Modèles Linéaires Appliqués

Arthur Charpentier

Automne 2020

Rappels #4.4 (tests & vraisemblance)

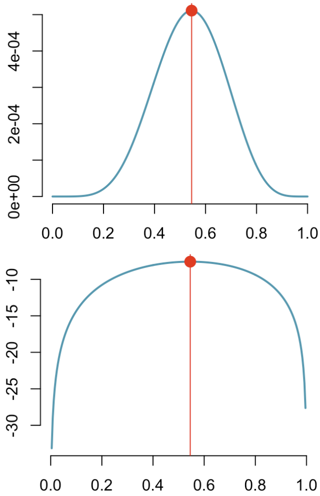
Likelihood (Bernoulli trials)

Here, the likelihood is

$$\theta \mapsto \prod_{i=1}^{n} \theta^{y_i}(1-\theta)^{1-y_i}$$

```
> y = c(1,1,0,1,0,1,0,0,1,0,1)
> L=function(p) prod(dbinom(y,size
    =1,prob = p))
> logL=function(p) sum(dbinom(y,size
    =1,prob = p,log = TRUE))
> neglogL=function(p) -logL(p)
> optim(par = .5,fn = neglogL)
$par
[1] 0.5454102

$value
[1] 7.579102
```

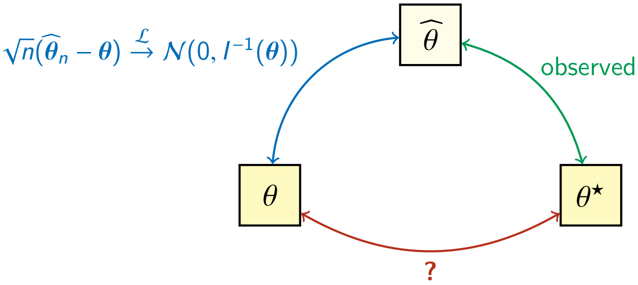


On veut tester $H_0 : \theta = \theta^\star$

Likelihood-based-test

Consider some parametric family, $\mathcal{F} = \{F_\theta\}$.
We want to test $H_0 : \theta = \theta^\star$,
based on some observations $\{y_1, y_2, \cdots, y_n\}$

- θ is unknown
- $\theta^\star$ is given
- from $\{y_1, y_2, \cdots, y_n\}$, we can compute $\widehat{\theta}$ (maximum likelihood)



Likelihood-based-test

Wald test, difference between $\widehat{\theta}$ and $\theta^\star$.

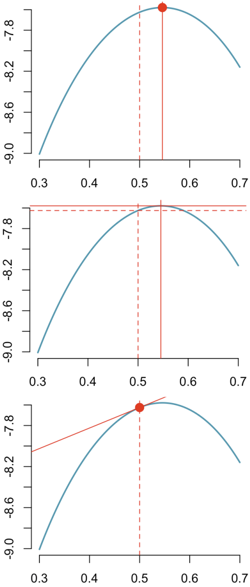
$$T = n\frac{(\widehat{\theta}-\theta^\star)^2}{I^{-1}(\theta^\star)} \overset{\mathcal{L}}{\rightarrow} \chi^2(1)$$

likelihood ratio test, difference between $\log\mathcal{L}(\widehat{\theta})$ and $\log\mathcal{L}(\theta^\star)$

$$T = 2\log\left(\frac{\log\mathcal{L}(\theta^\star)}{\log\mathcal{L}(\widehat{\theta})}\right) \overset{\mathcal{L}}{\rightarrow} \chi^2(1)$$

Score test, difference between $\dfrac{\partial\log\mathcal{L}(\theta^\star)}{\partial\theta}$ and 0.

$$T = \left(\frac{1}{n}\sum_{i=1}^n \frac{\partial\log f_{\theta^\star}(x_i)}{\partial\theta}\right)^2 \overset{\mathcal{L}}{\rightarrow} \chi^2(1)$$



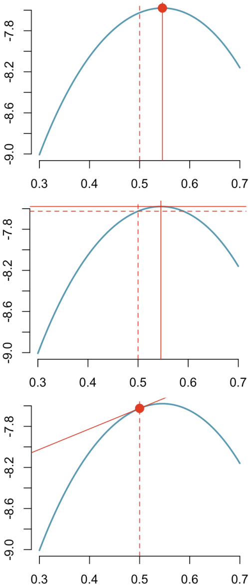
Likelihood-based-test

Wald test, difference between $\widehat{\theta}$ and $\theta^\star$.

$$T = n\frac{(\widehat{\theta}-\theta^\star)^2}{I^{-1}(\theta^\star)} \xrightarrow{\mathcal{L}} \chi^2(1)$$

if H_0 is true.

```
> ny = sum(y==1)
> f = expression(ny*log(p)+(n-ny)*
    log(1-p))
> Df = D(f, "p")
> Df2 = D(Df, "p")
> p = pstar = 0.5
> (IFn=-eval(Df2))
[1] 44
> 1/(pstar*(1-pstar)/n)
[1] 44
> pml=optim(.5, neglogL)$par
> (T = (pml-pstar)^2*IFn)
[1] 0.09073162
```



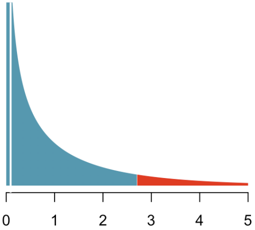
Likelihood-based-test

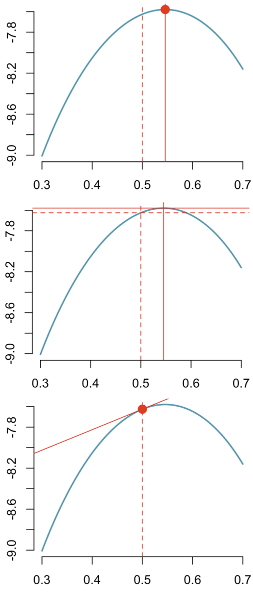
Wald test, difference between $\widehat{\theta}$ and $\theta^\star$.

$$T = n\frac{(\widehat{\theta}-\theta^\star)^2}{I^{-1}(\theta^\star)} \xrightarrow{\mathcal{L}} \chi^2(1)$$

if H_0 is true.

```
> T>qchisq(1-alpha,df=1)
[1] FALSE
> 1-pchisq(T,df=1)
[1] 0.7632491
```





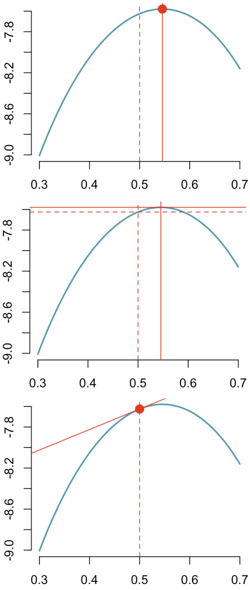
Likelihood-based-test

Likelihood ratio test,
difference between $\log \mathcal{L}(\widehat{\theta})$ and $\log \mathcal{L}(\theta^\star)$

$$T = 2\log\left(\frac{\log \mathcal{L}(\theta^\star)}{\log \mathcal{L}(\widehat{\theta})}\right) \xrightarrow{\mathcal{L}} \chi^2(1)$$

if H_0 is true.

```
> (T = 2*(logL(pml)-logL(pstar)))
[1] 0.09103464
> T>qchisq(1-alpha,df=1)
[1] FALSE
> 1-pchisq(T,df=1)
[1] 0.7628659
```



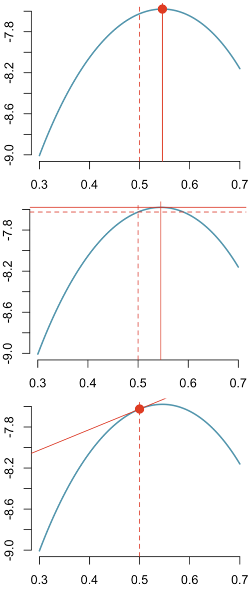
Likelihood-based-test

Score test, difference between $\dfrac{\partial \log \mathcal{L}(\theta^\star)}{\partial\theta}$ and 0.

$$T=\left(\frac{1}{n}\sum_{i=1}^{n}\frac{\partial \log f_{\theta^\star}(x_i)}{\partial\theta}\right)^2 \xrightarrow{\mathcal{L}} \chi^2(1)$$

if H_0 is true.

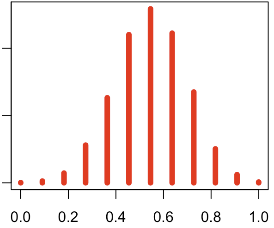
```
> Df = D(f, "p")
> p = pstar
> score=eval(Df)
> (T=score^2/IF)
[1] 0.09090909
> T>qchisq(1-alpha,df=1)
[1] FALSE
> 1-pchisq(T,df=1)
[1] 0.7630246
```

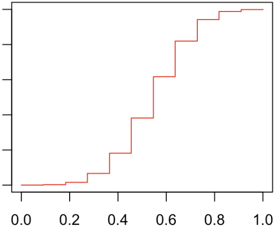


Simulation-based-test (bootstrap)

$\mathbf{y} = \{y_1, \cdots, y_n\}$, with average $\overline{y}$
consider some resampled samples
$\mathbf{y}^{(b)} = \{y_1^{(b)}, \cdots, y_n^{(b)}\}$, with average $\overline{y}^{(b)}$

```
> for(b in 1:9999){
 ys = sample(y,size=n,replace=TRUE)
 m[b] = mean(ys) }
> quantile(m,c(.025,.975))
      2.5%      97.5%
0.2727273 0.8181818
```





Simulation-based-test (bootstrap)

$\mathbf{y} = \{y_1, \cdots, y_n\}$, with average $\overline{y}$

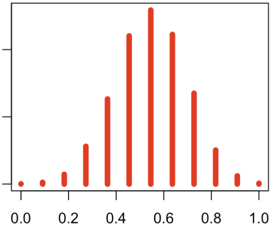
consider some resampled samples

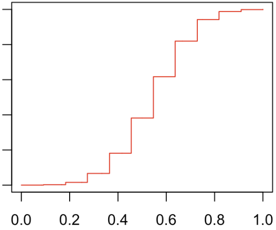
$\mathbf{y}^{(b)} = \{y_1^{(b)}, \cdots, y_n^{(b)}\}$, with average $\overline{y}^{(b)}$

```
> library(boot)
> datbf = function(data,index){d =
    data[index]; mean(d)}
> ys = boot(y,datbf,R=999)
> boot.ci(ys)
BOOTSTRAP CONFIDENCE INTERVAL
Based on 999 bootstrap replicates

Intervals :
Level     Normal              Basic
95%  (0.2546,0.8408)  (0.2727,0.8182)

Level   Percentile            BCa
95%  (0.2727,0.8182)  (0.0909,0.7273)
```





Simulation-based-test (bootstrap)

To compute (numerically) the p-value, generate sample of size n under H_0

```
> for(b in 1:9999){
  ys = sample(0:1,size=n,replace=
    TRUE)
  m[b] = mean(ys) }
> mean(abs(m-phat)> 0)
[1] 0.7744774
```