

# coded\_distributions

Lin Yang, Yida Wang

2022-05-04

```
source("shared_code/setup.R")

## Warning: package 'tidyverse' was built under R version 4.1.2
## -- Attaching packages ----- tidyverse 1.3.1 --
## v ggplot2 3.3.5      v purrr  0.3.4
## v tibble  3.1.6      v dplyr  1.0.8
## v tidyr   1.2.0      v stringr 1.4.0
## v readr   2.0.1      v forcats 0.5.1
## Warning: package 'ggplot2' was built under R version 4.1.2
## Warning: package 'tibble' was built under R version 4.1.2
## Warning: package 'tidyr' was built under R version 4.1.2
## Warning: package 'dplyr' was built under R version 4.1.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## Warning: package 'lubridate' was built under R version 4.1.3
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##   date, intersect, setdiff, union
## Warning: package 'extraDistr' was built under R version 4.1.3
##
## Attaching package: 'extraDistr'
##
## The following object is masked from 'package:purrr':
##
##   rdunif
## Warning: package 'MASS' was built under R version 4.1.2
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##   select
```

```

source("shared_code/data_cleaning.R")

## Rows: 22038 Columns: 8

## -- Column specification -----
## Delimiter: ","
## chr (4): ID, Month, Nature, time
## dbl (4): Season, Latitude, Longitude, Wind.kt
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
## New names:
## * id -> id...1
## * time -> time...2
## * latitude -> latitude...3
## * longitude -> longitude...4
## * wind_kt -> wind_kt...5
## * ...

## Warning: `add_rownames()` was deprecated in dplyr 1.0.0.
## Please use `tibble::rownames_to_column()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.

## Joining, by = "id"

```

## Little Sigma<sup>2</sup>

```

#function calculating beta in inverse gamma distribution
beta_gamma <- function(dat, B) {
  res = NULL
  for (j in 1:700) {
    subdat = dat %>% filter(i == j)
    y = subdat[, 2]
    x = cbind(rep(1, nrow(subdat)), subdat[, 3:6]) %>% as.matrix()
    beta = 0.5*(sum((y - x %*% t(B[j, ]))^2))
    res = rbind(res, beta)
  }
  return(sum(res))
}

#test function
B <- data.frame(matrix(1, nrow = 700, ncol = 5))
#beta_gamma <- beta_gamma(dt, B)

sigmasq <- function(dat, B) {
  alpha = nrow(dat)/2
  beta = beta_gamma(dat, B)
  sigmasq = rinvgamma(1, alpha = alpha, beta = beta)
  return(sigmasq)
}

set.seed(2022)
sigmasq(dt, B)

```

```
## [1] 5.815114
```

## Big B

```
#function calculating beta in MVN
beta_i <- function(dat, siginv, sigmasqinv, mu) {
  res = rep(NA,5)
  newdat <- dat

  for (j in 1:700) {
    j = 1
    subdat = newdat %>% dplyr::filter(i == j)
    y = subdat[, 2]
    x = cbind(rep(1, nrow(subdat)), subdat[, 3:6]) %>% as.matrix()
    I = diag(x = 1, nrow(subdat), nrow(subdat))
    Vinv = solve(siginv + sigmasqinv*(t(x) %*% I %*% x))
    M = sigmasqinv*(t(y)%*% I %*%x) + t(mu) %*% siginv
    betai = mvrnorm(1, Vinv %*% t(M), Vinv)
    res = rbind(res, betai)
    j = j+1
  }
  return(res)
}

siginv <- matrix(1, nrow = 5, ncol = 5)
sigmasqinv = 0.2
mu <- c(1,1,1,1,1)
aa <- beta_i(dt, siginv, sigmasqinv, mu) %>% na.omit()
```