##tongue_cancer analysis

Chang Tu

```
##tongue_cancer  analysis
cancerdata <- read.csv("~/workspace/Baysian-inference/PART 2/Term 4  Lecture 1 materi
als-20211023/tongue_cancer.csv",header=TRUE)
str(cancerdata)
```

```
## 'data.frame':    20 obs. of  6 variables:
##  $ Year      : int  2018 2018 2018 2018 2018 2018 2018 2018 2018 2018 ...
##  $ Desc      : chr  "Tongue - C01-C02" "Tongue - C01-C02" "Tongue - C01-C02" "Tong
ue - C01-C02" ...
##  $ Sex       : chr  "AllSex" "AllSex" "AllSex" "AllSex" ...
##  $ Demography: chr  "Northland" "Waitemata" "Auckland" "Counties Manukau" ...
##  $ Cases     : int  7 23 14 12 19 3 10 4 6 3 ...
##  $ Population: int  185800 615100 493300 567000 421000 113400 249700 49500 172300
121300 ...
```

```
#only a small dataset so why not just print it to have a look

cancerdata
```

Year	Desc	Sex	Demography	Cases	Population
<int>	<chr>	<chr>	<chr>	<int>	<int>
2018	Tongue - C01-C02	AllSex	Northland	7	185800
2018	Tongue - C01-C02	AllSex	Waitemata	23	615100
2018	Tongue - C01-C02	AllSex	Auckland	14	493300
2018	Tongue - C01-C02	AllSex	Counties Manukau	12	567000
2018	Tongue - C01-C02	AllSex	Waikato	19	421000
2018	Tongue - C01-C02	AllSex	Lakes	3	113400
2018	Tongue - C01-C02	AllSex	Bay of Plenty	10	249700
2018	Tongue - C01-C02	AllSex	Tairawhiti	4	49500
2018	Tongue - C01-C02	AllSex	Hawke's Bay	6	172300
2018	Tongue - C01-C02	AllSex	Taranaki	3	121300

1-10 of 20 rows Previous **1** 2 Next

```
rates <- cancerdata$Cases / cancerdata$Population
summary(rates)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 1.642e-05 2.409e-05 3.524e-05 4.371e-05 4.605e-05 1.349e-04
```

```
#may make more  sense if expressed as rate per 100000
rates100000 <- rates*100000
summary(rates100000)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.642   2.409   3.524   4.371   4.605  13.493
```

```
rates.df <-  data.frame(cancerdata$Demography,rates100000,rates,
                        cancerdata$Cases,cancerdata$Population)
names(rates.df) <-  c("Dhb","rate_100000","rate","Cases")
rates.df
```

Dhb <chr>	**rate_100000** <dbl>	**rate** <dbl>	**Cases** <int>	 <int>
Northland	3.767492	3.767492e-05	7	185800
Waitemata	3.739229	3.739229e-05	23	615100
Auckland	2.838030	2.838030e-05	14	493300
Counties Manukau	2.116402	2.116402e-05	12	567000
Waikato	4.513064	4.513064e-05	19	421000
Lakes	2.645503	2.645503e-05	3	113400
Bay of Plenty	4.004806	4.004806e-05	10	249700
Tairawhiti	8.080808	8.080808e-05	4	49500
Hawke's Bay	3.482298	3.482298e-05	6	172300
Taranaki	2.473207	2.473207e-05	3	121300


```
#What is the overall rate
rawrate <- sum(cancerdata$Cases) /sum(cancerdata$Population)
rawrate*100000 #3.634198
```

```
## [1] 3.634198
```

```
## Basic analysis to  estimate assumed common rate for  all  DHBs
##Simplistic model  but but useful  as  base to  compare against.

##Gamma prior
a <- 3/100000
b <- 1      #Like saying prior  evidence is  equivalent  to
            #one extra tiny DHB of size 1

totcases <- sum( cancerdata$Cases)
totpop <- sum(cancerdata$Population)

##update to get  parameters of the posterior, using conjugacy

apost <- a + totcases
bpost <- b + totpop

##Compute posterior  summaries
postmean <- apost/bpost
post_median <-  qgamma(0.5,shape=apost,rate = bpost)

postmean * 100000
```

```
## [1] 3.634198
```

```
post_median * 100000
```

```
## [1] 3.627357
```

```
q025 <- qgamma(0.025,shape=apost,rate=bpost)
q975 <-  qgamma(0.975,shape=apost,rate=bpost)

exact_quantiles <- 100000*c(q025,post_median,q975)
exact_quantiles
```

```
## [1] 3.118513 3.627357 4.188762
```

```
# Simulation approach - first look at modest size Monte Carlo  sample
post_lambda100 <- rgamma(n=100,shape=apost,rate=bpost)

##check quantiles
post_quantiles100 <-
  quantile(post_lambda100,probs=c(0.025,0.5,0.975))

post_quantiles100
```

```
##         2.5%          50%        97.5%
## 3.148095e-05 3.709534e-05 4.206509e-05
```

```
exact_quantiles
```

```
## [1] 3.118513 3.627357 4.188762
```

```
#check_mean
post_mean100 <- mean(post_lambda100)
100000*post_mean100  #3.662318
```

```
## [1] 3.680495
```

```
exact_mean <- apost / bpost
100000*exact_mean   #3.634198
```

```
## [1] 3.634198
```

```
#check standard deviation
post_sd100 <-  sd(post_lambda100)


##Monte Carlo  error for the posterior mean is
MCerror <- post_sd100 / sqrt(100)
MCerror   #tiny
```

```
## [1] 3.074952e-07
```

```
#makes more sense multiplied by 100000, as per the rates themselves
#  Var(C\theta) = C^2 x Var(\theta) so
# sd(C\theta)    = C X sd(\theta)

100000 * MCerror #0.02695065, still pretty small
```

```
## [1] 0.03074952
```

```
                #MC mean is about 1 MC standard error  from the exact mean



##see what  happens for  a bigger posterior sample

post_lambda1000 <- rgamma(n=1000,shape=apost,rate=bpost)

post_quantiles1000 <- quantile(post_lambda1000,probs=c(0.025,0.5,0.975))
##Compare true and simulation results

exact_quantiles
```

```
## [1] 3.118513 3.627357 4.188762
```

```
100000*post_quantiles100
```

```
##     2.5%      50%    97.5%
## 3.148095 3.709534 4.206509
```

```
100000*post_quantiles1000
```

```
##     2.5%      50%    97.5%
## 3.100291 3.623775 4.222505
```

```
#tail quantiles looking pretty  good by the time Monte Carlo
#simulation size reaches 1000

#MC error for nsim=1000

post_sd1000 <-  100000*sd(post_lambda1000)
post_sd1000
```

```
## [1] 0.2743784
```

```
MC_error1000 <- post_sd1000/sqrt(1000)
MC_error1000
```

```
## [1] 0.008676607
```

```
post_mean1000 <- 100000*mean(post_lambda1000)
post_mean1000
```

```
## [1] 3.635492
```

```
100000*exact_mean  #So the MC mean is just over MC standard error from
```

```
## [1] 3.634198
```

```
                  # the true mean
                 #The MC error is fairly trivial though and represents
                 # 1/sqrt(1000) = 3.2% of the posterior standard deviation

# Monte Carlo for  a more complex estimand and model -----------------------
#estimand is "thing to be estimated"

#Instead of  simple common  rate model, let's go to the
#other extreme and let  each DHB have its own parameter;
#probability that each DHB has highest rate among all DHBs.

#Then we can ask questions like "what is the probability that  each  DHB
#" has the highest underlying rate among all DHBs"
#" How does a particular  DHB rank in a 'league table' of rates by DHB

# we will use the same prior  for  each DHB
#\lambda ~ gamma(a,b)  a = 3/100000; b=1

fulla_post <-  a + cancerdata$Cases        #vector
fullb_post <- b +   cancerdata$Population  #vector

fulla_post
```

```
##  [1]  7.00003 23.00003 14.00003 12.00003 19.00003  3.00003 10.00003  4.00003
##  [9]  6.00003  3.00003  9.00003  4.00003 12.00003  7.00003  1.00003  4.00003
## [17]  3.00003 20.00003  1.00003 15.00003
```

```
fullb_post
```

```
##  [1] 185801 615101 493301 567001 421001 113401 249701  49501 172301 121301
## [11]  66701 181701 153901 315901  46801 155501  32401 560801  60901 307401
```

```
## rgamma is partially vectorised; Easiest to  loop
## over simulations and on each iteration generate   the vector of lambda
##lambda values for the 20 DHBs
## also  need to  work out the maximum and rank for  each  set  of  lambdas
##generated

M <- 1000 ##number of  draws from  the posterior
n <- length(rates)  #number of groups - DHBs in this case

##Set-up structures  for  storing output

post_fullambda <- matrix(nrow=M,ncol=n )

post_max <- matrix(nrow=M,ncol=n)

post_rank  <- matrix(nrow=M,ncol=n)

for (i in 1:M ) {
  ##can probably draw gammas for all DHBs in one-hit
  fullambda <- rgamma(n,shape=fulla_post,rate=fullb_post)

  ranks <- rank(fullambda)
  ismax <- (ranks == max(ranks) )

  post_fullambda[i,] <- fullambda
  post_rank[i,]  <- ranks
  post_max[i,]  <- ismax

  }

##check results

##posterior  quantiles for  each DHB

fullpost_quantiles <-  apply(post_fullambda,MARGIN=2,FUN=quantile,
                             probs=c(0.025,0.5,0.975))

fullpost_quantiles.df <-  data.frame(rates.df$Dhb,t(100000*fullpost_quantiles))


fullpost_quantiles.df <-
cbind(fullpost_quantiles.df,rates.df$Cases)

names(fullpost_quantiles.df) <- c("DHB","q025","q50","q975","cases")
fullpost_quantiles.df
```

DHB <chr>	**q025** <dbl>	**q50** <dbl>	**q975** <dbl>	**cases** <int>
Northland	1.52771418	3.665952	6.814496	7
Waitemata	2.43392728	3.664628	5.399270	23
Auckland	1.60349855	2.751969	4.525735	14
Counties Manukau	1.10817269	2.053547	3.519880	12
Waikato	2.82912502	4.413380	6.801002	19

DHB <chr>	q025 <dbl>	q50 <dbl>	q975 <dbl>	cases <int>
Lakes	0.55107164	2.268146	6.241065	3
Bay of Plenty	1.82512512	3.868801	6.898003	10
Tairawhiti	2.14642621	7.164522	17.600046	4
Hawke's Bay	1.31000363	3.344196	7.013541	6
Taranaki	0.58137076	2.224773	5.619313	3

1-10 of 20 rows Previous **1** 2 Next

```
##posterior quantiles for  each DHB's rank
fullpost_ranks_quantiles.df <-
  apply(post_rank,MARGIN=2,FUN=quantile,
                                  probs=c(0.025,0.5,0.975))

fullpost_ranks_quantiles.df <-
 data.frame( t(apply(post_rank,MARGIN=2,FUN=quantile,
        probs=c(0.025,0.5,0.975)) ) )

fullpost_ranks_quantiles.df$Dhb <- rates.df$Dhb
fullpost_ranks_quantiles.df
```

X2.5. <dbl>	X50. <dbl>	X97.5. <dbl>	Dhb <chr>
3.000	12	18	Northland
6.000	11	16	Waitemata
3.000	8	14	Auckland
1.975	5	11	Counties Manukau
8.000	14	18	Waikato
1.000	6	17	Lakes
4.000	12	18	Bay of Plenty
6.000	18	20	Tairawhiti
2.000	10	17	Hawke's Bay
1.000	6	16	Taranaki

1-10 of 20 rows Previous **1** 2 Next

```
names(fullpost_ranks_quantiles.df)[1:3] <- c("q025","q50","q975")

fullpost_ranks_quantiles.df
```

q025 <dbl>	q50 <dbl>	q975 <dbl>	Dhb <chr>

q025 <dbl>	q50 <dbl>	q975 <dbl>	Dhb <chr>
3.000	12	18	Northland
6.000	11	16	Waitemata
3.000	8	14	Auckland
1.975	5	11	Counties Manukau
8.000	14	18	Waikato
1.000	6	17	Lakes
4.000	12	18	Bay of Plenty
6.000	18	20	Tairawhiti
2.000	10	17	Hawke's Bay
1.000	6	16	Taranaki

1-10 of 20 rows Previous **1** 2 Next

```
##posterior probability that rate in each DHB is the maximum

fullpost_max <- colMeans(post_max)
fullpost_max.df <- data.frame(rates.df$Dhb,fullpost_max)
names(fullpost_max.df) <- c("Dhb","prob")

##What about probability in the top 5

intop5 <- (post_rank >= 16)

Prtop5 <- colMeans((intop5) )

Prtop5.df <- data.frame(rates.df$Dhb,Prtop5)
Prtop5.df
```

rates.df.Dhb <chr>	Prtop5 <dbl>
Northland	0.159
Waitemata	0.064
Auckland	0.007
Counties Manukau	0.001
Waikato	0.242
Lakes	0.064
Bay of Plenty	0.174
Tairawhiti	0.760
Hawke's Bay	0.134
Taranaki	0.045