03 PPV Handout

Harsh Patel

2024-09-17

#1. #Test Positive: Disease Present: 122.5, Disease Absent: 499.375, Total: 621.875 #Test Negative: Disease Present: 2.5, Disease Absent: 99,375.62, Total: 99,378.12 #Overall Total: Disease Present: 125, Disease Absent: 99,875, Grand Total: 100,000

#a) With trisomy 21 occurring in 1 in 800 births, the expected number of cases in a population of 100,000 is (1/800) × 100,000 = 125. Consequently, 99,875 children are anticipated not to have trisomy 21, which can also be calculated as (799/800) × 100,000 = 99,875.

```
#parameters
prevalence = 1/800
sensitivity = 0.980
specificity = 0.995
population.size = 100000

#expected number with trisomy 21
expected.cases = population.size * prevalence
expected.cases
```

```
## [1] 125
```

```
#expected number with trisomy 21
expected.cases = population.size * prevalence
expected.cases
```

```
## [1] 125
```

```
#expected number without trisomy 21
expected.noncases = population.size - expected.cases
expected.noncases
```

```
## [1] 99875
```

```
#expected number with trisomy 21, tested positive (true pos)
expected.true.positives = expected.cases * sensitivity
expected.true.positives
```

```
## [1] 122.5
```

```
#expected number without trisomy 21, tested positive (false pos)
expected.false.positives = expected.noncases * (1 - specificity)
expected.false.positives
```

```
## [1] 499.375

#total expected positives
total.expected.positives = expected.true.positives + expected.false.positives
total.expected.positives

## [1] 621.875

#expected number with trisomy 21, tested negative (false neg)
expected.false.negatives = expected.cases * (1 - sensitivity)
expected.false.negatives

## [1] 2.5

#expected number without trisomy 21, tested negative (true neg)
expected.true.negatives = expected.noncases * specificity
expected.true.negatives

## [1] 99375.62

#total expected negatives
total.expected.negatives = expected.true.negatives + expected.false.negatives
total.expected.negatives

## [1] 99378.12
```

#c) The probability of having the disease given a positive test is computed as P(D|T+) = P(D and T+) / P(T+). This equals 122.5 / 621.875 ≈ 0.197. This represents the proportion of true positives among all positive test results.

```
#ppv
ppv = expected.true.positives/total.expected.positives
ppv

## [1] 0.1969849
```

#2.

```
#define parameters
population.size = 100000
prevalence = 1/800
sensitivity = 0.980
specificity = 0.995

#create empty vectors to store results
disease.status = vector("numeric", population.size)
test.result = vector("numeric", population.size)

#set the seed for a pseudo-random sample
#set.seed(2018)

#assign disease status (part a)
```

```r
disease.status = sample(c(0,1), size = population.size,
                    prob = c(1 - prevalence, prevalence),
                    replace = TRUE)

#assign test result (part b)
for(k in 1:population.size){

  if(disease.status[k] == 0){
test.result[k] = sample(c(0,1), size = 1,
                        prob = c(specificity, 1 - specificity))

  }

  if(disease.status[k] == 1){
test.result[k] = sample(c(0,1), size = 1,
                        prob = c(1 - sensitivity, sensitivity))
  }

}
#create matrix of disease status and test result (part c)
disease.status.and.test.result = cbind(disease.status, test.result)

#create a table of test result by disease status
addmargins(table(test.result, disease.status))
```

```
##            disease.status
## test.result      0      1    Sum
##         0    99338      2  99340
##         1      536    124    660
##         Sum  99874    126 100000
```

```r
#calculate ppv (part d)
ppv = sum(test.result[disease.status == 1])/sum(test.result)
ppv
```

```
## [1] 0.1878788
```

```r
#calculate npv (part e)
npv = sum(test.result == 0 & disease.status == 0) / sum(test.result == 0)
npv
```

```
## [1] 0.9999799
```

#a) Disease status is assigned based on prevalence. An individual with a 0 does not have the disease, while a 1 indicates the presence of the disease. The probability of 0 is (1 - prevalence), and for 1, it is prevalence.

#b) For individuals with the disease, the test outcome is determined by sensitivity and the false negative rate (1 - specificity). For those without the disease, the outcome relies on

specificity and the false positive rate (1 - sensitivity). In both cases, 0 signifies a negative result and 1 a positive result.

#c) A row showing 0 in both columns indicates a person without trisomy 21 who tested negative, representing a true negative result.

#d) The numerator consists of individuals who tested positive and actually have the disease. This is calculated as sum(test.result[disease.status == 1]), which totals the positive test results for those with the disease. The denominator is the total number of positive test results.

#e) The positive predictive value (PPV) from the simulation slightly differs from the table method, yielding 0.213 compared to 0.197. Some variation is normal in simulations; a larger population would likely yield a result closer to 0.197.

#f) The negative predictive value (NPV) is derived by dividing the number of true negatives by the total number of negative results. Based on the simulation, the estimated NPV is 1.

#3.

#a) #DISEASE (1/800) # • TEST + (0.980) # o 0.001225 (P(D ∩ T+)) # • TEST - (0.020) # o 0.000025 (P(D ∩ T-)) #NO DISEASE (799/800) # • TEST + (0.995) # o 0.99376 (P(D^c ∩ T+)) # • TEST - (0.005) # o 0.00499 (P(D^c ∩ T-))

#b) To find P(D|T+), use the formula: P(D|T+) = P(D and T+) / P(T+). This can be expressed as: [P(D) × P(T+|D)] / [P(D) × P(T+|D) + P(D^c) × P(T+|D^c)] = (1/800 × 0.980) / [(1/800 × 0.980) + (799/800 × 0.995)] ≈ 0.197. This probability accounts for true positives relative to all positive results.

#c) The probability of not having the disease given a negative test is P(D^c | T-) = P(D^c and T-) / P(T-). This can be computed as: [P(D^c) × P(T-|D^c)] / [P(D^c) × P(T-|D^c) + P(D) × P(T-|D)] = (799/800 × 0.995) / [(799/800 × 0.995) + (1/800 × (1 - 0.980))] ≈ 0.999974.

#4.

#a) Apply Bayes' theorem to calculate P(D|T+): P(D|T+) = [P(T+|D) × P(D)] / [P(T+|D) × P(D) + P(T+|D^c) × P(D^c)]. With values of 0.85 for sensitivity, 0.0356 for prevalence, and 0.05 for 1 - specificity, this gives: 0.85 × 0.0356 / [0.85 × 0.0356 + (1 - 0.95) × (1 - 0.0356)] ≈ 0.3856.

#b) The simulation results indicate 5,426 positive tests, with an estimated PPV of 0.0735.

```
#define parameters
population.size = 100000
prevalence = 0.0044
sensitivity = 0.85
specificity = 0.95

#create empty vectors to store results
```

```r
disease.status = vector("numeric", population.size)
test.result = vector("numeric", population.size)

#set the seed for a pseudo-random sample
#set.seed(2018)

#assign disease status
disease.status = sample(c(0,1), size = population.size,
                  prob = c(1 - prevalence, prevalence),
                  replace = TRUE)

#assign test result
for(k in 1:population.size){

  if(disease.status[k] == 0){
test.result[k] = sample(c(0,1), size = 1,
                        prob = c(specificity, 1 - specificity))

  }

  if(disease.status[k] == 1){
test.result[k] = sample(c(0,1), size = 1,
                        prob = c(1 - sensitivity, sensitivity))

  }
}

#calculate expected number of positive tests
sum(test.result)
```

```
## [1] 5280
```

```r
#calculate ppv
ppv = sum(test.result[disease.status == 1])/sum(test.result)
ppv
```

```
## [1] 0.06515152
```

#c) As breast cancer prevalence rises, the PPV increases. More disease cases lead to a higher chance that a positive test is a true positive. This can be understood by examining the PPV formula and extreme prevalence cases. When prevalence is very low, the numerator is close to zero, leading to a low PPV. As prevalence nears 1, the effect of false positives diminishes, resulting in a higher PPV.

```r
#calculations
prev = 0.0382
sens = 0.85
spec = 0.95

numerator = prev*sens
```

```
denominator = prev*sens + (1-spec)*(1-prev)

ppv = numerator/denominator
ppv
```

```
## [1] 0.4030536
```

#c (part 2) The R code provided can calculate PPV using Bayes' theorem, illustrating how PPV relates to disease prevalence and screening implications. Low PPV in low-prevalence diseases can make screening less practical and potentially unethical, as seen with HIV in the past.

#d) Increasing test specificity to 99% has a notable effect. For example, in the 70-80 age group, increasing specificity from 0.403 to 0.771 has a significant impact compared to raising sensitivity from 0.403 to 0.440. This is because a higher specificity reduces false positives, thereby increasing PPV. Sensitivity affects PPV but to a lesser extent than specificity.

```
#test high sensitivity
prev = 0.0382
sens = 0.99
spec = 0.95
numerator = prev*sens
denominator = prev*sens + (1-spec)*(1-prev)
ppv = numerator/denominator
ppv
```

```
## [1] 0.4402151
```

```
#test high specificity
prev = 0.0382
sens = 0.85
spec = 0.99
numerator = prev*sens
denominator = prev*sens + (1-spec)*(1-prev)
ppv = numerator/denominator
ppv
```

```
## [1] 0.7714788
```

#5.

#a) R can be utilized to efficiently compute PPV and NPV using the following equations: PPV = P(D|T+) = [P(D) × P(T+|D)] / P(T+), and NPV = P(D^c | T-) = [P(D^c) × P(T-|D^c)] / P(T-).

```
prevalence = c(0.001, 0.020, 0.060, 0.100)
sensitivity = rep(0.20, 4)
specificity = rep(0.94, 4)

ppv.numerator = prevalence*sensitivity
```

```r
ppv.denominator = ppv.numerator + (1 - prevalence)*(1 - specificity)
ppv = ppv.numerator/ppv.denominator
ppv
```

```
## [1] 0.003325574 0.063694268 0.175438596 0.270270270
```

```r
npv.numerator = (1 - prevalence)*specificity
npv.denominator = npv.numerator + (prevalence)*(1 - sensitivity)
npv = npv.numerator/npv.denominator
npv
```

```
## [1] 0.9991488 0.9829279 0.9484757 0.9136069
```

#b) With increasing prevalence of prostate cancer across age groups, PPV rises. However, NPV decreases as prevalence grows.

#c) The likelihood of having prostate cancer given a positive test increases with the overall prevalence. Higher prevalence means a positive test is more likely to be a true positive, while lower prevalence reduces this likelihood and increases the chance of false positives. The decreasing NPV reflects a higher probability of false negatives as prevalence decreases.

#d) Lowering the positive test cutoff increases the number of positive results, including more false positives. Sensitivity rises as more true cases are detected, but specificity decreases because more non-diseased individuals are incorrectly identified as positive. This adjustment affects test performance, as demonstrated with a hypothetical population example.