

# Covid-19-false-positive

Maike Heinrich

24 5 2021

## Probability of getting a false positive Covid-19 PCR-test

Getting a positive result in a Covid-19 PCR-test doesn't always mean to actually have the disease. While the probability of being actually healthy when getting a negative test-result is almost 100%, the probability of getting a false positive test varies between around 37% and 43%. This means that more than one third of all people, who are tested positive on Covid-19 are actually healthy. This can be explained by Bayes' Rule, which says that one probability affects another. In this case, the probability of getting the disease, which is the incidence and the accuracy of the test, which is around 99.9%. The following codes show, how this probability is calculated and name the range of the variation of probabilities to get a false positive Covid-19 test.

In this example the current incidence is set on 150 and the accuracy of the PCR-test is set on 99.9%

```
#Covid-19-Incidence of 150/100000 citizens
prev<-0.00150
#number of people tested
N<-100000

#testing people with probability of getting Covid-19 by Incidence
outcome<-sample(c("Corona","Healthy"),N,replace=TRUE, prob=c(prev,1-prev))
N_C<-sum(outcome=="Corona")
N_H<-sum(outcome=="Healthy")

#Accuracy of a Covid-19 Test (99,9%)
accuracy<-0.999

#Calculating the test-outcome with Bayes' Rule
test<-vector("character",N)
test[outcome=="Corona"]<-sample(c("+", "-"), N_C, replace=TRUE, prob=c(accuracy, 1-accuracy))
test[outcome=="Healthy"]<-sample(c("-", "+"), N_H, replace=TRUE, prob=c(accuracy, 1-accuracy))
```

Creating a tabel for the results where “-” means a negative test, “+” stands for a positive test and “Corona” and “Healthy” for the actual result.

```
#Creating a table for the outcome
result<-table(outcome,test)
result
```

```
##          test
## outcome      -      +
##   Corona      0    156
##   Healthy  99750     94
```

People who will get a false positive result (This is a random number based on probability and will vary every time this code will be run)

```
#Number of people per 100000, getting a false positive result
result[2,2]
```

```
## [1] 94
```

Probability of being healthy although the test was positive

```
#Probability in percent of being healthy with getting a positive test
(result[2,2]/(result[2,2]+result[1,2]))*100
```

```
## [1] 37.6
```

Because the upper value is based on probability it is random and varies each time, the code is run. To see the range of this variation a Monte Carlo simulation is made to be able to calculate the 95% confidence intervals - the two probabilities between which 95% of all calculated random probabilities will be found.

```
#Running a Monte Carlo simulation to calculate the 95% confidence interval of this probability
#samplesize for Monte Carlo simulation
B<-10000
MC_prob<-replicate(B,{MC_outcome<-sample(c("Corona","Healthy"),N,replace=TRUE, prob=c(prev,1-prev))
MC_N_C<-sum(MC_outcome=="Corona")
MC_N_H<-sum(MC_outcome=="Healthy"))
MC_test<-vector("character",N)
MC_test[MC_outcome=="Corona"]<-sample(c("+","-"),MC_N_C,replace=TRUE,prob=c(accuracy,1-accuracy))
MC_test[MC_outcome=="Healthy"]<-sample(c("-","+"),MC_N_H,replace=TRUE,prob=c(accuracy,1-accuracy))
MC_result<-table(MC_outcome,MC_test)
MC_prob_healthy<-(MC_result[2,2]/(MC_result[2,2]+MC_result[1,2]))*100
mean(MC_prob_healthy)
})
mean(MC_prob)
```

```
## [1] 40.03135
```

Now the confidence interval can be calculated

```
#Calculation for 95% confidence interval
N<-1000 #samplesize for standard error
X_hat<-mean(MC_prob)/100 #expected value
SE_hat<-sqrt(X_hat*(1-X_hat)/N) #standard error
ci<-c(lower<-X_hat-qnorm(0.975)*SE_hat,upper<-X_hat+qnorm(0.975)*SE_hat)
ci
```

  

```
## [1] 0.3699459 0.4306811
```

In between this range, 95% of the calculated probabilities of being healthy although being tested positiv for Covid-19 can be found.

```
ci[1]*100
```

```
## [1] 36.99459
```

```
ci[2]*100
```

```
## [1] 43.06811
```