**Задание ЛОГ1 - Построение и применение однофакторной логистической регрессионной модели**

Выполнил студент 2 курса

группы 09-715(1)

Санамян Артак Размикович

**Текст задания:**

1. Найдите коэффициенты логистической регрессионной модели.
2. Постройте логистическую кривую и облако точек.
3. Выполните прогноз переменной отклика для нескольких новых объектов и нескольких объектов обучающей выборки.
4. Проиллюстрируйте на графике результаты прогнозирования.
5. Сделайте выводы о качестве логистической регрессионной модели.
6. Результаты оформите в виде Word-файла и прикрепите его здесь (Весь R-код поместите в конец файла как приложение).

Источник числовых данных, использованный для выполнения задания: http://archive.ics.uci.edu/ml/datasets/banknote+authentication

**Описание выполненной работы:**

Данные были извлечены из изображений, которые были взяты для оценки процедуры аутентификации банкнот. Для оцифровки использовалась промышленная камера, обычно используемая для проверки печати. Изображения имеют 400 x 400 пикселей. Для извлечения признаков из изображений использовалось вейвлет-преобразование.

Необходимо предложить инструментарий для поддержки принятия решения о подлинности банкноты (или о подделке банкноты).

Переменная отклика – подлинность образца банкноты:

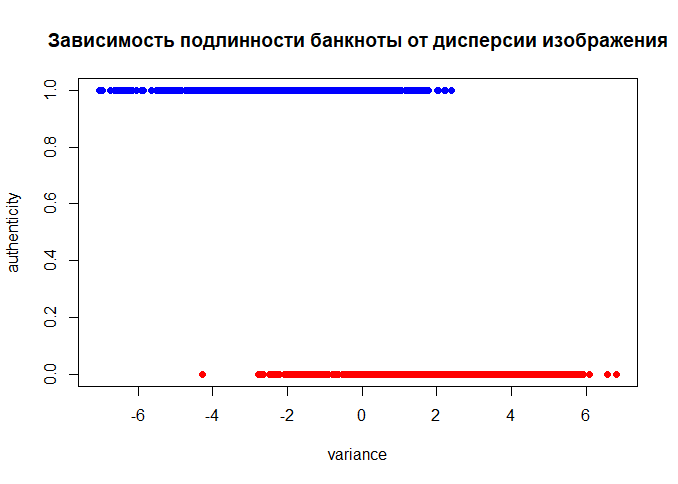
* 1 – подлинная банкнота;
* 0 – поддельная банкнота.

Предикторами являются следующие признаки:

* дисперсия Вейвлет-Преобразованного изображения;
* коэффициент асимметрии Вейвлет-Преобразованного изображения;
* энтропия изображения.

Рассмотрим однофакторный случай: исследуем зависимость подлинности банкноты от дисперсии Вейвлет-Преобразованного изображения.

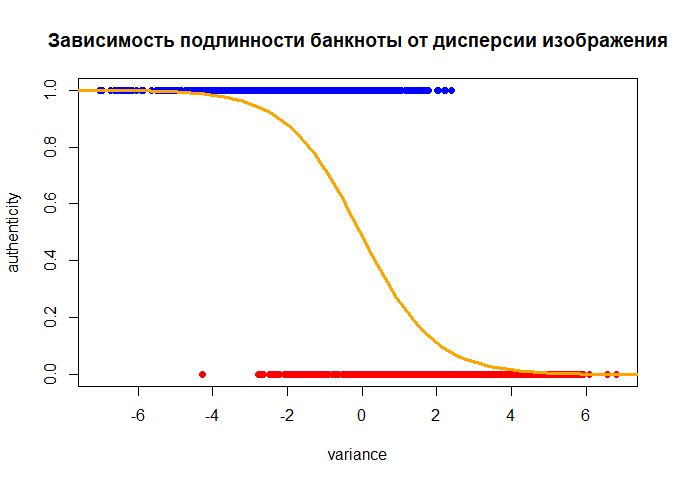
Построим облако точек.



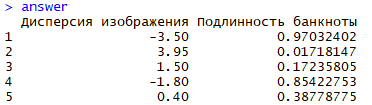
Коэффициенты логистической регрессионной модели:



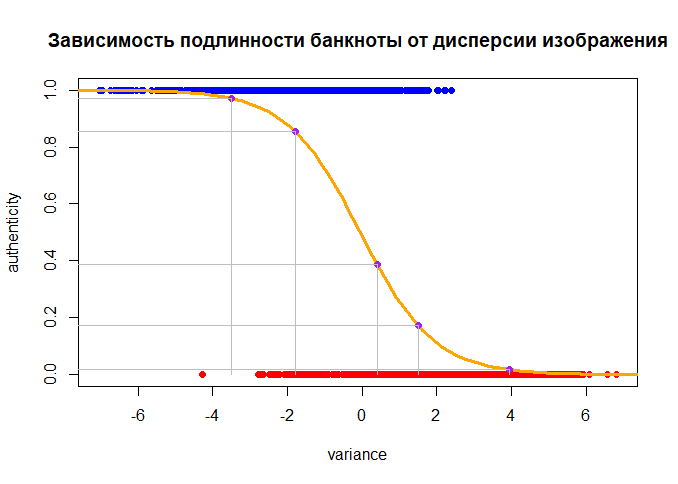
Совместим логистическую кривую с облаком точек:

 Выполним прогноз переменной отклика для нескольких новых объектов. Определим подлинность банкнот со следующими дисперсиями изображения:





Проиллюстрируем на графике результаты прогнозирования:



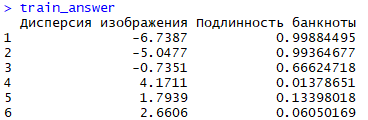
Пусть «пороговое» значение вероятности того, что банкнота окажется подлинной, равно 0.6.

Таким образом, мы получили:

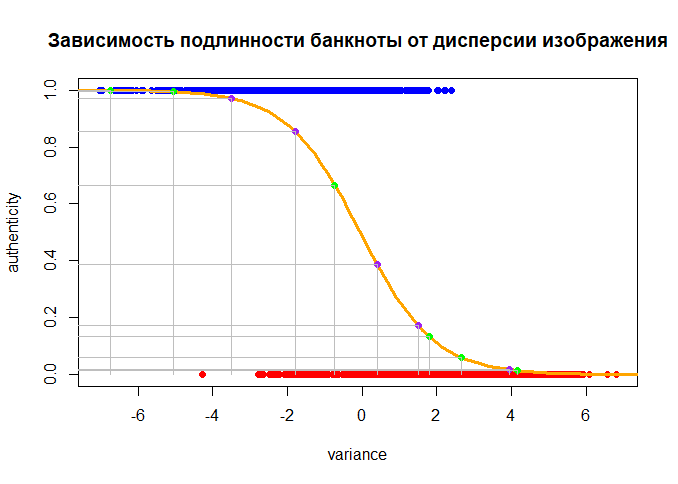
1. Банкноту, дисперсия изображения которой = -3.50, логистическая модель отнесет к классу подлинных (вычисленное значение вероятности попасть в класс подлинных банкнот = 0.97);
2. Банкноту, дисперсия изображения которой = 3.95, логистическая модель отнесет к классу поддельных (вычисленное значение вероятности попасть в класс подлинных банкнот = 0.017);
3. Банкноту, дисперсия изображения которой = 1.50, логистическая модель отнесет к классу поддельных (вычисленное значение вероятности попасть в класс подлинных банкнот = 0.17);
4. Банкноту, дисперсия изображения которой = -1.80, логистическая модель отнесет к классу подлинных (вычисленное значение вероятности попасть в класс подлинных банкнот = 0.85);
5. Банкноту, дисперсия изображения которой = 0.40, логистическая модель отнесет к классу поддельных (вычисленное значение вероятности попасть в класс подлинных банкнот = 0.39).

Выполним прогноз переменной отклика для нескольких объектов обучающей выборки. Определим подлинность банкнот со следующими дисперсиями изображения:





Проиллюстрируем на графике результаты прогнозирования (зеленые точки):



Таким образом, мы получили, что первые три банкноты логистическая модель отнесет к классу подлинных, а следующие три банкноты – к классу поддельных. Все 6 предсказаний оказались верными.

**Код программы:**

# Назначим рабочий директорий

setwd("C://Users/pc/Documents/Магистратура/2 семестр/R/task6 Log1")

# Прочтём данные о банкнотах

bankdata <- read.csv("banknote.txt", sep=",", head=FALSE)

# Выберем дисперсию Вейвлет-Преобразованного изображения(столбец 1)

# и подлинность банкноты(столбец 5)

bankdata = bankdata[c(1, 5)]

bankdata

# Построим облако точек

# для этого запишем фактор (дисперсию изображения) и значение признака подлинности банкноты

# в векторы variance и authenticity, соответственно

variance = bankdata[,1]

variance

authenticity = bankdata[,2]

authenticity

# Чтобы задать размеры облака, нанесём на график все точки, но сделаем их невидимыми

plot(variance, authenticity, type="n", main="Зависимость подлинности банкноты от дисперсии изображения")

# Определим размер выборки (число банкнот)

n = length(variance)

n

for(i in 1:n)

{

color = ifelse(authenticity[i]==1, "blue", "red") # цвет точки

points(variance[i], authenticity[i], type="p",pch=16, col = color, add = "TRUE")

}

# Озаглавим столбцы bankdata

colnames(bankdata)<-c("Variance","Authenticity")

bankdata

# Представим озаглавленные данные bankdata как таблицу

B<-as.data.frame(bankdata)

B

# Вызовем метод максимизации функции правдоподобия glm

mymodel<-glm(formula = Authenticity ~ Variance, family = binomial, data = B)

# Запишем найденные коэффициенты в вектор Teta

Teta=coef(mymodel)

Teta

# Для удобства напишем функцию

fexp = function (x)

{

vspom = exp(Teta[1]+Teta[2]\*x);

return ( vspom/(1+vspom) )

}

#Совместим логистическую кривую с облаком точек

curve(fexp(x),-9, 9, col = "Orange", lwd = 3, add=TRUE)

# Определим подлинность новых банкнот

# с дисперсией изображения -3.5, 3.95, 1.5, -1.8, 0.4

newBanknote = c(-3.5, 3.95, 1.5, -1.8, 0.4)

newBanknote

newy = fexp(newBanknote)

# Нанесём новые точки на график

points(newBanknote, newy, col="Purple",type="p",pch=16,add=TRUE)

# Для наглядности построим вспомогательные линии

nnew = length(newBanknote)

for(i in 1:nnew)

{

segments(newBanknote[i],0, newBanknote[i],newy[i], col="gray",add="TRUE")

segments(newBanknote[i], newy[i], -9, newy[i],col="gray",add="TRUE")

}

# Определим подлинность банкнот из обучающей выборки

# с дисперсией изображения -6.7387, -5.0477, -0.7351, 4.1711, 1.7939, 2.6606

trainBanknote = c(-6.7387, -5.0477, -0.7351, 4.1711, 1.7939, 2.6606)

trainBanknote

trainy = fexp(trainBanknote)

# Нанесём новые точки на график

points(trainBanknote, trainy, col="Green",type="p",pch=16,add=TRUE)

# Для наглядности построим вспомогательные линии

nline = length(trainBanknote)

for(i in 1:nline)

{

segments(trainBanknote[i],0, trainBanknote[i],trainy[i], col="gray",add="TRUE")

segments(trainBanknote[i], trainy[i], -9, trainy[i],col="gray",add="TRUE")

}

solutions = matrix(1:(nnew\*2),ncol=2)

# Запишем в первый столбец матрицы solutions значения дисперсии изображений новых банкнот

solutions[,1] = newBanknote

# Запишем во второй столбец матрицы solutions значения переменной отклика (вероятность того, что банкнота подлинна) для новых банкнот

solutions[,2] = newy

# Сохраним матрицу как фрейм данных

answer = as.data.frame(solutions)

# Добавим заголовки

colnames(answer) = c("Дисперсия изображения", "Подлинность банкноты")

answer

#аналогично для банкнот из обучающей выборки

train\_solutions = matrix(1:(nline\*2),ncol=2)

train\_solutions[,1] = trainBanknote

train\_solutions[,2] = trainy

# Сохраним матрицу как фрейм данных

train\_answer = as.data.frame(train\_solutions)

# Добавим заголовки

colnames(train\_answer) = c("Дисперсия изображения", "Подлинность банкноты")

train\_answer