Data Simulation

Yaqi

May 16, 2016

Title: Glycated Albumin and Hemoglobin in Controlling Complications of Infants of Diabetic Mothers During Pregnancy

References:

1. Complications in Infants of Diabetic Mothers Related to Glycated Albumin and Hemoglobin Levels During Pregnancy. content source: http://www.ncbi.nlm.nih.gov/pubmed/27131880 (http://www.ncbi.nlm.nih.gov/pubmed/27131880)
2. Analysis of the method for conversion between levels of HbA1c and glycated albumin by linear regression analysis using a measurement error model. content source: http://www.ncbi.nlm.nih.gov/pubmed/19380169 (http://www.ncbi.nlm.nih.gov/pubmed/19380169)

Introduction:

Infants of diabetic mothers (IDMs) often have complications associated with maternal hyperglycemia. Glycated Albumin(GA) and Hemoglobin(HbA1c) are usully used as markers in diabetic patients. However, the HbA1c level is affected by an abnormal erythrocyte life span, which may occur in iron deficiency anemia. Pregnant women with diabetes mellitus or gestational diabetes mellitus (GDM) often develop iron deficiency anemia; therefore, HbA1c may be insufficient for assessing glycemic control in these women.

In this project, I simulated a dateset from the paper “Complications in Infants of Diabetic Mothers Related to Glycated Albumin and Hemoglobin Levels During Pregnancy”[1] which suggests to use GA as a marker to control diabeties during pregnancy in order to lower the incidence of compliations ammoun IDMs. In the end, I build a logistic model to predice the probabiliy of having complications given a GA value.

Part1 Data Simulation

1.Data set to simulate:

*Observations: In the research article, Daisuke Sugawara observed 42 infants of diabetics mothers(IDMs) 25 of which have at least 1 complication after birth. 17 of the IDMs has no complications. In order to reduce the bias of simulating a small dataset(less than 50), I decided to simulate a data set that contains 420 observations.

*Attributes There are 10 attributes for each observation in this dataset:

Complications/ Labs	Column name	Type	probability of having the condition
1.Hypoglycemia	Hypo_glc	Logic: T/F	13/42
2. Respiratory disorder	Resp_dis	Logic: T/F	11/42
3. Hypocalcemia	Hypo_cal	Logic: T/F	9/42
4. Hypervolemia	Hype_vol	Logic: T/F	4/42
5. Hyperbilirubinemia	Hype_bil	Logic: T/F	6/42
6. Myocardial hypertrophy	Hype_myocar	Logic: T/F	17/42
7. Large for date	Large_for_date	Logic: T/F	8/42
8. Glycated albumin	GA	Logic: T/F	NA
HbA1C	HbA1C	Logic: T/F	NA
Complications	complication	Logic: T/F	0.595

2.Creation of the data set and associated code:

- In the research paper, means and stander deviations(sd) of GA and HbA1C for patients with/without complications are given. However, the mean and sd of GA and HbA1C for all patients are not specified in the paper. So I first created vector ‘temp’ to hold to classes (Complications, NoComplications). Then use temp to help stimulating the distribution of GA for all patients.
- “Hb temp” is created to help to simulate a temporary Hb vectors for both patients with/without complications. This temporary Hb vector later is used to calculate the mean and sd for all the patients. However, a further research on correlations between HbA1C and GA indicates these 2 variables are strongly correlated (0.747)[2], a linear model using GA to predict HbA1C is HbA(1c)=1.73+0.245GA [2]. I used this model along with the mean and sd calculated from Hb_temp to simulate HbA1C for each patient.
- “Hypo_glc, Resp_dis, Hypo_cal, Hype_vol, Hype_bil, Hype_myocar, Large_for_date” are 7 common complications for IDMs. The paper[1] discovered that GA and HbA1C for mothers of IDMs who have Hype_vol, Hype_bil are not significantly different with those without these 2 conditions.
- “Hypo_glc, Resp_dis, Hypo_cal, Hype_myocar, Large_for_date” are simulated using logistic function with the probability specified(refer to attributes).
- “Hype_vol, Hype_bil” are simulated randomly with the probability specified(refer to attributes).

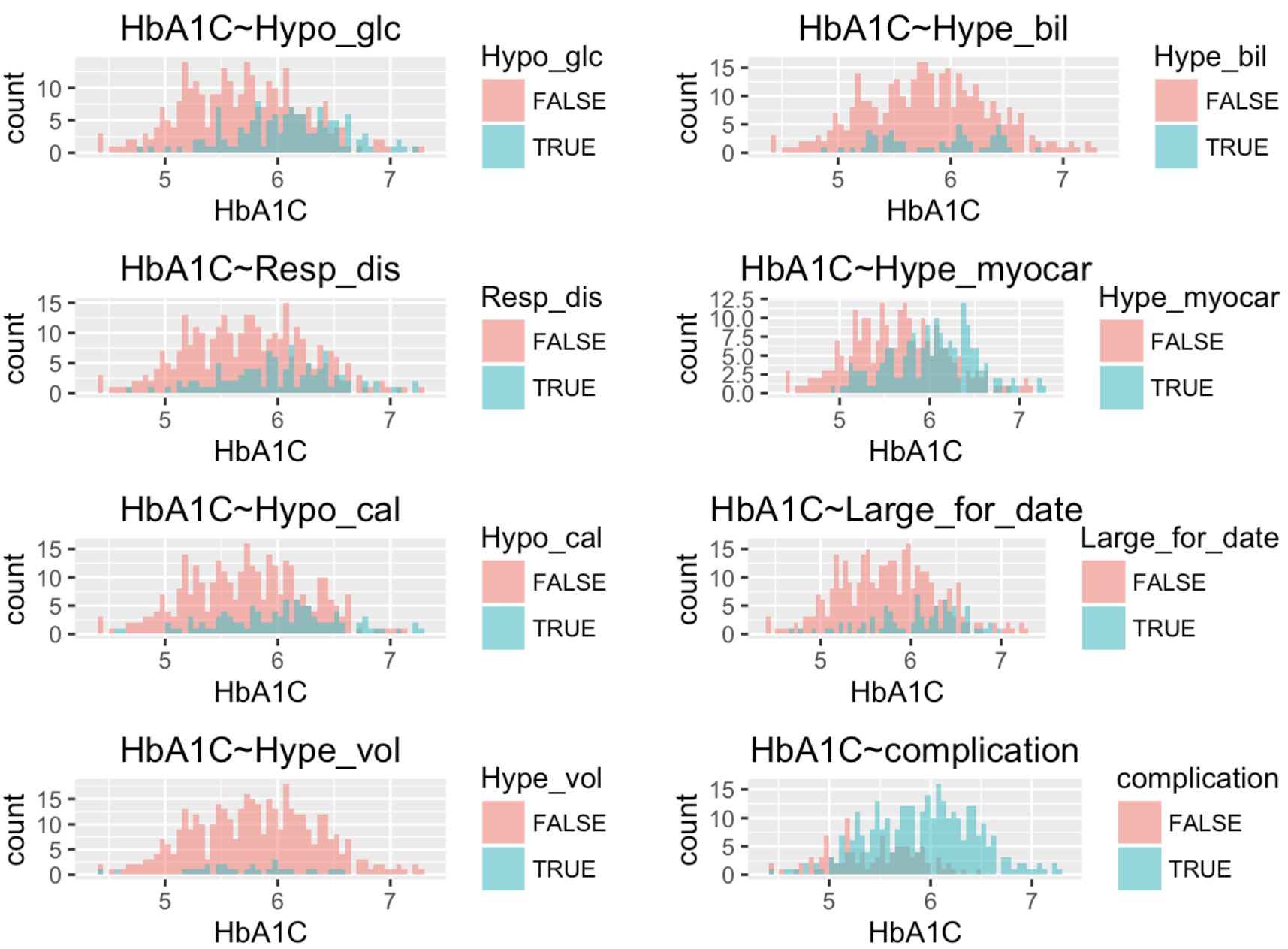
```
########################################## Part1 Data Simulation ####################
#########################
set.seed(123)
logistic <- function(t) 1 / (1 + exp(-t))

generate_dataset <- function(N){
  temp <- sample(c("Complication", "NoComplication"), N, replace=TRUE, prob=c(0.595,
0.415))
  GA_mean <- c(Complication= 15.3, NoComplication = 13.3)
  GA_sd <- c(Complication = 1.6, NoComplication = 1.1)
  GA <- rnorm(N, mean = GA_mean[temp], sd = GA_sd[temp])

  Hb_temp_mean <- c(Complication = 5.9, NoComplication = 5.6)
  Hb_temp_sd <- c(Complication = 0.3, NoComplication = 0.4)
  Hb_temp <- rnorm(N, mean = Hb_temp_mean[temp], sd = Hb_temp_sd[temp])
  HbA1C <- 0.245 * (GA-mean(GA)) + mean(Hb_temp) + rnorm(N, sd = sd(Hb_temp))

  Hypo_glc <- runif(N) < (13/42)*2*logistic(GA-mean(GA))
  Resp_dis <- runif(N) < (11/42)*2*logistic(GA-mean(GA))
  Hypo_cal <- runif(N) < (9/42)*2*logistic(GA-mean(GA))
  Hype_vol <- sample(c(TRUE, FALSE), N,replace=TRUE, prob=c(4/42, 1-4/42))
  Hype_bil <- sample(c(TRUE, FALSE), N,replace=TRUE, prob=c(6/42, 1-6/42))
  Hype_myocar <- runif(N) < (17/42)*2*logistic(GA-mean(GA))
  Large_for_date <- runif(N) < (8/42)*2*logistic(GA-mean(GA))
  df <- data.frame(Hypo_glc,Resp_dis,Hypo_cal,Hype_vol,Hype_bil,Hype_myocar,Large_for
_date)
  df$complication <- ifelse(apply(df[,c(1:7)],1,sum) != 0, TRUE, FALSE)
  df$GA <- GA
  df$HbA1C <- HbA1C
  return(df)
}
N <- 420
mydata <- generate_dataset(N)
```

Part2 Data Exploration

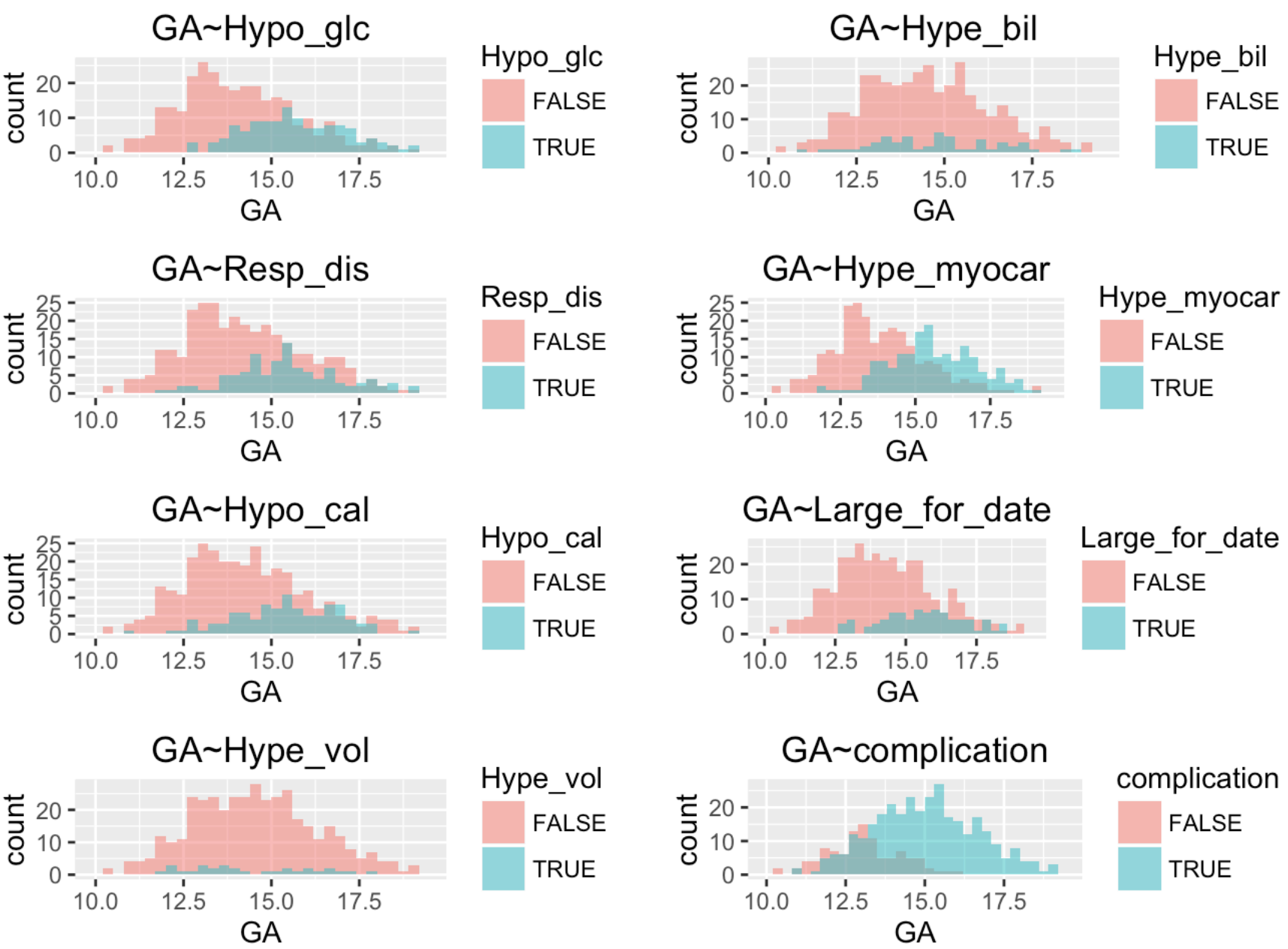
1.Distributions of GA and HbA1C for each complication

```
########################################## Part2 Data Exploration ####################
#########################
library(ggplot2)

# Multiple plot function

# ggplot objects can be passed in ..., or to plotlist (as a list of ggplot objects)
# - cols:   Number of columns in layout
# - layout: A matrix specifying the layout. If present, 'cols' is ignored.
# If the layout is something like matrix(c(1,2,3,3), nrow=2, byrow=TRUE),
# then plot 1 will go in the upper left, 2 will go in the upper right, and
```

```
# 3 will go all the way across the bottom.
multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {
  library(grid)
  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)
  numPlots = length(plots)
  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {
    # Make the panel
    # ncol: Number of columns of plots
    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                     ncol = cols, nrow = ceiling(numPlots/cols))
  }
  if (numPlots==1) {
    print(plots[[1]])
    } else {
    # Set up the page
    grid.newpage()
    pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))
    # Make each plot, in the correct location
    for (i in 1:numPlots) {
      # Get the i,j matrix positions of the regions that contain this subplot
      matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

      print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
                                      layout.pos.col = matchidx$col))
    }
  }
}


Hb1 <- ggplot(data = mydata) + geom_histogram(aes(x = HbA1C, fill = Hypo_glc), binwid
th = 0.05, alpha = 0.5, position = "identity" )+ggtitle('HbA1C~Hypo_glc')
Hb2 <- ggplot(data = mydata) + geom_histogram(aes(x = HbA1C, fill = Resp_dis), binwid
th = 0.05, alpha = 0.5, position = "identity")+ggtitle('HbA1C~Resp_dis')
Hb3 <- ggplot(data = mydata) + geom_histogram(aes(x = HbA1C, fill = Hypo_cal), binwid
th = 0.05, alpha = 0.5, position = "identity")+ggtitle('HbA1C~Hypo_cal')
Hb4 <- ggplot(data = mydata) + geom_histogram(aes(x = HbA1C, fill = Hype_vol), binwid
th = 0.05, alpha = 0.5, position = "identity")+ggtitle('HbA1C~Hype_vol')
Hb5 <- ggplot(data = mydata) + geom_histogram(aes(x = HbA1C, fill = Hype_bil), binwid
th = 0.05, alpha = 0.5, position = "identity")+ggtitle('HbA1C~Hype_bil')
Hb6 <- ggplot(data = mydata) + geom_histogram(aes(x = HbA1C, fill = Hype_myocar), bin
width = 0.05, alpha = 0.5, position = "identity")+ggtitle('HbA1C~Hype_myocar')
Hb7 <- ggplot(data = mydata) + geom_histogram(aes(x = HbA1C, fill = Large_for_date),
binwidth = 0.05, alpha = 0.5, position = "identity")+ggtitle('HbA1C~Large_for_date')
Hb8 <- ggplot(data = mydata) + geom_histogram(aes(x = HbA1C, fill = complication), bi
nwidth = 0.05, alpha = 0.5, position = "identity")+ggtitle('HbA1C~complication')
multiplot(Hb1,Hb2,Hb3,Hb4,Hb5,Hb6,Hb7,Hb8, cols = 2)
```




```
GA1 <- ggplot(data = mydata) + geom_histogram(aes(x = GA, fill = Hypo_glc), binwidth
= 0.3, alpha = 0.5, position = "identity" )+ggtitle('GA~Hypo_glc')
GA2 <- ggplot(data = mydata) + geom_histogram(aes(x = GA, fill = Resp_dis), binwidth
= 0.3, alpha = 0.5, position = "identity")+ggtitle('GA~Resp_dis')
GA3 <- ggplot(data = mydata) + geom_histogram(aes(x = GA, fill = Hypo_cal), binwidth
= 0.3, alpha = 0.5, position = "identity")+ggtitle('GA~Hypo_cal')
GA4 <- ggplot(data = mydata) + geom_histogram(aes(x = GA, fill = Hype_vol), binwidth
= 0.3, alpha = 0.5, position = "identity")+ggtitle('GA~Hype_vol')
GA5 <- ggplot(data = mydata) + geom_histogram(aes(x = GA, fill = Hype_bil), binwidth
= 0.3, alpha = 0.5, position = "identity")+ggtitle('GA~Hype_bil')
GA6 <- ggplot(data = mydata) + geom_histogram(aes(x = GA, fill = Hype_myocar), binwid
th = 0.3, alpha = 0.5, position = "identity")+ggtitle('GA~Hype_myocar')
GA7 <- ggplot(data = mydata) + geom_histogram(aes(x = GA, fill = Large_for_date), bin
width = 0.3, alpha = 0.5, position = "identity")+ggtitle('GA~Large_for_date')
GA8 <- ggplot(data = mydata) + geom_histogram(aes(x = GA, fill = complication), binwi
dth = 0.3, alpha = 0.5, position = "identity")+ggtitle('GA~complication')
multiplot(GA1,GA2,GA3,GA4,GA5,GA6,GA7,GA8, cols = 2)
```



- Each plot delivers the stories of the distributions of GA/HbA1C between patients with/without a particular complication.
- There are no significant differences of mean and sd of GA/HbA1C for infants with/without Hype_bil and Hype_vol.

2.2-sample T test between case and control for each complication

```
T_test <- function(data){
  for (col in names(data)[c(1:8)]){
    print(sprintf("======================================%s=========================
============",col))
    print(sprintf("======================================GA~%s",col))
    print (t.test(data$GA~data[[col]]))
    print(sprintf("====================================HbA1C~%s",col))
    print (t.test(data$HbA1C~data[[col]]))
  }
}

T_test(mydata)
```

```
## [1] "======================================Hypo_glc=============================
======="
```

```
## [1] "=====================================GA~Hypo_glc"
##
##  Welch Two Sample t-test
##
## data:  data$GA by data[[col]]
## t = -10.538, df = 269.88, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.981569 -1.357686
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            14.04067             15.71030
##
## [1] "==================================HbA1C~Hypo_glc"
##
##  Welch Two Sample t-test
##
## data:  data$HbA1C by data[[col]]
## t = -7.6515, df = 263.03, p-value = 3.767e-13
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5009189 -0.2958736
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            5.682429             6.080826
##
## [1] "=====================================Resp_dis=============================
======="
## [1] "=====================================GA~Resp_dis"
##
##  Welch Two Sample t-test
##
## data:  data$GA by data[[col]]
## t = -6.9277, df = 197.75, p-value = 5.876e-11
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.5864138 -0.8833704
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            14.22990             15.46479
##
## [1] "==================================HbA1C~Resp_dis"
##
##  Welch Two Sample t-test
##
## data:  data$HbA1C by data[[col]]
## t = -4.8378, df = 192.74, p-value = 2.681e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.3912847 -0.1646387
```

```
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            5.731796            6.009758
##
## [1] "=====================================Hypo_cal=============================
======="
## [1] "=====================================GA~Hypo_cal"
##
##  Welch Two Sample t-test
##
## data:  data$GA by data[[col]]
## t = -6.0848, df = 190.03, p-value = 6.302e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.4390518 -0.7344548
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            14.28281            15.36956
##
## [1] "==================================HbA1C~Hypo_cal"
##
##  Welch Two Sample t-test
##
## data:  data$HbA1C by data[[col]]
## t = -4.4017, df = 160.18, p-value = 1.954e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.3959575 -0.1506940
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            5.736871            6.010196
##
## [1] "=====================================Hype_vol=============================
======="
## [1] "=====================================GA~Hype_vol"
##
##  Welch Two Sample t-test
##
## data:  data$GA by data[[col]]
## t = 0.36192, df = 33.582, p-value = 0.7197
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5559425  0.7967276
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            14.55016            14.42977
##
## [1] "==================================HbA1C~Hype_vol"
##
##  Welch Two Sample t-test
```

```
##
## data:  data$HbA1C by data[[col]]
## t = 0.99375, df = 34.333, p-value = 0.3273
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1005431  0.2931010
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            5.808825            5.712546
##
## [1] "=====================================Hype_bil=============================
======="
## [1] "=====================================GA~Hype_bil"
##
##  Welch Two Sample t-test
##
## data:  data$GA by data[[col]]
## t = -1.0004, df = 71.077, p-value = 0.3205
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.7599918  0.2521702
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            14.50831            14.76222
##
## [1] "=====================================HbA1C~Hype_bil"
##
##  Welch Two Sample t-test
##
## data:  data$HbA1C by data[[col]]
## t = -1.0598, df = 77.646, p-value = 0.2925
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.21467557  0.06552705
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            5.792182            5.866757
##
## [1] "=====================================Hype_myocar==========================
==========="
## [1] "=====================================GA~Hype_myocar"
##
##  Welch Two Sample t-test
##
## data:  data$GA by data[[col]]
## t = -11.903, df = 386.28, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.059970 -1.475912
## sample estimates:
```

```
## mean in group FALSE  mean in group TRUE
##            13.81334             15.58128
##
## [1] "====================================HbA1C~Hype_myocar"
##
##  Welch Two Sample t-test
##
## data:  data$HbA1C by data[[col]]
## t = -7.7701, df = 394.54, p-value = 6.859e-14
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.4816397 -0.2871266
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            5.643619             6.028002
##
## [1] "=======================================Large_for_date=========================
============"
## [1] "=======================================GA~Large_for_date"
##
##  Welch Two Sample t-test
##
## data:  data$GA by data[[col]]
## t = -7.0505, df = 132.09, p-value = 8.958e-11
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.7094919 -0.9604189
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            14.29046             15.62542
##
## [1] "====================================HbA1C~Large_for_date"
##
##  Welch Two Sample t-test
##
## data:  data$HbA1C by data[[col]]
## t = -4.9143, df = 120.88, p-value = 2.83e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.4425054 -0.1883570
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            5.742617             6.058048
##
## [1] "=======================================complication==========================
==========="
## [1] "=======================================GA~complication"
##
##  Welch Two Sample t-test
##
```

```
## data:  data$GA by data[[col]]
## t = -13.396, df = 246.85, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.248112 -1.671758
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            13.06694            15.02688
##
## [1] "==================================HbA1C~complication"
##
##  Welch Two Sample t-test
##
## data:  data$HbA1C by data[[col]]
## t = -9.337, df = 205.58, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5885134 -0.3833073
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            5.436358            5.922269
```

- The results of 2 sample t-test justified what we saw from the plots above that Hype_vol and Hype_bil are not significantly different between control and case groups.

3.The linear relation between GA and HbA1C

```
cor(mydata[,c(9,10)])
```

```
##               GA     HbA1C
## GA    1.0000000 0.7577657
## HbA1C 0.7577657 1.0000000
```

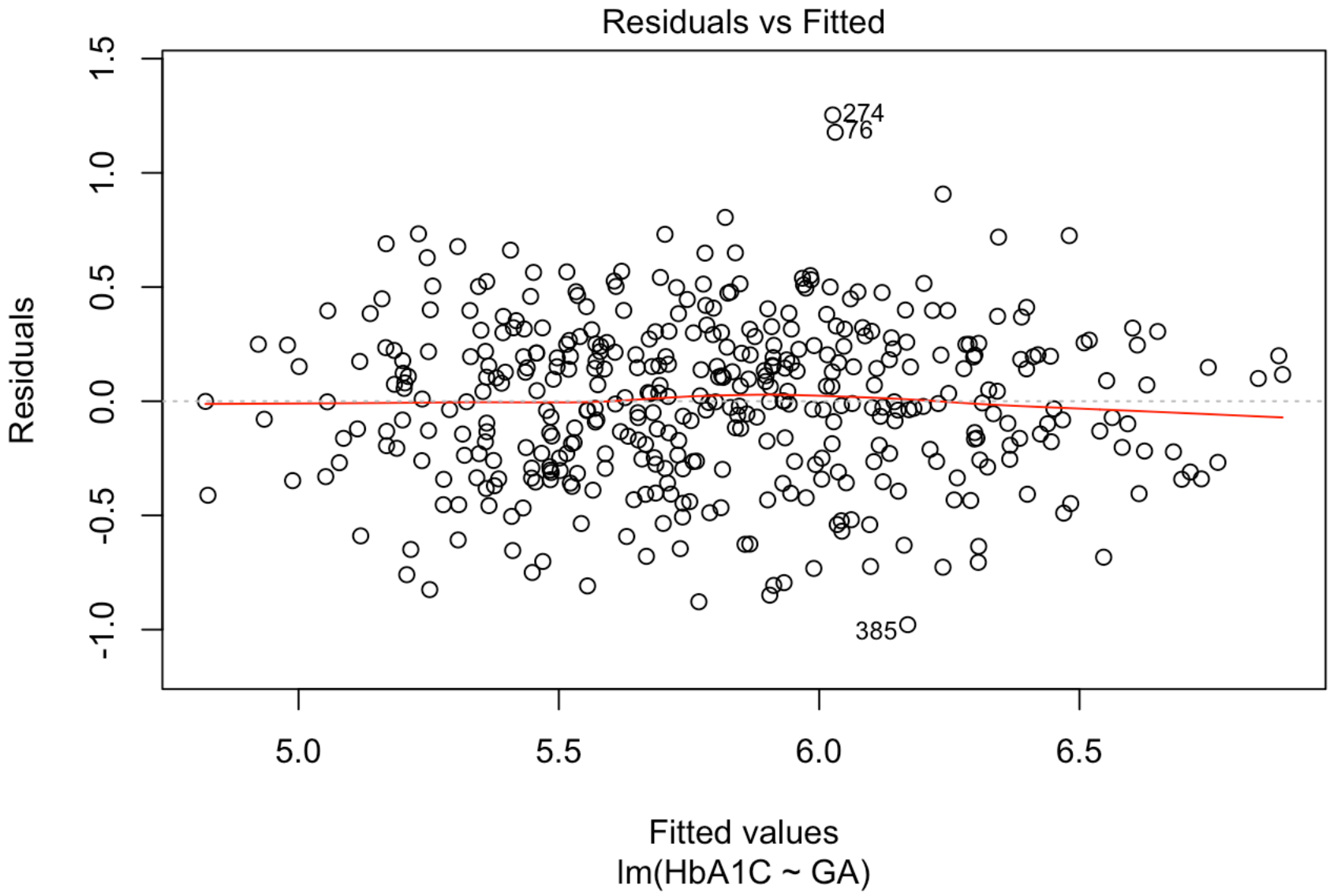
- GA and HbA1C are positively correlated, with a correlation of 0.758 very close to 0.747[2]

```
fit_Hb <- lm(HbA1C~GA,data = mydata)
pred_HB <- predict(fit_Hb)
summary(fit_Hb)
```

```
##
## Call:
## lm(formula = HbA1C ~ GA, data = mydata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.97841 -0.25337  0.00078  0.24092  1.25351
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.381311   0.145108   16.41   <2e-16 ***
## GA          0.235232   0.009908   23.74   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3545 on 418 degrees of freedom
## Multiple R-squared:  0.5742, Adjusted R-squared:  0.5732
## F-statistic: 563.7 on 1 and 418 DF,  p-value: < 2.2e-16
```

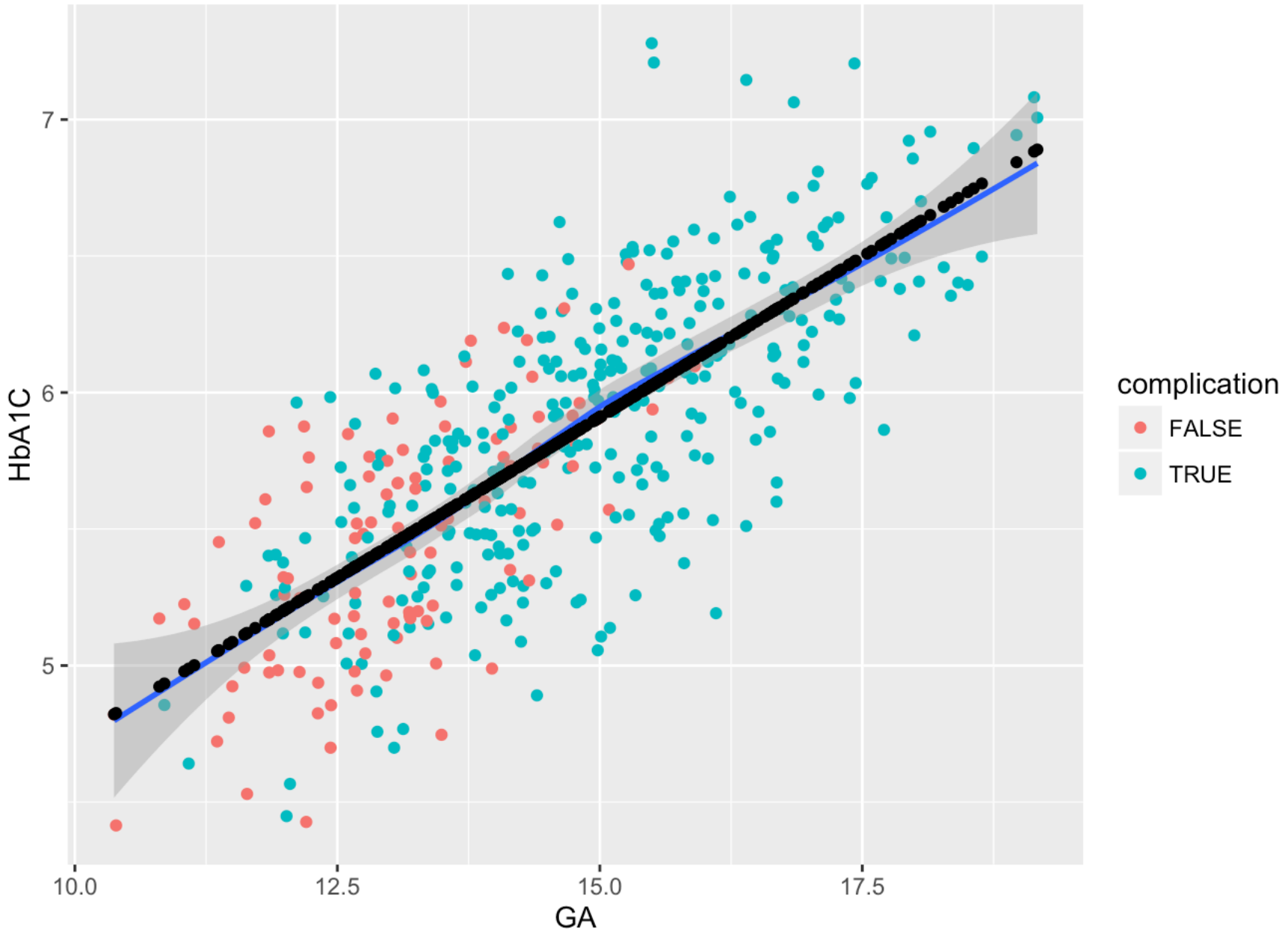
- Hb = 0.235*GA + 2.38 vary close to the linear model HbA(1c)=1.73+0.245GA [2]
- R^2 = 0.5742, P-value < 2.2e-16

```
plot(fit_Hb,1)
```



- The residuals are randomly distributed

```
ggplot(data=mydata)+geom_point(aes(x=GA,y=HbA1C,color = complication))+ geom_smooth(a
es(x=GA,y=HbA1C))+geom_point(aes(x=GA,y=pred_HB))
```

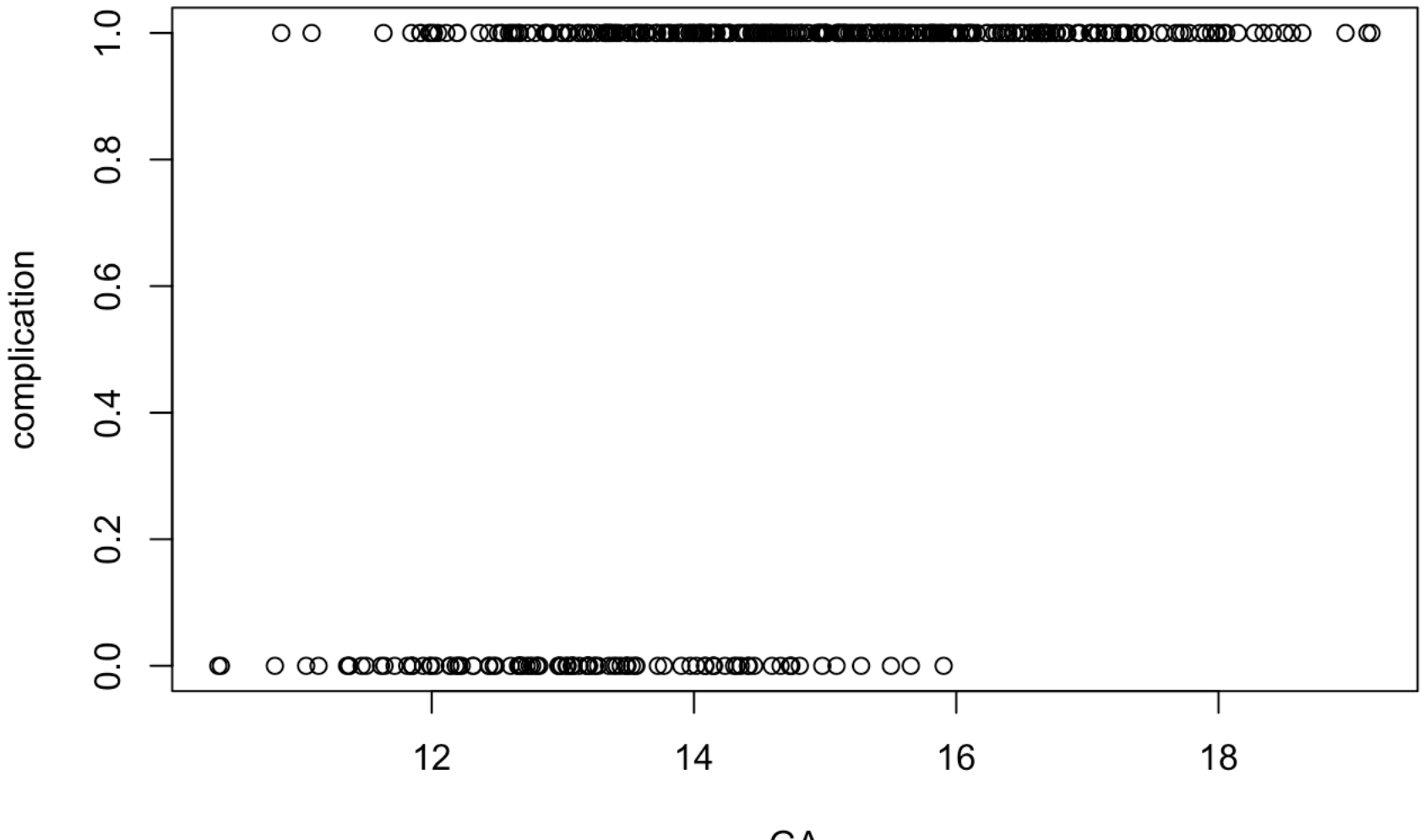


- this plot indicates the model is significant, although R^2 is 0.5742, this may due to the data is pretty spread out

Part3 Data modeling

1.Let’s plot complications vs GA again

```
mydata1 <- data.frame(apply(mydata,2,as.numeric))   # change the T/F to 1/0
plot(complication~GA, data = mydata1)
```



- The plot indicates babies with complications their moms have higher GA.
- Babies without complications their moms have lower GA.

2.Let's fit a logistic regression model

```
logisticPseudoR2s <- function(LogModel) {
  dev <- LogModel$deviance
  nullDev <- LogModel$null.deviance
  modelN <-  length(LogModel$fitted.values)
  R.l <-  1 -  dev / nullDev
  R.cs <- 1- exp ( -(nullDev - dev) / modelN)
  R.n <- R.cs / ( 1 - ( exp (-(nullDev / modelN))))
  cat("Pseudo R^2 for logistic regression\n")
  cat("Hosmer and Lemeshow R^2  ", round(R.l, 3), "\n")
  cat("Cox and Snell R^2        ", round(R.cs, 3), "\n")
  cat("Nagelkerke R^2           ", round(R.n, 3),    "\n")
}
fit_Com <- glm(complication~GA,family=binomial, data = mydata1)
pred_com <- predict(fit_Com,type="response")
summary(fit_Com)
```

```
##
## Call:
## glm(formula = complication ~ GA, family = binomial, data = mydata1)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.44778   0.07026   0.35904   0.67551   1.98682
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -12.069      1.499  -8.051 8.22e-16 ***
## GA             0.944      0.110   8.584  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 470.15  on 419  degrees of freedom
## Residual deviance: 350.86  on 418  degrees of freedom
## AIC: 354.86
##
## Number of Fisher Scoring iterations: 5
```

```
logisticPseudoR2s(fit_Com)
```

```
## Pseudo R^2 for logistic regression
## Hosmer and Lemeshow R^2   0.254
## Cox and Snell R^2         0.247
## Nagelkerke R^2            0.367
```

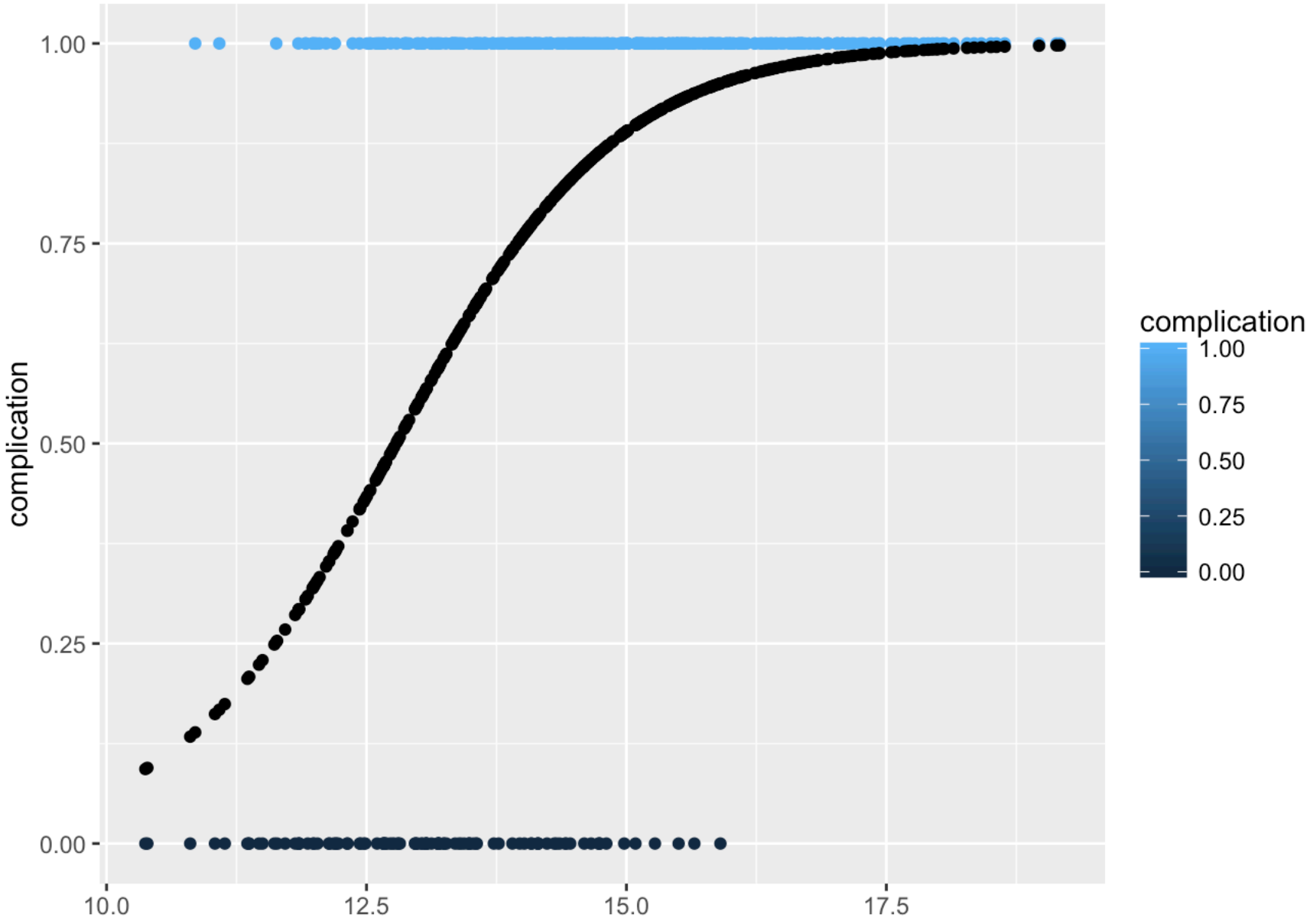
- Log(Have_Complications/no_Complications) = 0.944*GA - 12.069
- pseudoR^2:

1. Hosmer and Lemeshow R^2 0.254
2. Cox and Snell R^2 0.247
3. Nagelkerke R^2 0.367

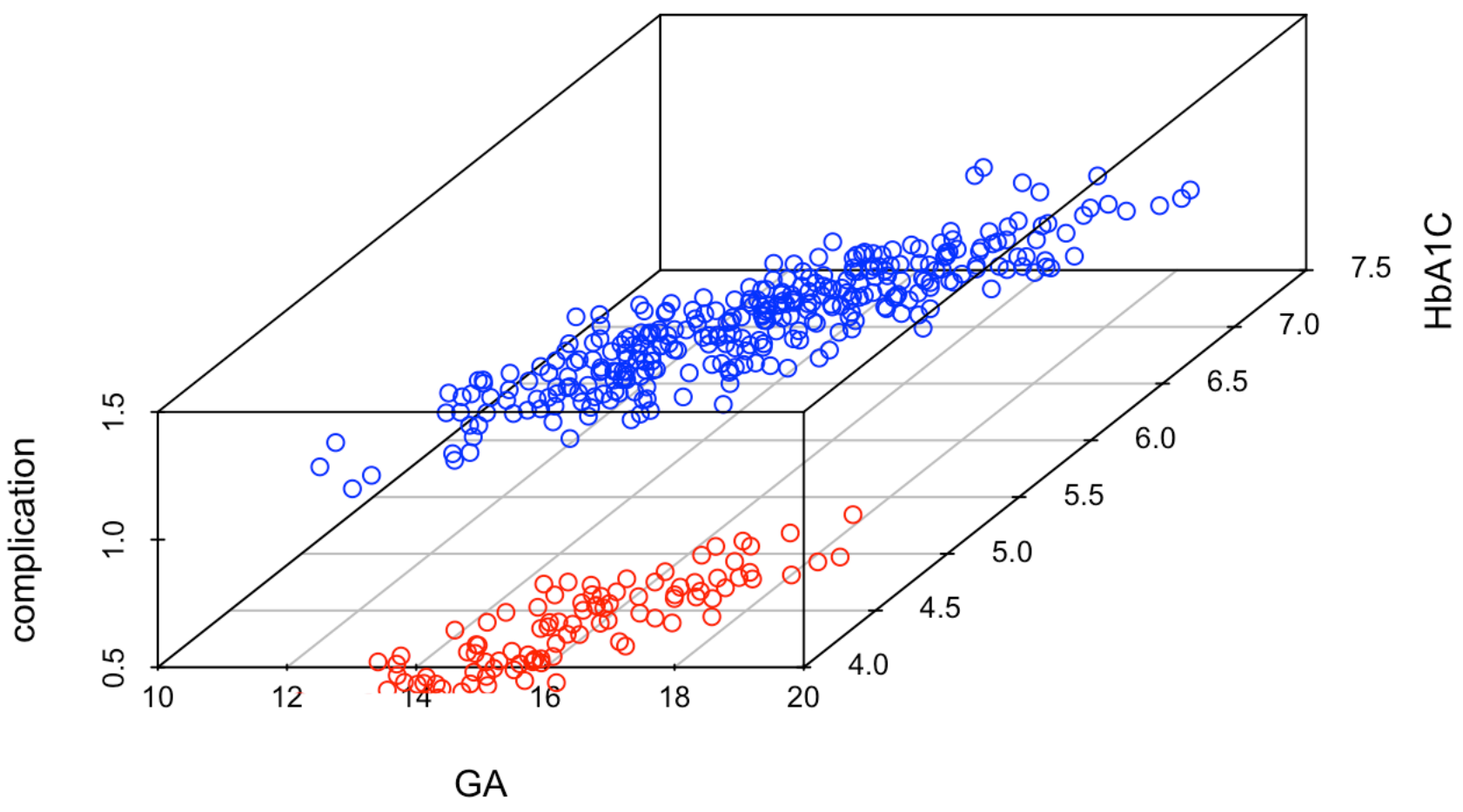
- AIC: 354.86

3. Let’s plot out the model we just built

```
ggplot(data=mydata1)+geom_point(aes(x=GA,y=complication,color = complication))+
  geom_point(aes(x=GA,y= pred_com))
```




```
library(scatterplot3d)
s3d <-scatterplot3d( mydata$GA[mydata$complicatio==1],mydata$HbA1C[mydata$complicatio
==1],mydata$complication[mydata$complicatio==1],
                     color = "blue",
                     xlab= "GA",ylab=  "HbA1C",zlab="complication",
                     main="GA VS HbA1C VS complication")
s3d$points3d(mydata$GA[mydata$complicatio==0],mydata$HbA1C[mydata$complicatio==0],myd
ata$complication[mydata$complicatio==0], col = "red")
```



Conlusion

- Patients having higher GA their babies will have a higher chance to have complications after birth.
- The model to predic the probablity of have conditions is Log(p/(1-p)) = 0.944*GA - 12.069.
- HbA1C is correlated with GA, the linear relation can be described as HbA1C = 0.235*GA + 2.38
- HbA1C can be affected by diabeties condition during pragnency. GA can be used to control diabeties in order to lower the incidence of complications in IDMs.