = "two.sided")
```

```
##
##  F test to compare two variances
##
## data:  diabetes_ppl$Insulin and healthy_ppl$Insulin
## F = 1.6767, num df = 129, denom df = 263, p-value = 0.0004706
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  1.252981 2.278564
## sample estimates:
## ratio of variances
##           1.676656
```

```
# shows variances are not equal
t.test(x = diabetes_ppl$Insulin, y = healthy_ppl$Insulin, alternative = "two.sided", var.equal = F)
```

```
##
##  Welch Two Sample t-test
##
## data:  diabetes_ppl$Insulin and healthy_ppl$Insulin
```

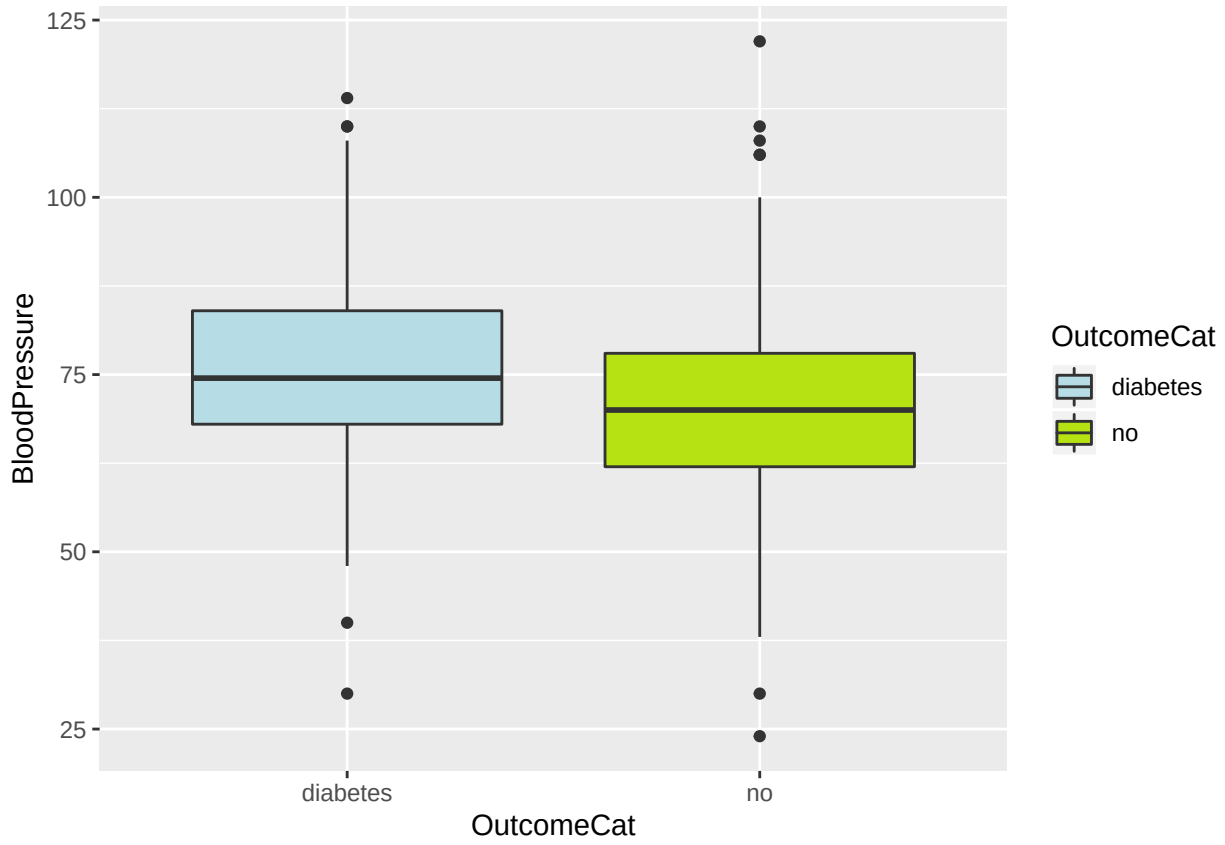
```
## t = 5.7833, df = 207.14, p-value = 2.672e-08
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   50.46025 102.65630
## sample estimates:
## mean of x mean of y
##  206.8462  130.2879
```

```
# accept H0 if p-value > 0.05 - to a 95 % confidence level we assume
# the mean Insulin level is different in the two groups!
t.test(x = diabetes_ppl$Insulin, y = healthy_ppl$Insulin, alternative = "greater", var.equal = F)
```

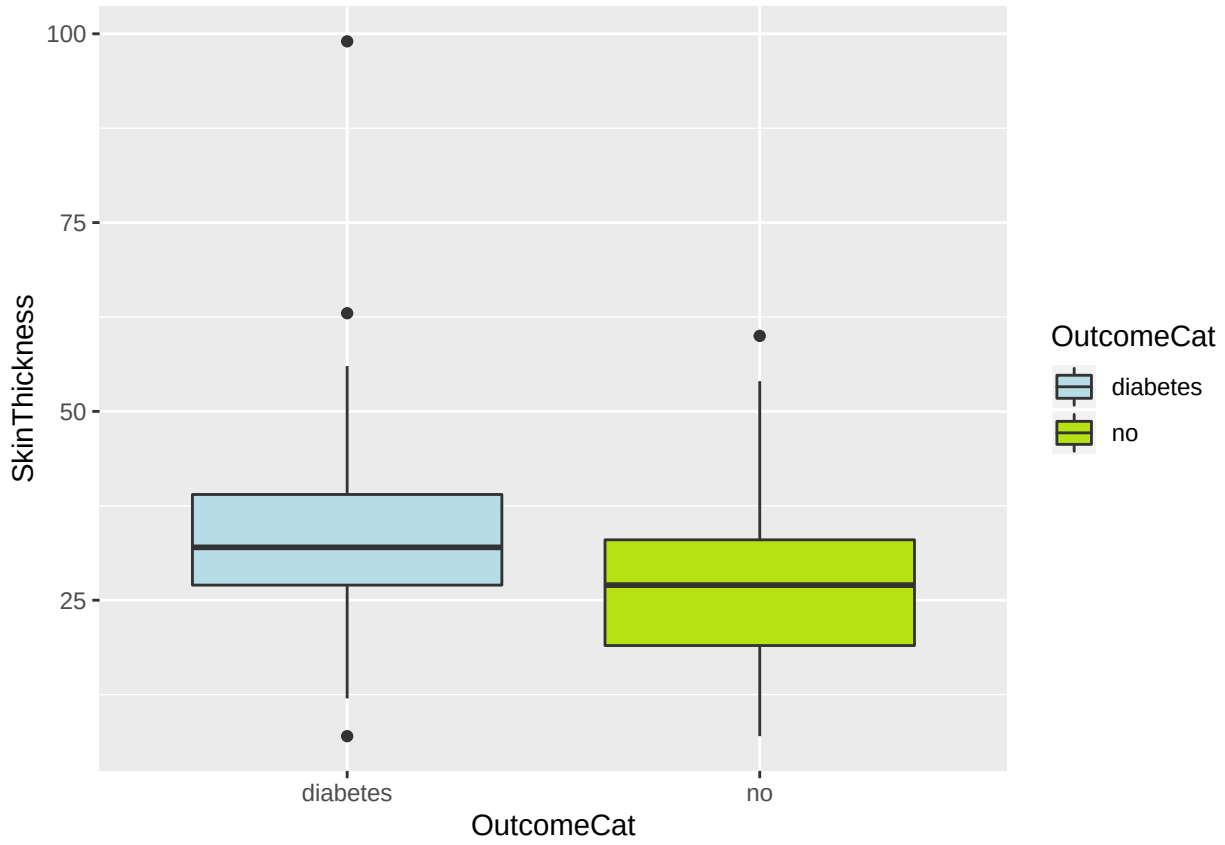
```
##
##  Welch Two Sample t-test
##
## data:  diabetes_ppl$Insulin and healthy_ppl$Insulin
## t = 5.7833, df = 207.14, p-value = 1.336e-08
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  54.68626      Inf
## sample estimates:
## mean of x mean of y
##  206.8462  130.2879
```

```
# accept H0 if p-value > 0.05 - to a 95 % confidence level we assume
# the mean Insulin level is bigger in the diabetes group!
```

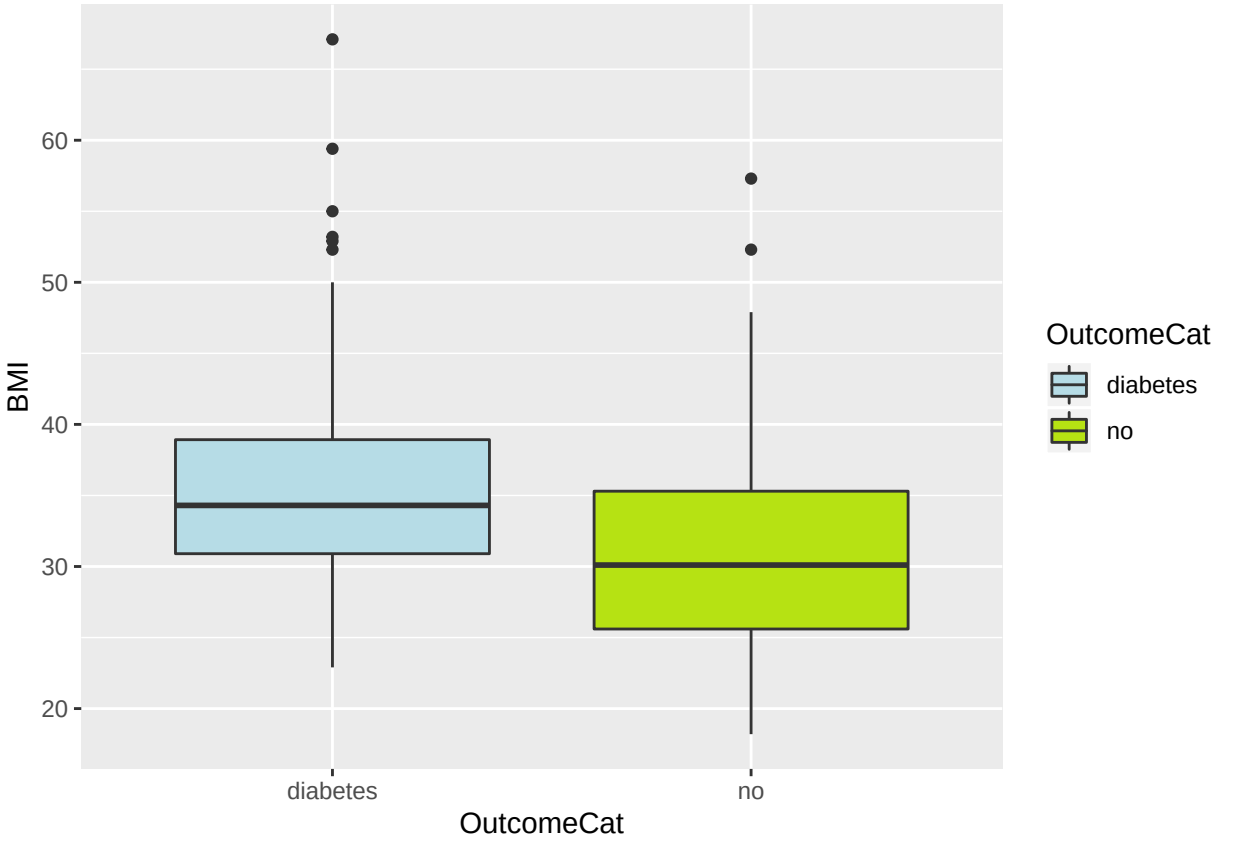
```
ggplot(data, aes(x = OutcomeCat, y = BloodPressure))+
  geom_boxplot(aes(fill = OutcomeCat))+
  scale_fill_manual(values=c("no" = "#B6E213", "diabetes" = "#B6DCE6"))
```



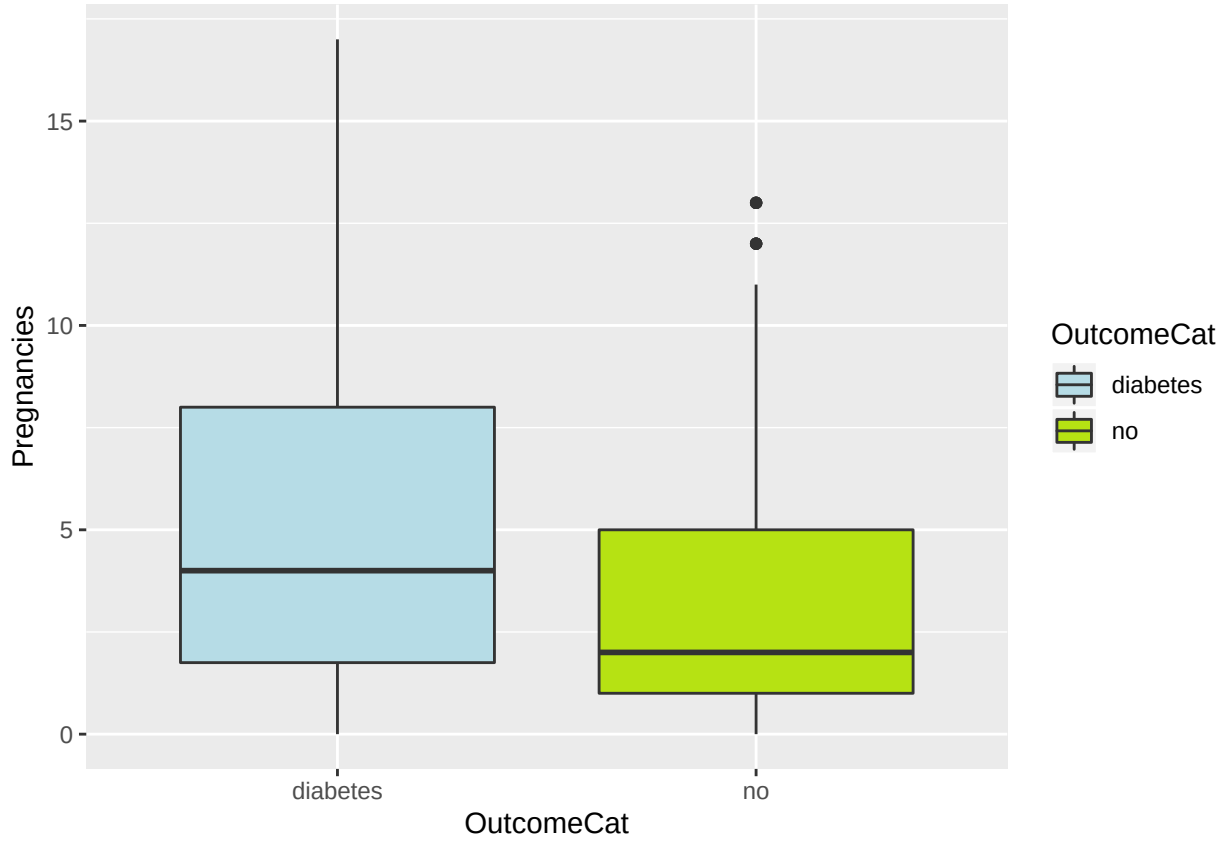

```
ggplot(data, aes(x = OutcomeCat, y = SkinThickness))+
  geom_boxplot(aes(fill = OutcomeCat))+
  scale_fill_manual(values=c("no" = "#B6E213", "diabetes" = "#B6DCE6"))
```



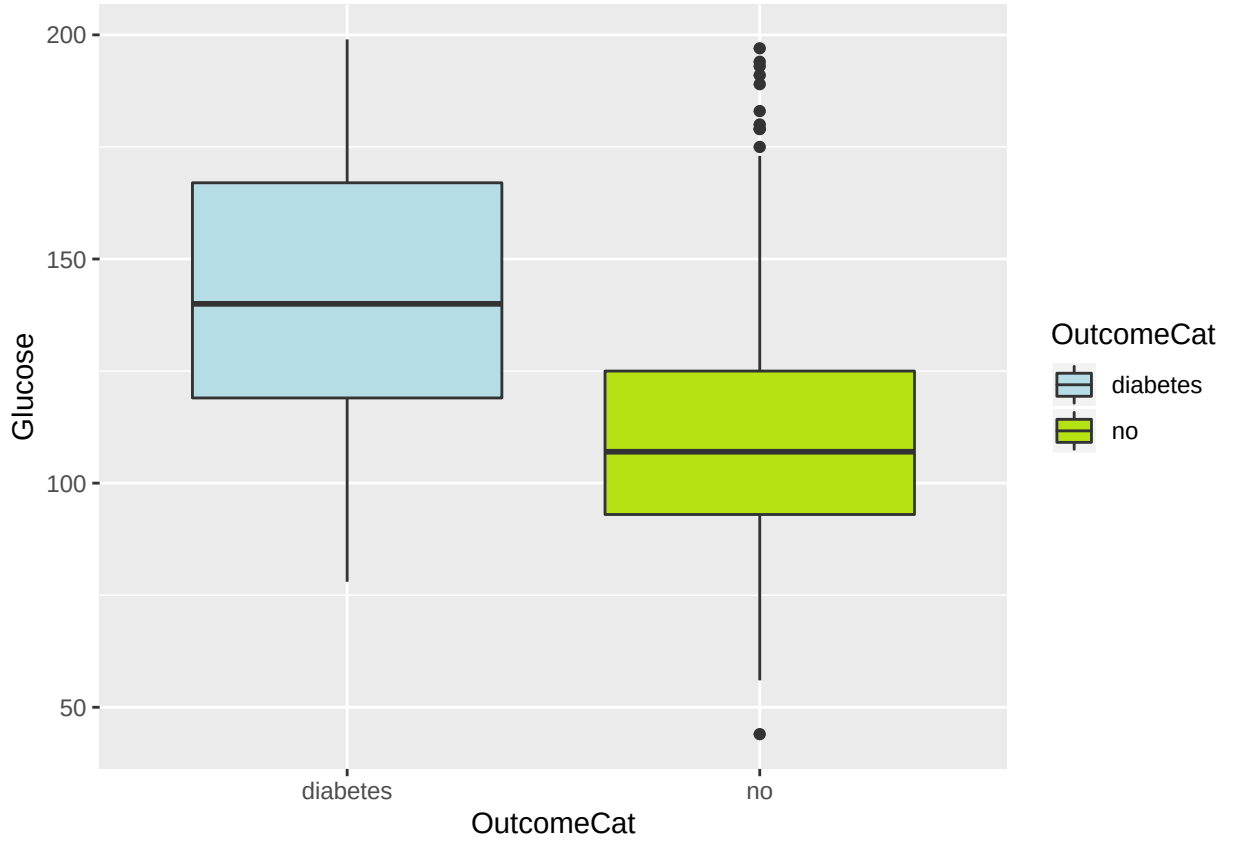

```
ggplot(data, aes(x = OutcomeCat, y = BMI))+
  geom_boxplot(aes(fill = OutcomeCat))+
  scale_fill_manual(values=c("no" = "#B6E213", "diabetes" = "#B6DCE6"))
```




```
ggplot(data, aes(x = OutcomeCat, y = Pregnancies))+
  geom_boxplot(aes(fill = OutcomeCat))+
  scale_fill_manual(values=c("no" = "#B6E213", "diabetes" = "#B6DCE6"))
```




```
ggplot(data, aes(x = OutcomeCat, y = Glucose))+
  geom_boxplot(aes(fill = OutcomeCat))+
  scale_fill_manual(values=c("no" = "#B6E213", "diabetes" = "#B6DCE6"))
```




```
var.test(x = diabetes_ppl$Glucose, y = healthy_ppl$Glucose, alternative = "two.sided")
```

```
##
##  F test to compare two variances
##
## data:  diabetes_ppl$Glucose and healthy_ppl$Glucose
## F = 1.4271, num df = 265, denom df = 496, p-value = 0.0007627
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  1.159308 1.768942
## sample estimates:
## ratio of variances
##           1.427137
```

```
# shows variances are not equal
t.test(x = diabetes_ppl$Glucose, y = healthy_ppl$Glucose, alternative = "two.sided", var.equal = F)
```

```
##
##  Welch Two Sample t-test
##
## data:  diabetes_ppl$Glucose and healthy_ppl$Glucose
## t = 14.884, df = 466.02, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  27.49380 35.85757
## sample estimates:
## mean of x mean of y
```

```
##  142.3195  110.6439
```

```
# accept H0 if p-value > 0.05 - to a 95 % confidence level we assume
# the mean glucose level is different in the two groups!
t.test(x = diabetes_ppl$Glucose, y = healthy_ppl$Glucose, alternative = "greater", var.equal = F)
```

```
##
##  Welch Two Sample t-test
##
## data:  diabetes_ppl$Glucose and healthy_ppl$Glucose
## t = 14.884, df = 466.02, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  28.16828      Inf
## sample estimates:
## mean of x mean of y
##  142.3195  110.6439
```

```
# accept H0 if p-value < 0.05 - to a 95 % confidence level we assume
# the mean glucose level is higher in the diabetes group!
```

Linear Modelling: Finding significant relations

For first statistical regression model we assume that all attributes can have a significant influence on the Outcome. So we start with all attributes in a linear model and then remove all insignificant attributes one after another

```
##
## Call:
## lm(formula = Outcome ~ Pregnancies + BMI + BloodPressure + SkinThickness +
##     Glucose + Insulin + DiabetesPedigreeFunction + Age, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.07966 -0.25711 -0.06177  0.25851  1.03750
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -1.103e+00  1.436e-01  -7.681 1.34e-13 ***
## Pregnancies                1.295e-02  8.364e-03   1.549  0.12230
## BMI                        9.325e-03  3.901e-03   2.391  0.01730 *
## BloodPressure              5.465e-05  1.730e-03   0.032  0.97482
## SkinThickness              1.678e-03  2.522e-03   0.665  0.50631
## Glucose                    6.409e-03  8.159e-04   7.855 4.07e-14 ***
## Insulin                   -1.233e-04  2.045e-04  -0.603  0.54681
## DiabetesPedigreeFunction   1.572e-01  5.804e-02   2.708  0.00707 **
## Age                        5.878e-03  2.787e-03   2.109  0.03559 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3853 on 383 degrees of freedom
##   (376 observations deleted due to missingness)
## Multiple R-squared:  0.3458, Adjusted R-squared:  0.3321
## F-statistic:  25.3 on 8 and 383 DF,  p-value: < 2.2e-16
```

```
##
```

```
## Call:
## lm(formula = Outcome ~ Pregnancies + BMI + SkinThickness + Glucose +
##     Insulin + DiabetesPedigreeFunction + Age, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.07942 -0.25706 -0.06165  0.25874  1.03655
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -1.1004388  0.1246954  -8.825  < 2e-16 ***
## Pregnancies               0.0129628  0.0083471   1.553   0.1213
## BMI                       0.0093542  0.0037850   2.471   0.0139 *
## SkinThickness             0.0016772  0.0025184   0.666   0.5058
## Glucose                   0.0064112  0.0008107   7.909  2.8e-14 ***
## Insulin                  -0.0001238  0.0002037  -0.608   0.5438
## DiabetesPedigreeFunction  0.1570167  0.0577001   2.721   0.0068 **
## Age                       0.0058936  0.0027398   2.151   0.0321 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3847 on 384 degrees of freedom
##   (376 observations deleted due to missingness)
## Multiple R-squared:  0.3458, Adjusted R-squared:  0.3338
## F-statistic: 28.99 on 7 and 384 DF,  p-value: < 2.2e-16
```

```
##
## Call:
## lm(formula = Outcome ~ Pregnancies + BMI + SkinThickness + Glucose +
##     DiabetesPedigreeFunction + Age, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.17912 -0.25139 -0.06825  0.26355  1.02461
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -1.0782689  0.1005565 -10.723  < 2e-16 ***
## Pregnancies               0.0207430  0.0065096   3.187 0.001525 **
## BMI                       0.0115668  0.0032128   3.600 0.000348 ***
## SkinThickness             0.0005571  0.0020959   0.266 0.790479
## Glucose                   0.0060014  0.0005796  10.354  < 2e-16 ***
## DiabetesPedigreeFunction  0.1823318  0.0492005   3.706 0.000233 ***
## Age                       0.0039119  0.0020897   1.872 0.061766 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3826 on 527 degrees of freedom
##   (234 observations deleted due to missingness)
## Multiple R-squared:  0.3498, Adjusted R-squared:  0.3424
## F-statistic: 47.26 on 6 and 527 DF,  p-value: < 2.2e-16
```

```
##
## Call:
## lm(formula = Outcome ~ Pregnancies + BMI + Glucose + DiabetesPedigreeFunction +
```

```
##     Age, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1415 -0.2918 -0.0750  0.3010  1.0085
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -1.0640595  0.0869491 -12.238  < 2e-16 ***
## Pregnancies               0.0194424  0.0051109   3.804 0.000154 ***
## BMI                       0.0135780  0.0021618   6.281 5.71e-10 ***
## Glucose                   0.0063476  0.0005053  12.562  < 2e-16 ***
## DiabetesPedigreeFunction  0.1387755  0.0445520   3.115 0.001910 **
## Age                       0.0017978  0.0015138   1.188 0.235360
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3956 on 746 degrees of freedom
##   (16 observations deleted due to missingness)
## Multiple R-squared:  0.3187, Adjusted R-squared:  0.3141
## F-statistic: 69.79 on 5 and 746 DF,  p-value: < 2.2e-16
```

The final model we ended up, is the following:

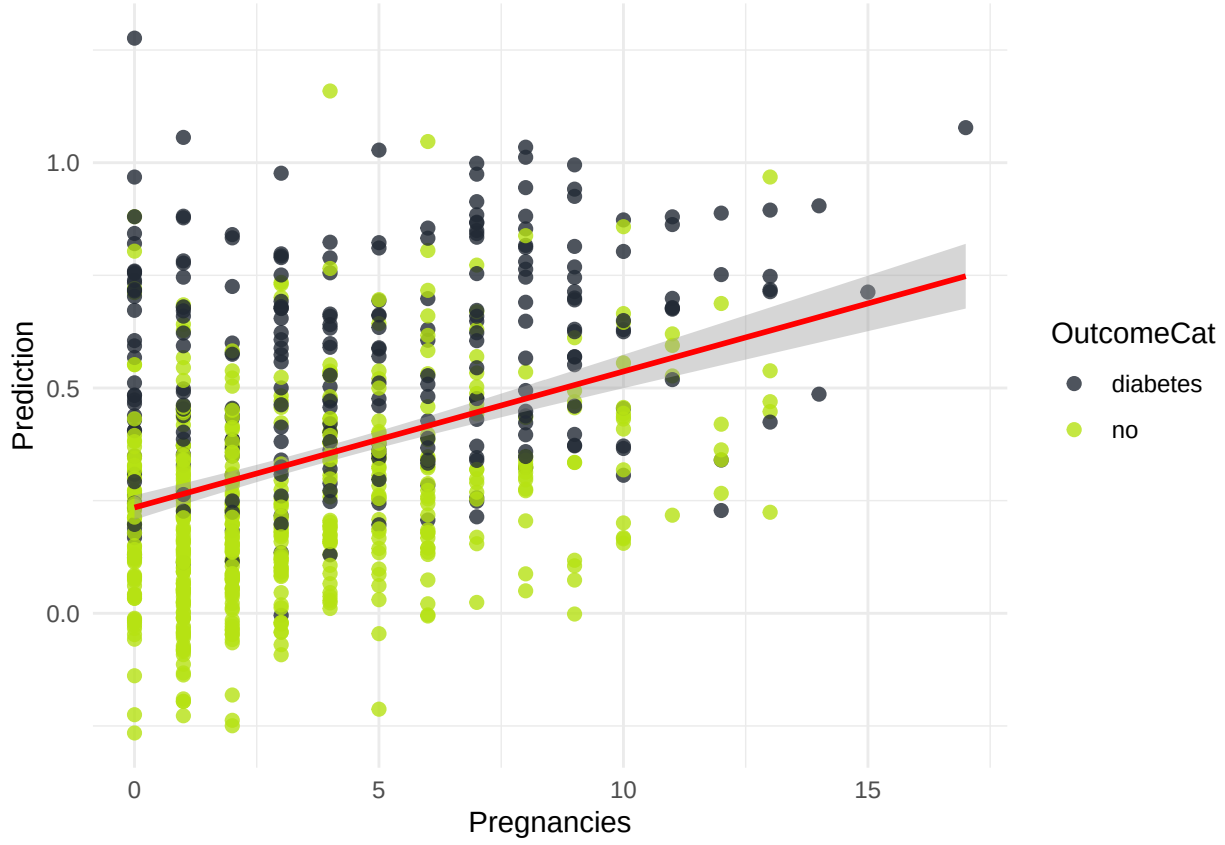
```
summary(model1)
```

```
##
## Call:
## lm(formula = (Outcome) ~ (Pregnancies) + (BMI) + (Glucose) +
##     (DiabetesPedigreeFunction), data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.15905 -0.28837 -0.07765  0.30048  1.00445
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -1.0305919  0.0822784 -12.526  < 2e-16 ***
## Pregnancies               0.0226942  0.0043167   5.257 1.91e-07 ***
## BMI                       0.0134618  0.0021601   6.232 7.70e-10 ***
## Glucose                   0.0064862  0.0004918  13.189  < 2e-16 ***
## DiabetesPedigreeFunction  0.1404196  0.0445427   3.152  0.00168 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3957 on 747 degrees of freedom
##   (16 observations deleted due to missingness)
## Multiple R-squared:  0.3174, Adjusted R-squared:  0.3137
## F-statistic: 86.83 on 4 and 747 DF,  p-value: < 2.2e-16
```

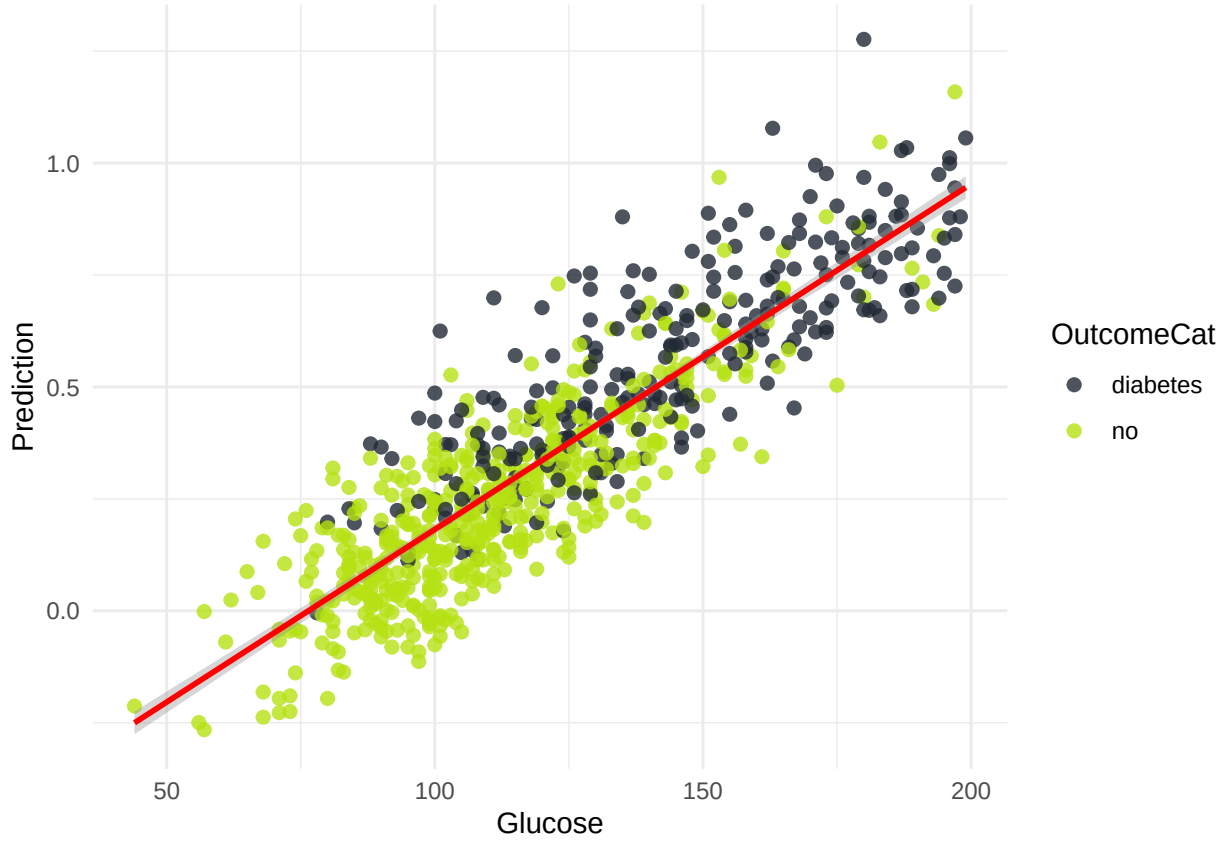
The model calculates a value for the outcome. Values below 0.5 show that the women is more likely to not have diabetes (Outcome = 0), according to her health values. While values above 0.5 mostly categorize her into diabetical. Nevertheless, this model was not generated for prediction but more to explain significant relations within the data. So we found out that *amount of prenancies*, *BMI*, the *glucose level* and the *diabetes pedigree function* have a major influence on the outcome.

Mainly one more pregnancy will increase the value of outcome by 0.02 (the outcome can have values mostly between 0 and 1). If the BMI increases by one point, the outcome will also increase by 0.013. When the glucose level increases by one point, the value of outcome will increase by 0.006. The pedigree function whose value describes the likeliness of having diabetes according to one's ancestors and their likeliness to have diabetes has also an influence on the outcome in our case. When it is increased by one unit, the outcome will increase by 0.14. This absolutely makes sense to us because Pima Indian women have to deal with diabetes since a long time. No one knows how it started for them that they often get diabetes but now that it is in their genes and gets passed along in every generation, the probability to get diabetes is high.

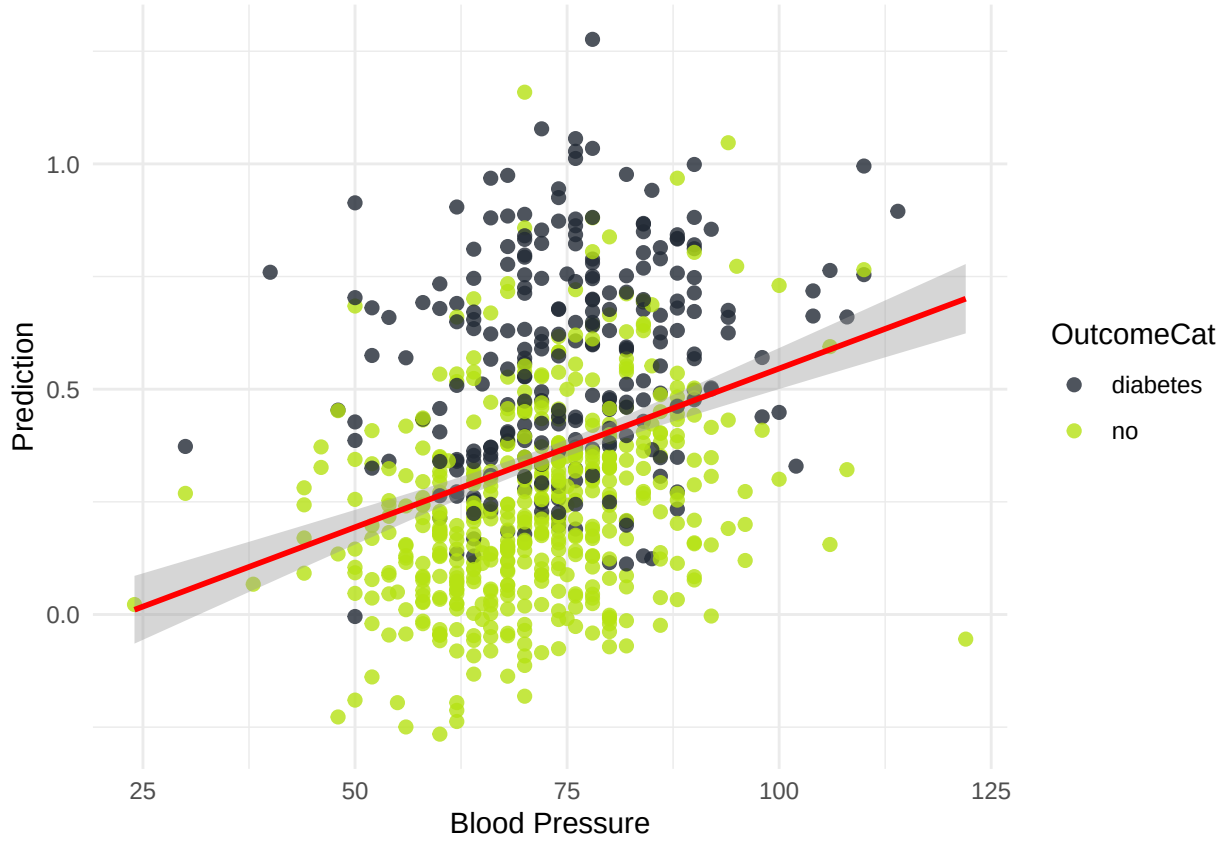
```
data$prediction <- predict(model1, newdata = data)
data$pred1 <- round(data$prediction,digits = 0)

ggplot(data, aes(x = Pregnancies, y = prediction))+
  geom_point(aes(color = OutcomeCat), size = 2, alpha = 0.8)+
  scale_color_manual(values = c("no"="#B6E213", "diabetes"="#222A35"))+
  geom_smooth(method = "lm", color = "red")+
  scale_x_continuous(name = "Pregnancies")+
  scale_y_continuous(name = "Prediction")+
  theme_minimal()
```



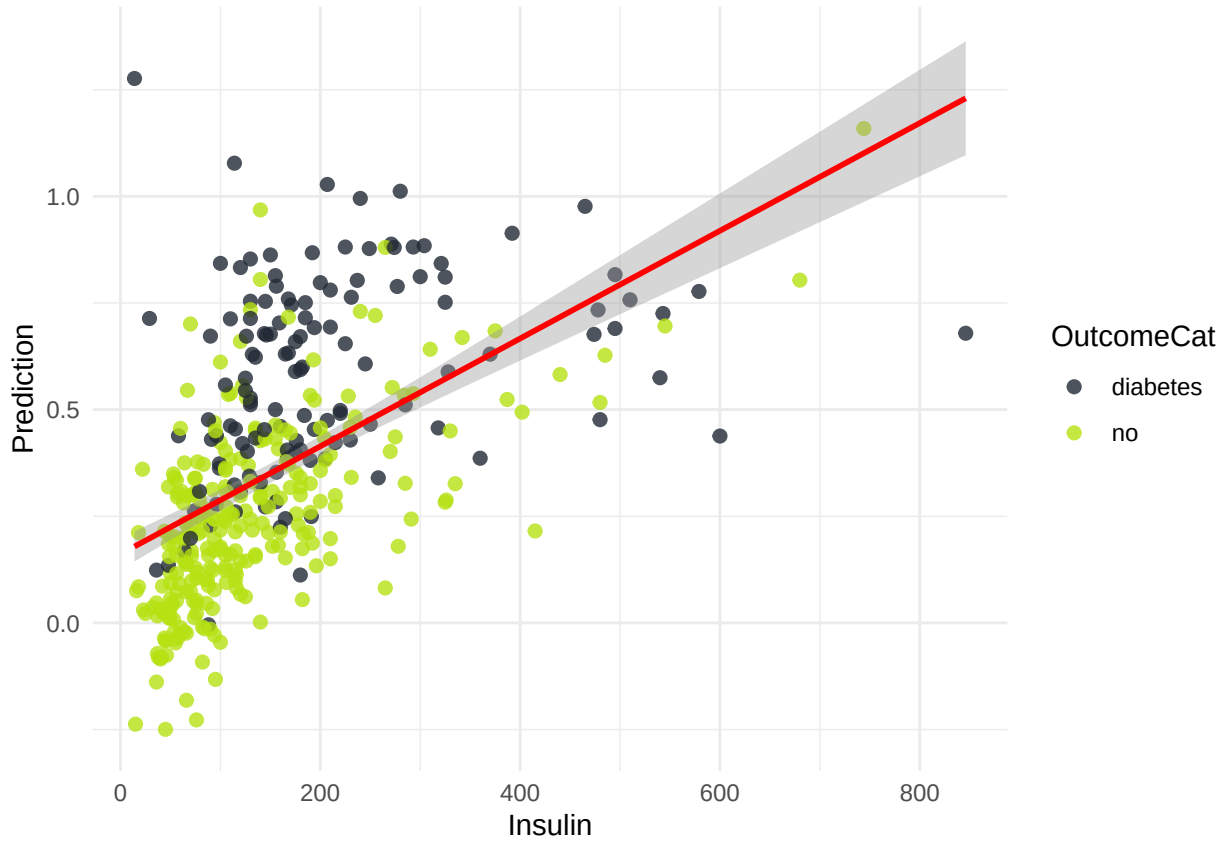

```
ggplot(data, aes(x = Glucose, y = prediction))+
  geom_point(aes(color = OutcomeCat), size = 2, alpha = 0.8)+
  scale_color_manual(values = c("no"="#B6E213", "diabetes"="#222A35"))+
  geom_smooth(method = "lm", color = "red")+
  scale_x_continuous(name = "Glucose")+
  scale_y_continuous(name = "Prediction")+
  theme_minimal()
```




```
ggplot(data, aes(x = BloodPressure, y = prediction))+
  geom_point(aes(color = OutcomeCat), size = 2, alpha = 0.8)+
  scale_color_manual(values = c("no"="#B6E213", "diabetes"="#222A35"))+
  geom_smooth(method = "lm", color = "red")+
  scale_x_continuous(name = "Blood Pressure")+
  scale_y_continuous(name = "Prediction")+
  theme_minimal()
```




```
ggplot(data, aes(x = Insulin, y = prediction))+
  geom_point(aes(color = OutcomeCat), size = 2, alpha = 0.8)+
  scale_color_manual(values = c("no"="#B6E213", "diabetes"="#222A35"))+
  geom_smooth(method = "lm", color = "red")+
  scale_x_continuous(name = "Insulin")+
  scale_y_continuous(name = "Prediction")+
  theme_minimal()
```




```
TP <- data %>%
  filter(Outcome == 1, pred1 == 1, !is.na(pred1))%>%
  summarize(summe = length(Outcome))

TN <- data %>%
  filter(Outcome == 0, pred1 == 0, !is.na(pred1))%>%
  summarize(summe = length(Outcome))

FP <- data%>%
  filter(Outcome == 0, pred1 == 1, !is.na(pred1))%>%
  summarize(summe = length(Outcome))

FN <- data%>%
  filter(Outcome == 1, pred1 == 0, !is.na(pred1))%>%
  summarize(summe = length(Outcome))

total <- TP+TN+FP+FN

acc <- (TP+TN)/total #accuracy
preci <- TP/(TP+FP) #precision
recall <- TP/(TP+FN) #recall

F_stat <- 2 / ((1/preci)+(1/recall))

"Accuracy"
```

```
## [1] "Accuracy"
```

```
acc
```

```
##       summe
## 1 0.7672872
```

```
"Precision"
```

```
## [1] "Precision"
```

```
preci
```

```
##      summe
## 1 0.721393
```

```
"Recall"
```

```
## [1] "Recall"
```

```
recall
```

```
##       summe
## 1 0.5492424
```

```
"F statistic"
```

```
## [1] "F statistic"
```

```
F_stat
```

```
##       summe
## 1 0.6236559
```