

A3

Sebastian Doka

2023-03-01

Question 1: A) If I test positive on a Covid Rapid Test, what's the probability that I actually have Covid?

Let Y represent the probability of having covid, with Y=0 for not having covid and Y=1 for having covid. Let X represent the result of the Covid Rapid Test with X=0 for a negative result and X=1 for a positive result.

$$P(Y = 1|X = 1) = P(X=1|Y=1)P(Y=1)/P(X=1)$$

$$P(Y = 1|X = 1) = P(X=1|Y=1)P(Y=1)/P(X=1|Y=0)P(Y=0)+P(X=1|Y=1)P(Y=1)$$

$$P(Y = 1|X = 1) = 0.60.01/0.0050.99+0.6*0.01$$

$$P(Y = 1|X = 1) = 0.5479452$$

Therefore the probability of having covid if testing positive for a Covid Rapid Test is 54.79%.

B) If I test negative on a Covid Rapid Test, what's the probability that I actually have Covid anyway, i.e. the test was wrong?

$$P(Y = 1|X = 0) = P(X=0|Y=1)P(Y=1)/P(X=0)$$

$$P(Y = 1|X = 0) = P(X=0|Y=1)P(Y=1)/P(X=0|Y=0)P(Y=0)+P(X=0|Y=1)P(Y=1)$$

$$P(Y = 1|X = 0) = 0.40.99/0.9950.99+0.4*0.01$$

$$P(Y = 1|X = 0) = 0.0040443$$

Therefore the probability of having covid if testing negative for a Covid Rapid Test is 0.4044%.

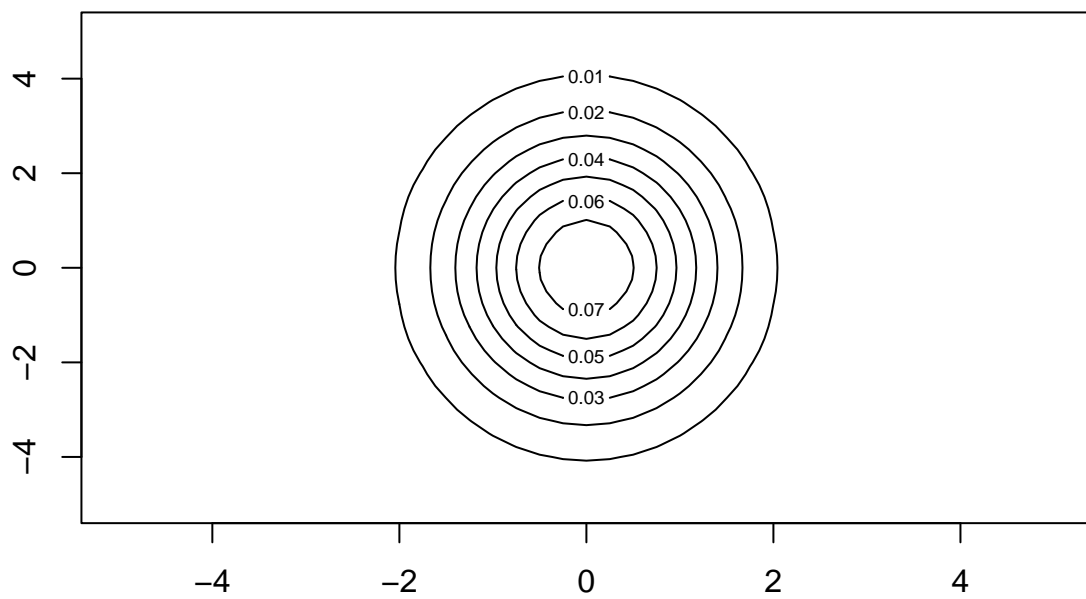
C) If I test negative twice, and we assume the results from the two tests are independent (a questionable assumption), then what's the probability that I actually have Covid?

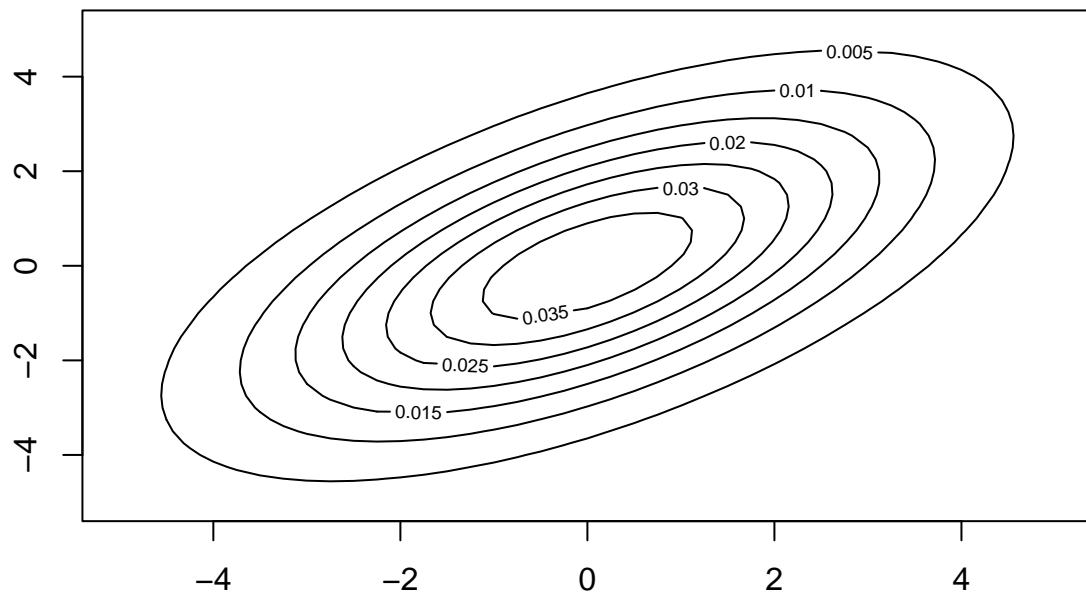
Because we are assuming the two tests are independent therefore the probability of having covid if negative twice for a Covid Rapid Test is the probability of having covid if testing negative once for Covid Rapid squared meaning $0.404^2 = 0.001636\%$.

Question 2:

A)

Warning: package 'mnormt' was built under R version 4.1.3



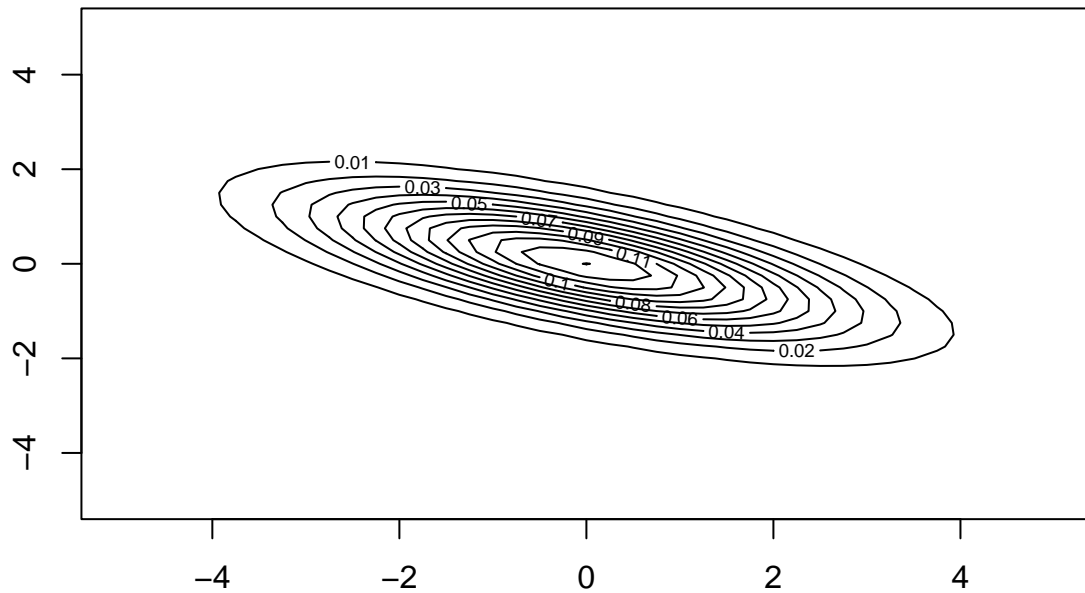


B)

```
library(mnormt)

x <- seq(-5, 5, 0.25)
y <- seq(-5, 5, 0.25)
mu <- c(0,0)
sigma <- matrix(c(3, -1.1, -1.1, 0.9), nrow = 2)
f <- function(x, y) {dmnorm(cbind(x, y), mu, sigma)}
z <- outer(x, y, f)

contour(x, y, z)
```



Question 3

A)

```
## Warning: package 'palmerpenguins' was built under R version 4.1.3

## Warning: package 'caret' was built under R version 4.1.3

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.1.3

## Loading required package: lattice

## [1] "Mean of flipper length in subset male: 222.0323"

## [1] "Standard deviation of flipper length in subset male: 5.77629564667958"

## [1] "Variance of flipper length in subset male: 33.3655913978495"

## [1] "Mean of flipper length in subset female: 212.51724137931"

## [1] "Standard deviation of flipper length in subset female: 3.97869944342165"
```

```
## [1] "Variance of flipper length in subset female: 15.8300492610837"

## [1] "Mean of bill length in subset male: 49.7612903225806"

## [1] "Standard deviation of bill length in subset male: 2.92297535847235"

## [1] "Variance of bill length in subset male: 8.54378494623656"

## [1] "Mean of bill length in subset female: 45.6344827586207"

## [1] "Standard deviation of bill length in subset female: 2.18505898286996"

## [1] "Variance of bill length in subset female: 4.77448275862069"
```

B)

```
mu_m = c(mu_m_flipper, mu_m_bill)
mu_f = c(mu_f_flipper, mu_f_bill)
sd_m = c(sd_m_flipper, sd_m_bill)
sd_f = c(sd_f_flipper, sd_f_bill)
# Gaussian Naive Bayes for 2 classes (male and female) and 2 predictor variables
GNB2 = function(X, pi_m, mu_m, sd_m, mu_f, sd_f){
  # Inputs:
  # X = a dataframe with two columns (the predictors)
  # pi_m = a prior probability for class 'male'
  # All the other inputs are vectors of length 2, corresponding to 2 predictor variables
  # mu_m = a mean vector for class 'male'
  # sd_m = a standard deviation vector for class 'male'
  # mu_f = a mean vector for class 'female'
  # sd_f = a standard deviation vector for class 'female'
  ##
  #Output: a vector of posterior probabilities for class 'male'
  for (i in 1:nrow(X)){
    likelihood_male = dnorm(X$flipper_length_mm[i], mean=mu_m[1], sd = sd_m[1]) *
      dnorm(X$bill_length_mm[i], mean=mu_m[2], sd = sd_m[2])
    likelihood_female = dnorm(X$flipper_length_mm[i], mean=mu_f[1], sd = sd_f[1]) *
      dnorm(X$bill_length_mm[i], mean=mu_f[2], sd = sd_f[2])
    X$posterior_male[i] = (likelihood_male * pi_m) / (likelihood_male * pi_m + likelihood_female * pi_m)
  }

  return(X$posterior_male)
}

library(class)
```

```
## Warning: package 'class' was built under R version 4.1.3
```

```
predictors = df[,c("flipper_length_mm", "bill_length_mm" )]

grid = expand.grid(x=201:240, y=41:60)
classes.grid = GNB2(predictors,0.5,mu_m,sd_m,mu_f,sd_f)
```

```
## Warning: Unknown or uninitialised column: 'posterior_male'.
```

```
classes.grid
```

```
## [1] 0.040050138 0.999956003 0.152523735 0.884373997 0.253707228 0.040034899
## [7] 0.028683401 0.524061564 0.009097386 0.170642826 0.021688749 0.580861841
## [13] 0.067740288 0.233892245 0.027960516 0.728311560 0.008367105 0.948140829
## [19] 0.028556798 0.956672882 0.901468430 0.079309000 0.081440122 0.134134599
## [25] 0.039843125 0.122180611 0.280060517 0.110412695 0.954144651 0.892106862
## [31] 0.008060137 0.012993983 1.000000000 0.906783836 0.850095785 0.019533520
## [37] 0.269597140 0.009634934 0.113875993 0.006785339 0.996893554 0.022178384
## [43] 0.694097408 0.990278824 0.100050621 0.024270145 0.998604473 0.038960948
## [49] 0.082568646 0.042274227 0.859322046 0.020336668 0.997989655 0.278376187
## [55] 0.439876484 0.009038094 0.495520712 0.007624764 0.996973405 0.015937519
## [61] 0.707720273 0.074057942 0.999999829 0.403817447 0.999947566 0.999841673
## [67] 0.315550404 0.995630167 0.348007476 0.729727496 0.894701478 0.278376187
## [73] 0.193758598 0.999861005 0.059920180 0.983147685 0.082568646 0.831356156
## [79] 0.282633882 0.997740373 0.103083555 0.995659138 0.029596457 0.999888929
## [85] 0.141459853 0.963205716 0.109257772 0.999994421 0.526791211 0.999973550
## [91] 0.038260170 0.993331571 0.045376567 0.999486234 0.657304495 0.865886819
## [97] 0.073204004 0.999215189 0.784598496 0.999999751 0.207937976 0.999502889
## [103] 0.170642826 0.008041690 0.997685355 0.007826122 0.087617341 0.834969822
## [109] 0.999877455 0.024218748 0.999989508 0.246517858 0.999999856 0.959758112
## [115] 0.155986112 0.170642826 0.989343604 0.033525322 0.506332216
```

C)

```
#library(palmerpenguins)
df = subset(penguins, species == 'Gentoo')
predictors = df[,c("flipper_length_mm", "bill_length_mm" )]
#predictors = na.omit(predictors)

mu_m = c(mu_m_flipper, mu_m_bill)
mu_f = c(mu_f_flipper, mu_f_bill)
sd_m = c(sd_m_flipper, sd_m_bill)
sd_f = c(sd_f_flipper, sd_f_bill)

library(class)

grid = expand.grid(x=201:240, y=41:60)
classes.grid = GNB2(predictors,0.5,mu_m,sd_m,mu_f,sd_f)
```

```
## Warning: Unknown or uninitialised column: 'posterior_male'.
```

```
#contour(x=201:240, y=41:60, z=matrix(classes.grid, nrow=40),
#      levels=0.5,col="grey", drawlabels=FALSE, lwd=2)

#points(df$flipper_length_mm, df$bill_length_mm, col=df$sex)
```

D)

```
library(palmerpenguins)
df = subset(penguins, species == 'Gentoo')
df = df[,c("sex", "body_mass_g", "flipper_length_mm", "bill_length_mm")]
df = na.omit(df)
```

```
library(caret)
set.seed(1)
ind <- createDataPartition(df$sex, p = .5, list=FALSE)
train = df[ind,]
test = df[-ind,]
```

```
mu_m = c(mu_m_flipper, mu_m_bill)
mu_f = c(mu_f_flipper, mu_f_bill)
var_m = c(mean(var_m_flipper), mean(var_m_bill))
var_f = c(mean(var_f_flipper), mean(var_f_bill))
```

```
library(class)
predictors = df[,c("flipper_length_mm", "bill_length_mm" )]
```

```
grid = expand.grid(x=201:240, y=41:60)
classes.grid = GNB2(predictors, 0.5, mu_m, var_m, mu_f, var_f)
```

```
## Warning: Unknown or uninitialised column: 'posterior_male'.
```

```
#contour(x=201:240, y=41:60, z=matrix(classes.grid, nrow=1), levels=0.5,col="grey", drawlabels=FALSE, l
```

```
#points(df$flipper_length_mm, df$bill_length_mm, col=df$sex)
```

E)