16:954:581 Probability and Statistical Inference

Method of Moments

Keya Desai (kd706)

---

1. Theory

Method of Moments is a method of estimation of population parameters. It is based on the law of large numbers. If X_1, X2 ..., X_n are independent random variables, the sample mean converges to the distributed mean as the number of observations/samples increases (n -> ∞). If the model has m parameters, we compute the first m moments, obtaining m equations in m variables.

Method of moment involves equating sample moments with theoretical moments. The basic idea behind this form of the method is to:

(1) Equate the first sample moment about the origin $M_1 = \frac{1}{n}\sum_{i=1}^{n} X_i = \bar{X}$ to the first theoretical moment E(X)

(2) Equate the second sample moment about the origin $M_2 = \frac{1}{n}\sum_{i=1}^{n} X_i^2$ to the second theoretical moment $E(X^2)$.

(3) Continue equating sample moments about the origin, M_k, with the corresponding theoretical moments $E(X^k)$, k = 3, 4, ... until you have as many equations as you have parameters.

(4) Solve for the parameters.

The resulting values are called <u>method of moments estimators.</u>

For a 2-parameter distribution, we can form two equations by equating the theoretical mean with the sample mean and equating the theoretical variance with the population/sample variance (2nd moment about the mean). By solving the equations, we can get the estimated value of the parameters of the distribution based on the data.

The estimated parameters using method of moments estimator for the distributions are shown in Table 1. For each distribution, the equations obtained by equating theoretical mean with the sample mean and equating the theoretical variance with the population/sample variance, are solved and the estimation of the parameters in terms of the first and second moment about the sample mean are determined.

Index	Distribution	Mean	Variance	Parameter 1	Parameter 2
1	Point Mass(a)	a	0	$\hat{a} = \hat{\mu}$	-
2	Bernoulli(p)	p	p(1-p)	$\hat{p} = \hat{\mu}$	-
3	Binomial(n,p)	np	np(1-p)	$\hat{n} = \frac{\hat{\mu^2}}{\hat{\mu}-\hat{\sigma^2}}$	$\hat{p} = 1 - \frac{\hat{\sigma^2}}{\hat{\mu}}$
4	Geometric(p)	$\frac{1}{p}$	$\frac{1-p}{p^2}$	$\hat{p} = \frac{1}{\hat{\mu}}$	-
5	Poisson(λ)	λ	λ	$\hat{\lambda} = \hat{\mu}$	-
6	Uniform(a, b)	$\frac{a+b}{2}$	$\frac{(b-a)^2}{12}$	$\hat{a} = \hat{\mu} - \sqrt{3\hat{\sigma^2}}$	$\hat{b} = \hat{\mu} + \sqrt{3\hat{\sigma^2}}$
7	Exponential(β)	β	β^2	$\hat{\beta} = \frac{1}{\hat{\mu}}$	-
8	Normal(μ_0, σ_0^2)	μ_0	σ_0^2	$\hat{\mu_0} = \hat{\mu}$	$\hat{\sigma_0^2} = \hat{\sigma^2}$
9	Gamma(α, β)	$\alpha\beta$	$\alpha\beta^2$	$\hat{\alpha} = \frac{\hat{\mu^2}}{\hat{\sigma^2}}$	$\hat{\beta} = \frac{\hat{\sigma^2}}{\hat{\mu}}$
10	Beta(α, β)	$\frac{\alpha}{\alpha+\beta}$	$\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$	$\hat{\alpha} = \hat{\mu}[\frac{\hat{\mu}(1-\hat{\mu})}{\hat{\sigma^2}} - 1]$	$\hat{\beta} = (1-\hat{\mu})[\frac{\hat{\mu}(1-\hat{\mu})}{\hat{\sigma^2}} - 1]$
11	Chi-Square(p)	p	2p	$\hat{p} = \hat{\mu}$	-

Table 1: Theoretically calculated parameters using the first and second moments.

2. Simulation

Using the parameters derived above, the parameters of each distribution has been computed by generating 100 samples from that distribution. To infer about the accuracy of the estimated parameters, the parameters has been estimated 5 times for the same distribution (randomly generated 100 samples each time) and the average value of the estimated parameter is taken.

```
main <- function() {

 distributions <- c("bernoulli", "binomial", "geometric", "poisson",
    "uniform", "normal", "exponential", "gamma", "beta", "t",
     "chi_squared")

 n = 100   //sample size
 for (distr in distributions){
   method_of_moment(n, distr)  //function generates sample of size n
             //  with distribution ‘distr’ and estimates the parameters.
  }
 }
```

3. Results

The estimated parameters and the actual parameters are shown in the following table:

Index	Distribution	Actual Parameter 1	Actual Parameter 2	Estimated Parameter 1	Estimated Parameter 2
1	Bernoulli (p)	$p = 0.6$	-	$\widehat{p} = 0.53$	-
2	Binomial (n,p)	$n = 10$	$p = 0.7$	$\widehat{n} = 10.0031$	$\widehat{p} = 0.7058$
3	Geometric (p)	$p = 0.7$	-	$\widehat{p} = 0.6667$	-
5.	Poisson(λ)	$\lambda = 0.01$	-	$\widehat{\lambda} = 0.01$	-
6.	Uniform(a,b)	$a = 1$	$b = 5$	$\widehat{a} = 0.7903$	$\widehat{b} = 5.2075$
7.	Exponential(β)	$\beta = 5$	–	$\widehat{\beta} = 5.079$	-
8.	Normal(μ,σ^2)	$\mu = 10$	$\sigma^2 = 9$	$\widehat{\mu} = -0.12$	$\widehat{}^{2} = 7.9084$
9.	Gamma(α, β)	$\alpha = 2$	$\beta = 0.4$	$\widehat{\alpha} = 2.0715$	$\widehat{\beta} = 0.317$
10.	Beta(α, β)	$\alpha = 3$	$\beta =$	$\widehat{\alpha} = 3.153$	$\widehat{\beta} = 5.1286$
1	Chi-Square (p)	$p = 3$	-	$\widehat{p} = 2.8949$	-

3.1 Code output

```
Distribution : bernoulli
[1] "Population Parameter:   p = 0.6"
[1] "Estimated parameter:p_hat = 0.53"

Distribution : binomial
[1] "Population Parameters:    n_ = 10              p = 0.7"
[1] "Estimated parameters: n_hat = 10.5426      p_hat = 0.6829"

Distribution : geometric
[1] "Population Parameter:   p = 0.7"
[1] "Estimated parameter:p_hat = 0.6667"

Distribution : poisson
[1] "Population Parameter:   lambda = 0.01"
[1] "Estimated parameter:lambda_hat = 0.01"

Distribution : uniform
[1] "Population Parameters:    a = 1              b = 5"
[1] "Estimated parameters: a_hat = 1.2546      b_hat = 5.336"

Distribution : normal
[1] "Population Parameters:    mu = 10              sd = 9"
[1] "Estimated parameters: mu_hat = 8.0229      sd_hat = 9.6984"

Distribution : exponential
[1] "Population Parameter:   beta = 5"
[1] "Estimated parameter:beta_hat = 5.079"

Distribution : gamma
[1] "Population Parameters:    alpha = 2              beta = 0.4"
[1] "Estimated parameters: alpha_hat = 1.698      beta_hat = 0.4889"

Distribution : beta
[1] "Population Parameters:    alpha = 3              beta = 5"
[1] "Estimated parameters: alpha_hat = 3.0457      beta_hat = 4.9289"

Distribution : chi_squared
[1] "Population Parameter:   p = 3"
[1] "Estimated parameter:p_hat = 3.3803"
```

4. Conclusion

All the distributions we have considered have at most two parameters and it is easy to calculate the parameters using the first and second moments. The estimated values using method of moments estimator is quite consistent and accurately estimates the parameter values with an error of ~0.5 (which is not that small). Better methods for parametric estimation such as maximum likelihood can be used.

5. Appendix

5.1 Calculating first and second moments

```
## Compute first moment for the sample around origin -> Sample_Mean
first_moment <- function(vec){
  n <- length(data)
  sum_total <- sum(data, na.rm=TRUE)
  return (sum_total/n)
}

## Compute second moment for the sample around the sample mean -> Sample_Variance ##
second_moment <- function(vec){
  mu <- first_moment(vec)
  n <- length(data)
  square_total <- sum((vec - mu)^2, na.rm=TRUE)
  return (square_total/n)
}
```

5.2 Method of Moment for each distribution

```
poisson <- function(vec){
  lambda_hat = first_moment(vec)
  return(lambda_hat)
}

gamma <- function(vec){
  mu = first_moment(vec)
  var = second_moment(vec)
  alpha_hat <- (mu**2)/var
  beta_hat <- var/mu
  list(alpha_hat = alpha_hat, beta_hat = beta_hat)
}

beta <- function(vec){
  mu = first_moment(vec)
  var = second_moment(vec)
  temp = (mu*(1-mu)/var) - 1

  alpha_hat = mu * temp
  beta_hat = (1-mu) * temp

  list(alpha_hat = alpha_hat, beta_hat = beta_hat)
}

geometric <- function(vec){
```

```
  p_hat = 1/first_moment(vec) -1
  return(p_hat)
}

uniform <- function(vec){
  mu =  first_moment(vec)
  var = second_moment(vec)
  a_hat = mu - (var*sqrt(3))
  b_hat = mu + (var*sqrt(3))
  list(a_hat = a_hat, b_hat = b_hat)
}

exponential <- function(vec){
  beta_hat = 1/first_moment(vec)
  return(beta_hat)
}

chi_squared <- function(vec){
  p_hat = first_moment(vec)
  return(p_hat)
}

normal <- function(vec){
  mu_hat <- first_moment(vec)
  sd_hat <- sqrt(second_moment(vec))
 list(mu_hat = mu_hat, sd_hat = sd_hat)
}

binomial <- function(vec){
  mu <- first_moment(vec)
  var <- second_moment(vec)
  n_hat <- mu/(1-(var/mu))
  p_hat <- 1 - (var/mu)
  list(n_hat = n_hat, p_hat = p_hat)
}

bernoulli <- function(vec){
  p_hat = first_moment(vec)
  return(p_hat)
}
```

5.3 Verification of MOM using data from each distribution

```
#### Function to generate data of distribution and estimate the parameters ####
method_of_moment <- function(n, distribution){

  cat(paste("Distribution :", distribution ,"\n" ))

  if( distribution == "poisson" ){
    lambda = 0.01
    vec = rpois(n, lambda)
    lambda_hat = poisson(vec)
    print(paste0("Population Parameter:    ", "lambda = ", lambda))
    print(paste0("Estimated parameter:", "lambda_hat = ",round(lambda_hat,4) ))
  }

  if( distribution == "gamma" ){
    alpha = 2
    beta = 0.4
    vec <- rgamma(n, alpha, scale = beta)
    l <- gamma(vec)
    print(paste0("Population Parameters:    ", "alpha = ", alpha,
             " beta = ", beta))
    print(paste0("Estimated parameters: ", "alpha_hat = ",round(l$alpha_hat,4),
 "beta_hat = ", round(l$beta_hat,4) ) )
  }

  if( distribution == "beta" ){
    alpha = 3
    beta = 5
    vec <- rbeta(n, alpha, beta)
    l <- beta(vec)
    print(paste0("Population Parameters:    ", "alpha = ", alpha, "               beta = ", beta))
    print(paste0("Estimated parameters: ", "alpha_hat = ",round(l$alpha_hat,4), "      beta_hat = ",
round(l$beta_hat,4) ) )
  }

  if( distribution == "normal" ){
    mu = 10
    sd = 9
    vec <- rnorm(n, mu, sd)
    l <- normal(vec)
    print(paste0("Population Parameters:    ", "mu = ", mu, "               sd = ", sd))
    print(paste0("Estimated parameters: ", "mu_hat = ",round(l$mu_hat,4), "      sd_hat = ",
round(l$sd_hat,4) ) )
  }

  if( distribution == "geometric" ){
    p = 0.7
    vec = rgeom(n, p)
```

```
    p_hat = geometric(vec)
    print(paste0("Population Parameter:   ", "p = ", p))
    print(paste0("Estimated parameter:", "p_hat = ",round(p_hat,4) ))
  }

  if( distribution == "uniform" ){
    a = 1
    b = 5
    vec <- runif(n, a, b)
    l <- uniform(vec)
    print(paste0("Population Parameters:    ", "a = ", a, "               b = ", b))
    print(paste0("Estimated parameters: ", "a_hat = ",round(l$a_hat,4), "      b_hat = ", round(l$b_hat,4) )
)
  }

  if( distribution == "exponential" ){
    beta = 5
    vec = rexp(n, beta)
    beta_hat = exponential(vec)
    print(paste0("Population Parameter:   ", "beta = ", beta))
    print(paste0("Estimated parameter:", "beta_hat = ",round(beta_hat,4) ))
  }

  if( distribution == "chi_squared" ){
    p = 3
    vec = rchisq(n, p)
    p_hat = chi_squared(vec)
    print(paste0("Population Parameter:   ", "p = ", p))
    print(paste0("Estimated parameter:", "p_hat = ",round(p_hat,4) ))
  }


  if( distribution == "binomial" ){
    n_ = 10
    p = 0.7
    vec <- rbinom(n, n_, p)
    l <- binomial(vec)
    print(paste0("Population Parameters:    ", "n_ = ", n_, "               p = ", p))
    print(paste0("Estimated parameters: ", "n_hat = ",round(l$n_hat,4) , "      p_hat = ", round(l$p_hat,4))
)
  }

  if( distribution == "bernoulli" ){
    p = 0.6
    vec = rbern(n, p)
    p_hat = bernoulli(vec)
    print(paste0("Population Parameter:   ", "p = ", p))
    print(paste0("Estimated parameter:", "p_hat = ", round(p_hat,4) ))
  }
```

```
  if( distribution == "t" ){
    v = 3
    vec = rt(n, df = v)
    v_hat = t(vec)
    print(paste0("Population Parameter:    ", "v = ", v))
    print(paste0("Estimated parameter:", "v_hat = ",round(v_hat,4) ))
  }

  cat("\n")
}
```

6. References

[1] https://newonlinecourses.science.psu.edu/stat414/node/193/
[2] https://www.math.arizona.edu/~jwatkins/M_moments.pdf