BDA week 4

Solutions

## Exercise 1

### Task a

Formula: $ p(, \_1 … *n | y\_1 … y\_n) p()* {i=1}^n p(\_i | ) p(y\_i | \_i) $

Stan-model:

model\_a = "  
data {  
 int<lower=0> n;  
 real y[n];  
}  
  
parameters {  
 real alpha;  
 real theta[n];  
}  
  
model {  
 alpha ~ p()  
 for i in 1:n {  
 theta[i] ~ p(alpha)  
 y[i] ~ p(theta[i])  
 }  
}  
"

### Task b

Formula:

Stan-model:

model\_b = "  
data{  
 int<lower=0> n;  
 int<lower=0> m;  
 real y[n\*m];  
}  
  
parameters {  
 real alpha;  
 real mu[m];  
 real theta[m][n];  
}  
  
model {  
 alpha ~ p()  
 for i in 0:m-1 {  
 mu[i] ~ p(alpha)  
 for j in 0:n-1 {  
 theta[i][j] ~ p(mu[i])  
 y[m\*i+j] ~ p(theta[m\*i+j])  
 }  
 }  
}  
"

### Task c

Formula:

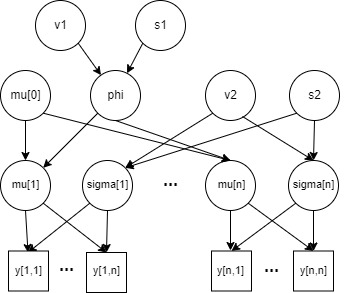
Stan-model:

model\_c = "  
data {  
 int<lower=0> n;  
 real x[n];  
 real y[n];  
}  
  
parameters {  
 real alpha;  
 real gamma;  
 real mu[n];  
 real sigma[n];  
}  
  
model {  
 alpha ~ p()  
 gamma ~ p()  
 for i in 1:n {  
 mu[i] ~ p(alpha)  
 sigma[i] ~ p(gamma)  
 y[i] ~ p(mu[i]\*x[i], sigma[i]\*x)  
 }  
}  
"

### Task d

Stan-model:

model\_d = "  
data {  
 int<lower=0> n;  
 int<lower=0> J;  
 real y[J][n]  
}  
  
parameters {  
 real v1;  
 real v2;  
 real s1;  
 real s2;  
 real phi;  
 real mu0;  
 real mu[J];  
 real sigma[J];  
}  
  
model {  
 v1 ~ p()  
 v2 ~ p()  
 s1 ~ p()  
 s2 ~ p()  
 mu0 ~ norm(0, 10^6)  
 phi ~ inv\_chi2(v1, s1)  
 for j in 1:J {  
 sigma[j] ~ inv\_chi2(v2, s2)  
 mu[j] ~ norm(mu0, phi)  
 for i in 1:n {  
 y[j][i] ~ norm(mu[j], sigma[j])  
 }  
 }  
}"



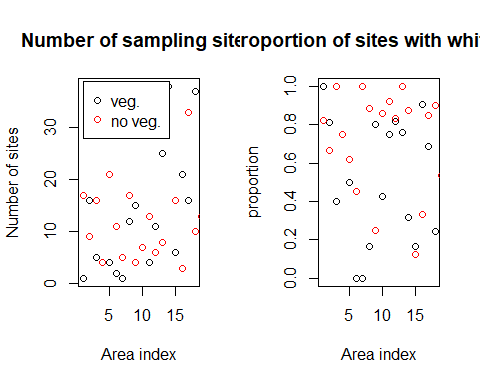
The diagram

## Exercise 2

In this exercise, we continue the analysis of the white fish larval areas (week 2, exercise 3). We are again interested in analysing whether or not bottom vegetation affects white fish larvae occurrence probability. However, instead of having a common probability of presence parameter across the Gulf of Bothnia, we expand the model so that it allows the probability of presence to vary between sampling areas. This modification to the model encodes an assumption that some areas may be more preferable to white fish than others.

Let’s first explore the data a bit more.

# Read the data  
data = read.csv("white\_fishes\_data.csv")  
# Form a data table for sites without bottom vegetation  
y.noveg = table(data$AREANAME[data$BOTTOMCOV==0], data$WHIBIN[data$BOTTOMCOV==0])  
colnames(y.noveg) <- c("y=0","y=1")  
N.noveg = rowSums(y.noveg)  
# Form a data table for sites with bottom vegetation  
y.veg = table(data$AREANAME[data$BOTTOMCOV==1], data$WHIBIN[data$BOTTOMCOV==1])  
colnames(y.veg) <- c("y=0","y=1")  
N.veg = rowSums(y.veg)  
  
par(mfrow=c(1,2))  
plot(N.veg, main="Number of sampling sites", xlab="Area index", ylab="Number of sites")  
points(N.noveg, col="red")  
legend(1, 39, c("veg.","no veg."),col=c("black","red"), pch=1, cex=1, box.lty=1)  
plot(y.veg[,2]/N.veg, main="proportion of sites with whitefish", xlab="Area index", ylab="proportion")  
points(y.noveg[,2]/N.noveg, col="red")



print(y.veg)

##   
## y=0 y=1  
## Bjuroklubb 0 1  
## Bygdea 3 13  
## Haaparanta 3 2  
## Hailuoto 1 3  
## Harnosand 2 2  
## Hornslandet 2 0  
## Lohtaja 1 0  
## Luvia 10 2  
## Mikkelinsaaret 3 12  
## Mjolefjarden 4 3  
## Nordingra 1 3  
## Pietarsaari 2 9  
## Pitea 6 19  
## Pori 26 12  
## Siipyy 5 1  
## Storsand 2 19  
## Tore 5 11  
## Vaasa 28 9

The first figure above shows the number of sampling sites for each of the 19 study areas and both bottom vegetation types (with and without). The second figure shows the proportion of the sites with white fish larvae within each area and bottom vegetation type. It is rather evident that there is considerable variation in the sample proportions of the second figure. However, we would want to know how much of this is actually due to varying probability of presence vs. pure chance. Note also, that there are no sampling sites in Kalajoki (sampling area number 7 below) with vegetation cover. Hence, we have missing data there.

veg <- y.veg[19:36]  
veg <- append(veg, 0, 6) #adding missing data, copying closest geographical location  
noveg <- y.noveg[20:38]  
y2 <- aperm(array(c(veg, noveg), dim=c(19,2)))

N.veg <- append(N.veg, 1,6)  
Ns <- aperm(array(c(N.veg, N.noveg), dim=c(19,2)))

We will denote by the probability that white fish larvae are present in area at sites with () or without () bottom vegetation. The data will be denoted by and where the former denotes the number of sites with white fish larvae and the latter the total number of sites inside an area with () or without () bottom vegetation. We will now implement the following model where is the prior mean of and governs the uncertainty about it. The parametrization of log-Gaussian distribution is such that and

### Task 1

fish\_model = "  
data {  
 int<lower = 0> dim;  
 int<lower = 0> n;  
 int y[dim, n];  
 int total[dim, n];  
}  
  
parameters {  
 real<lower=0, upper=1> theta[dim, n];  
 real<lower=0.01, upper=0.99> mu[dim];  
 real<lower=0.01> s[dim];  
}  
  
model {  
 for (d in 1:dim){  
 s[dim] ~ lognormal(4, 2);  
 for (site in 1:n){   
 theta[d, site] ~ beta(mu[d]\*s[d], s[d] - mu[d]\*s[d]);  
 y[d, site] ~ binomial(total[d, site], theta[d, site]);  
 }  
 }  
}  
"

dataset <- list("dim"=2, "n"=19, "y"=y2, "total"=Ns)  
post=stan(model\_code=fish\_model,data=dataset,warmup=500,iter=1000,chains=3,thin=1)

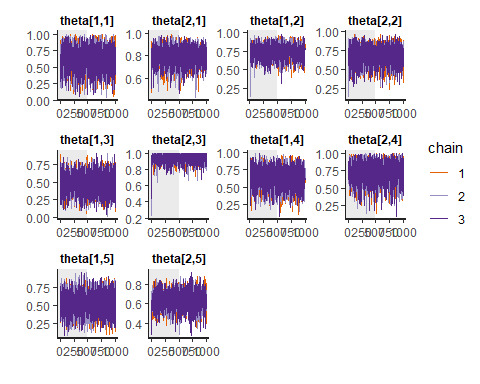
##   
## SAMPLING FOR MODEL 'anon\_model' NOW (CHAIN 1).  
## Chain 1:   
## Chain 1: Gradient evaluation took 0.000108 seconds  
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 1.08 seconds.  
## Chain 1: Adjust your expectations accordingly!  
## Chain 1:   
## Chain 1:   
## Chain 1: Iteration: 1 / 1000 [ 0%] (Warmup)  
## Chain 1: Iteration: 100 / 1000 [ 10%] (Warmup)  
## Chain 1: Iteration: 200 / 1000 [ 20%] (Warmup)  
## Chain 1: Iteration: 300 / 1000 [ 30%] (Warmup)  
## Chain 1: Iteration: 400 / 1000 [ 40%] (Warmup)  
## Chain 1: Iteration: 500 / 1000 [ 50%] (Warmup)  
## Chain 1: Iteration: 501 / 1000 [ 50%] (Sampling)  
## Chain 1: Iteration: 600 / 1000 [ 60%] (Sampling)  
## Chain 1: Iteration: 700 / 1000 [ 70%] (Sampling)  
## Chain 1: Iteration: 800 / 1000 [ 80%] (Sampling)  
## Chain 1: Iteration: 900 / 1000 [ 90%] (Sampling)  
## Chain 1: Iteration: 1000 / 1000 [100%] (Sampling)  
## Chain 1:   
## Chain 1: Elapsed Time: 0.496 seconds (Warm-up)  
## Chain 1: 0.45 seconds (Sampling)  
## Chain 1: 0.946 seconds (Total)  
## Chain 1:   
##   
## SAMPLING FOR MODEL 'anon\_model' NOW (CHAIN 2).  
## Chain 2:   
## Chain 2: Gradient evaluation took 5.4e-05 seconds  
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.54 seconds.  
## Chain 2: Adjust your expectations accordingly!  
## Chain 2:   
## Chain 2:   
## Chain 2: Iteration: 1 / 1000 [ 0%] (Warmup)  
## Chain 2: Iteration: 100 / 1000 [ 10%] (Warmup)  
## Chain 2: Iteration: 200 / 1000 [ 20%] (Warmup)  
## Chain 2: Iteration: 300 / 1000 [ 30%] (Warmup)  
## Chain 2: Iteration: 400 / 1000 [ 40%] (Warmup)  
## Chain 2: Iteration: 500 / 1000 [ 50%] (Warmup)  
## Chain 2: Iteration: 501 / 1000 [ 50%] (Sampling)  
## Chain 2: Iteration: 600 / 1000 [ 60%] (Sampling)  
## Chain 2: Iteration: 700 / 1000 [ 70%] (Sampling)  
## Chain 2: Iteration: 800 / 1000 [ 80%] (Sampling)  
## Chain 2: Iteration: 900 / 1000 [ 90%] (Sampling)  
## Chain 2: Iteration: 1000 / 1000 [100%] (Sampling)  
## Chain 2:   
## Chain 2: Elapsed Time: 1.574 seconds (Warm-up)  
## Chain 2: 0.653 seconds (Sampling)  
## Chain 2: 2.227 seconds (Total)  
## Chain 2:   
##   
## SAMPLING FOR MODEL 'anon\_model' NOW (CHAIN 3).  
## Chain 3:   
## Chain 3: Gradient evaluation took 6.1e-05 seconds  
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.61 seconds.  
## Chain 3: Adjust your expectations accordingly!  
## Chain 3:   
## Chain 3:   
## Chain 3: Iteration: 1 / 1000 [ 0%] (Warmup)  
## Chain 3: Iteration: 100 / 1000 [ 10%] (Warmup)  
## Chain 3: Iteration: 200 / 1000 [ 20%] (Warmup)  
## Chain 3: Iteration: 300 / 1000 [ 30%] (Warmup)  
## Chain 3: Iteration: 400 / 1000 [ 40%] (Warmup)  
## Chain 3: Iteration: 500 / 1000 [ 50%] (Warmup)  
## Chain 3: Iteration: 501 / 1000 [ 50%] (Sampling)  
## Chain 3: Iteration: 600 / 1000 [ 60%] (Sampling)  
## Chain 3: Iteration: 700 / 1000 [ 70%] (Sampling)  
## Chain 3: Iteration: 800 / 1000 [ 80%] (Sampling)  
## Chain 3: Iteration: 900 / 1000 [ 90%] (Sampling)  
## Chain 3: Iteration: 1000 / 1000 [100%] (Sampling)  
## Chain 3:   
## Chain 3: Elapsed Time: 0.625 seconds (Warm-up)  
## Chain 3: 0.652 seconds (Sampling)  
## Chain 3: 1.277 seconds (Total)  
## Chain 3:

print(post)

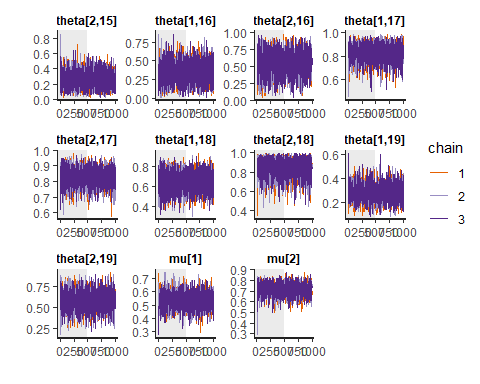
## Inference for Stan model: anon\_model.  
## 3 chains, each with iter=1000; warmup=500; thin=1;   
## post-warmup draws per chain=500, total post-warmup draws=1500.  
##   
## mean se\_mean sd 2.5% 25% 50% 75% 97.5% n\_eff  
## theta[1,1] 0.62 0.00 0.19 0.24 0.49 0.63 0.76 0.95 2178  
## theta[1,2] 0.74 0.00 0.09 0.54 0.68 0.75 0.81 0.91 1830  
## theta[1,3] 0.47 0.00 0.15 0.19 0.37 0.48 0.58 0.76 2368  
## theta[1,4] 0.63 0.00 0.16 0.30 0.52 0.63 0.75 0.90 2379  
## theta[1,5] 0.52 0.00 0.16 0.21 0.41 0.52 0.64 0.82 2772  
## theta[1,6] 0.38 0.00 0.19 0.06 0.24 0.39 0.52 0.74 1707  
## theta[1,7] 0.45 0.00 0.20 0.09 0.31 0.45 0.60 0.83 1980  
## theta[1,8] 0.44 0.00 0.19 0.08 0.29 0.45 0.59 0.80 2036  
## theta[1,9] 0.29 0.00 0.11 0.09 0.20 0.28 0.37 0.51 1656  
## theta[1,10] 0.73 0.00 0.10 0.52 0.67 0.74 0.81 0.91 1984  
## theta[1,11] 0.48 0.00 0.14 0.22 0.38 0.48 0.58 0.74 2244  
## theta[1,12] 0.64 0.00 0.15 0.33 0.53 0.65 0.74 0.90 2115  
## theta[1,13] 0.73 0.00 0.10 0.52 0.66 0.73 0.80 0.92 2120  
## theta[1,14] 0.72 0.00 0.08 0.55 0.66 0.72 0.78 0.87 2390  
## theta[1,15] 0.34 0.00 0.07 0.20 0.29 0.34 0.39 0.48 2066  
## theta[1,16] 0.34 0.00 0.15 0.08 0.24 0.33 0.45 0.63 1956  
## theta[1,17] 0.83 0.00 0.08 0.66 0.78 0.84 0.89 0.95 1427  
## theta[1,18] 0.65 0.00 0.10 0.46 0.58 0.66 0.72 0.83 2415  
## theta[1,19] 0.28 0.00 0.07 0.15 0.23 0.28 0.33 0.42 2632  
## theta[2,1] 0.80 0.00 0.08 0.62 0.75 0.82 0.86 0.93 2718  
## theta[2,2] 0.68 0.00 0.13 0.41 0.60 0.70 0.78 0.89 2602  
## theta[2,3] 0.94 0.00 0.05 0.81 0.92 0.96 0.98 1.00 1421  
## theta[2,4] 0.74 0.00 0.15 0.38 0.64 0.75 0.85 0.96 2166  
## theta[2,5] 0.63 0.00 0.10 0.43 0.57 0.64 0.70 0.83 2365  
## theta[2,6] 0.53 0.00 0.12 0.27 0.45 0.53 0.61 0.76 2433  
## theta[2,7] 0.88 0.00 0.10 0.62 0.83 0.91 0.96 1.00 1371  
## theta[2,8] 0.85 0.00 0.07 0.69 0.81 0.86 0.91 0.97 2356  
## theta[2,9] 0.49 0.00 0.17 0.15 0.36 0.49 0.61 0.80 1614  
## theta[2,10] 0.81 0.00 0.12 0.51 0.74 0.82 0.90 0.97 2760  
## theta[2,11] 0.88 0.00 0.08 0.68 0.83 0.89 0.94 0.98 2200  
## theta[2,12] 0.79 0.00 0.12 0.49 0.72 0.81 0.87 0.97 1621  
## theta[2,13] 0.91 0.00 0.08 0.70 0.87 0.93 0.97 1.00 1535  
## theta[2,14] 0.87 0.00 0.04 0.77 0.84 0.87 0.90 0.93 3027  
## theta[2,15] 0.25 0.00 0.11 0.07 0.17 0.24 0.31 0.48 1651  
## theta[2,16] 0.56 0.00 0.17 0.20 0.44 0.56 0.68 0.87 2296  
## theta[2,17] 0.83 0.00 0.06 0.71 0.80 0.84 0.88 0.93 2107  
## theta[2,18] 0.85 0.00 0.10 0.61 0.79 0.87 0.92 0.98 2300  
## theta[2,19] 0.58 0.00 0.12 0.35 0.50 0.59 0.67 0.79 2344  
## mu[1] 0.54 0.00 0.06 0.41 0.50 0.54 0.58 0.65 1616  
## mu[2] 0.72 0.00 0.06 0.59 0.68 0.72 0.76 0.82 1839  
## s[1] 6.00 0.14 3.36 2.15 3.69 5.19 7.41 14.72 571  
## s[2] 4.33 0.07 2.13 1.59 2.88 3.92 5.27 9.32 809  
## lp\_\_ -319.50 0.30 6.02 -332.45 -323.37 -319.01 -315.33 -308.93 392  
## Rhat  
## theta[1,1] 1  
## theta[1,2] 1  
## theta[1,3] 1  
## theta[1,4] 1  
## theta[1,5] 1  
## theta[1,6] 1  
## theta[1,7] 1  
## theta[1,8] 1  
## theta[1,9] 1  
## theta[1,10] 1  
## theta[1,11] 1  
## theta[1,12] 1  
## theta[1,13] 1  
## theta[1,14] 1  
## theta[1,15] 1  
## theta[1,16] 1  
## theta[1,17] 1  
## theta[1,18] 1  
## theta[1,19] 1  
## theta[2,1] 1  
## theta[2,2] 1  
## theta[2,3] 1  
## theta[2,4] 1  
## theta[2,5] 1  
## theta[2,6] 1  
## theta[2,7] 1  
## theta[2,8] 1  
## theta[2,9] 1  
## theta[2,10] 1  
## theta[2,11] 1  
## theta[2,12] 1  
## theta[2,13] 1  
## theta[2,14] 1  
## theta[2,15] 1  
## theta[2,16] 1  
## theta[2,17] 1  
## theta[2,18] 1  
## theta[2,19] 1  
## mu[1] 1  
## mu[2] 1  
## s[1] 1  
## s[2] 1  
## lp\_\_ 1  
##   
## Samples were drawn using NUTS(diag\_e) at Mon Dec 19 11:24:20 2022.  
## For each parameter, n\_eff is a crude measure of effective sample size,  
## and Rhat is the potential scale reduction factor on split chains (at   
## convergence, Rhat=1).

The rhat-values are looking splendid all the way, only 1s.

plot(post, plotfun= "trace", pars=colnames(as.matrix(post))[1:10], inc\_warmup = TRUE)

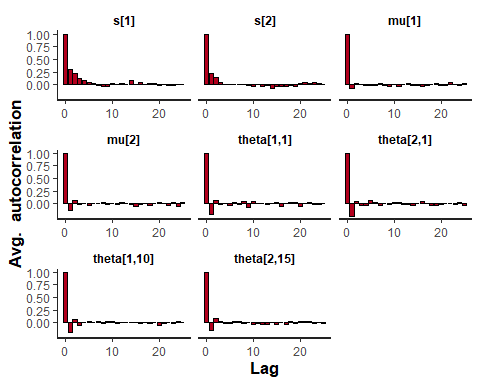


plot(post, plotfun= "trace", pars=colnames(as.matrix(post))[30:40], inc\_warmup = TRUE)

 (…)

This was just a sample of the parameters, but also when plotting the convergences looks sound.

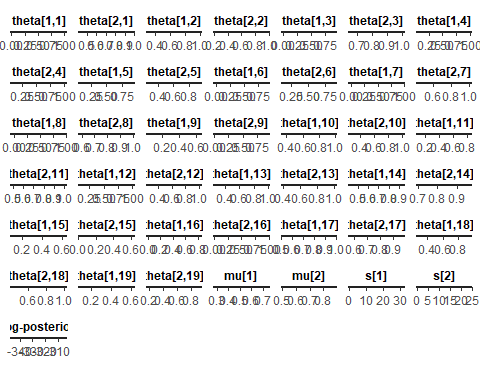
stan\_ac(post,c("s[1]", "s[2]", "mu[1]", "mu[2]", "theta[1,1]", "theta[2,1]", "theta[1,10]", "theta[2,15]"),inc\_warmup = FALSE, lags = 25)

 (…)

Also, the autocorrelation is, i.e., close to zero at higher lags.

Nsamp=as.matrix(post)

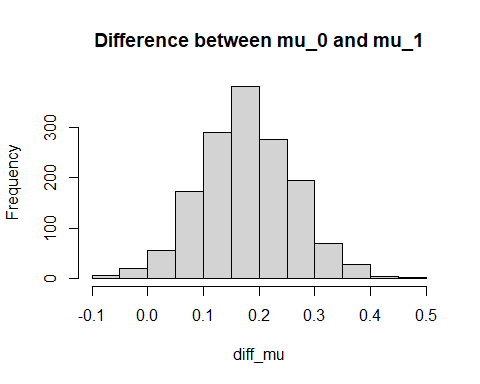
plot(post, plotfun = "hist", pars = colnames(Nsamp) ,bins=50)

 (…)

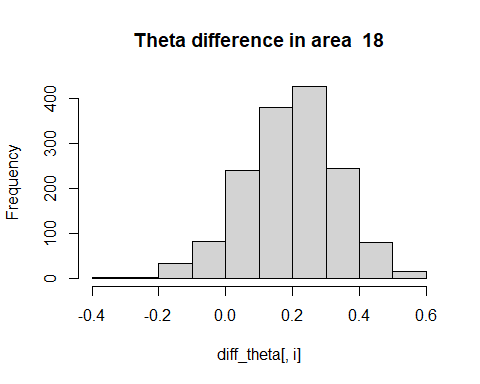
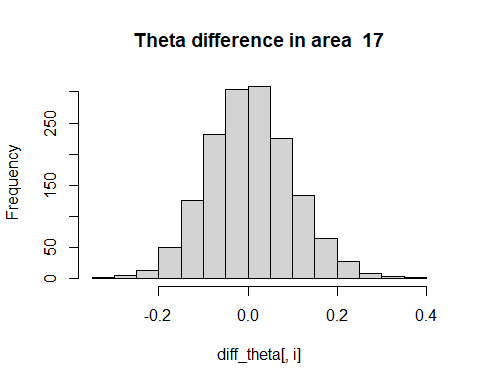
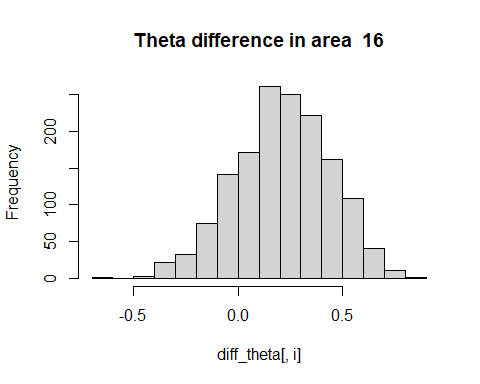
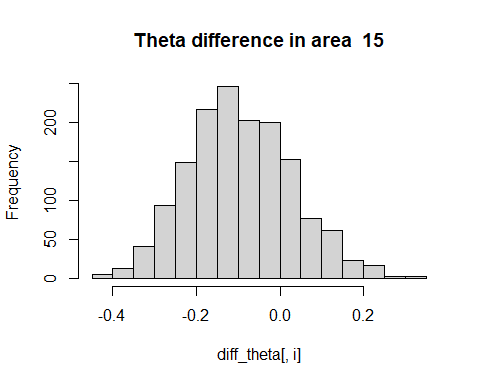
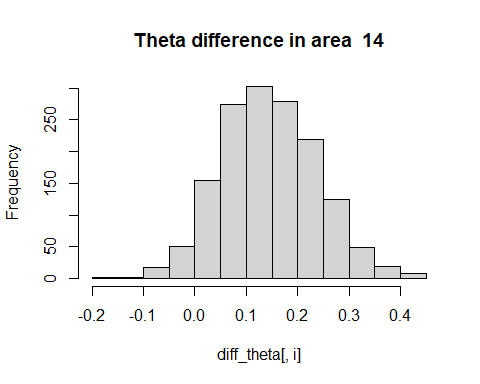
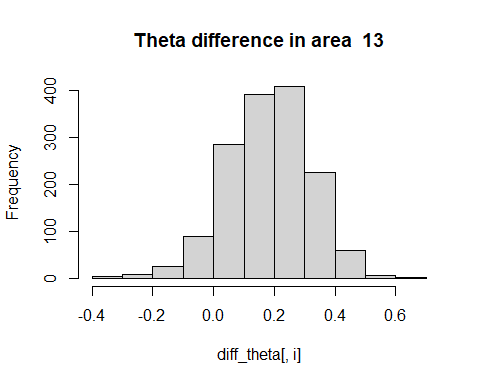
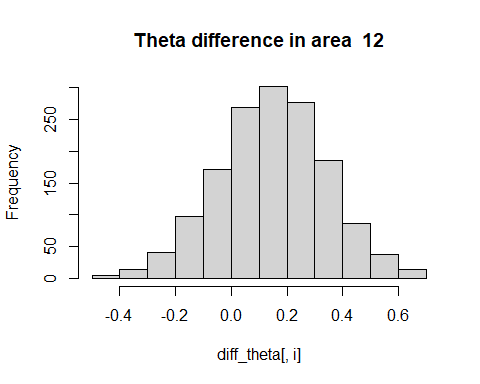
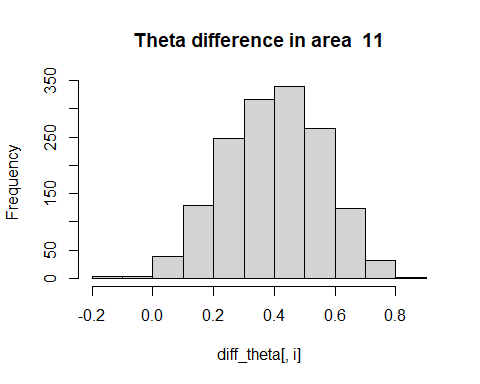
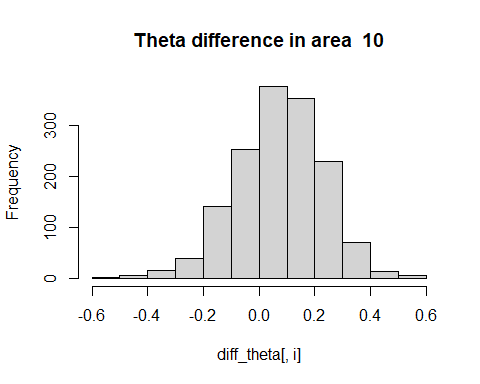
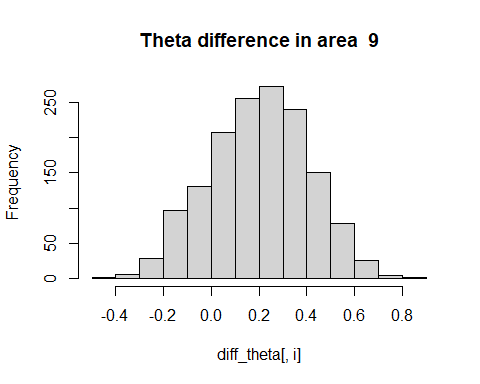
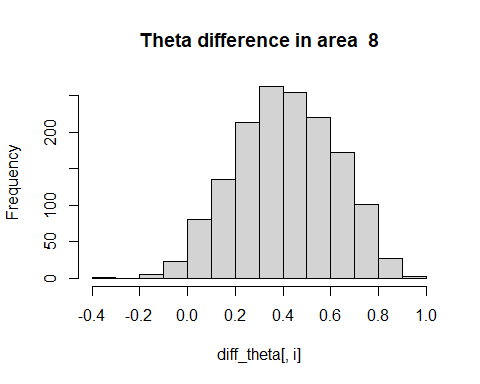
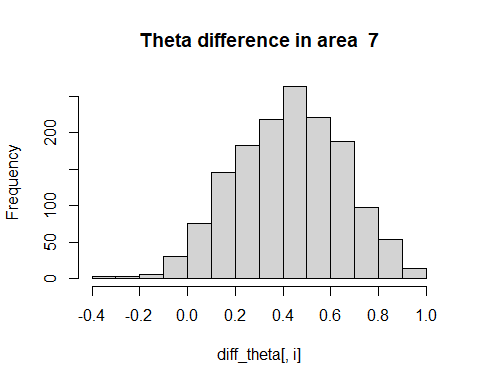
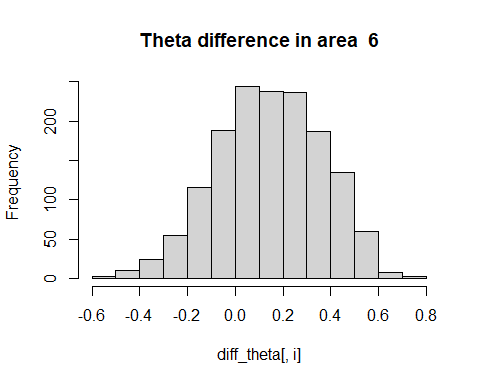
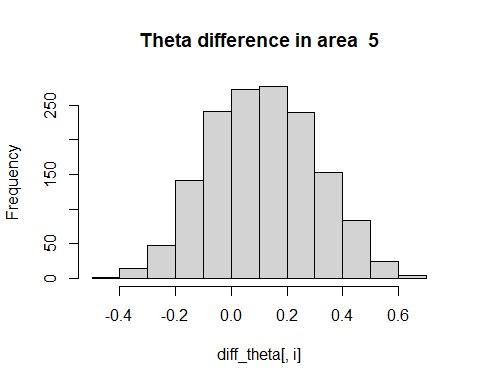
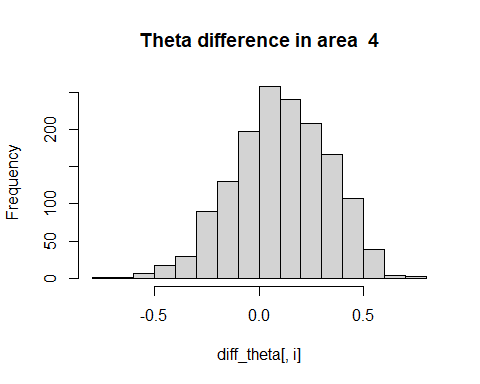
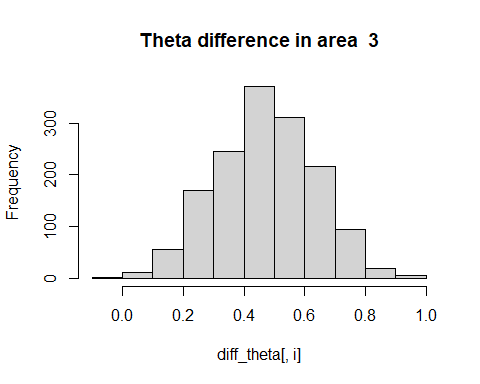
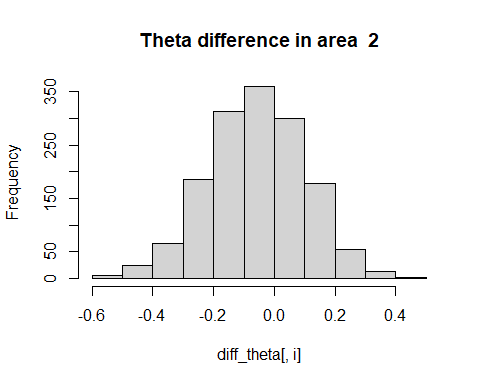
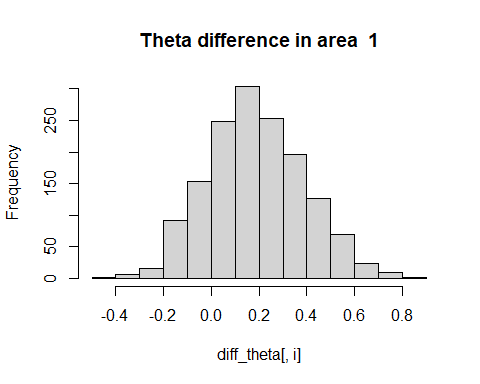
The plots are quite small, but the general idea can be sensed.

### Task 2

diff\_mu = Nsamp[,"mu[2]"] - Nsamp[,"mu[1]"]  
diff\_theta <- matrix(c(0), nrow = 1500, ncol = 18)  
ind = 1  
for (i in 1:36){  
 if (i %% 2 > 0){  
 diff\_theta[,ind] = Nsamp[,i+1] - Nsamp[,i]  
 ind = ind +1  
 } else {  
 next  
 }  
}  
hist(diff\_mu, main="Difference between mu\_0 and mu\_1")

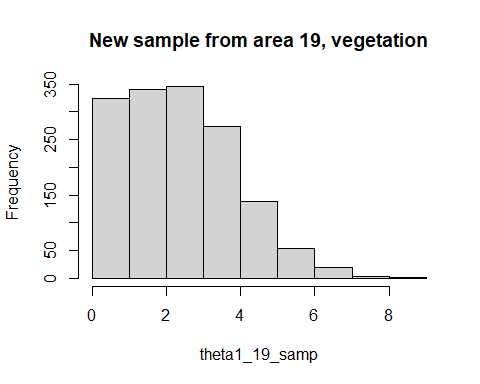


for (i in 1:18){  
 hist(diff\_theta[,i], main = paste("Theta difference in area " , i))  
}

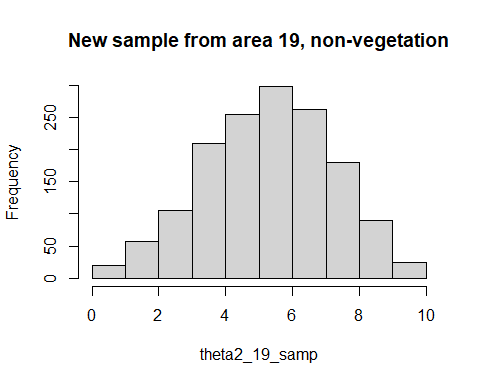


### Task 3

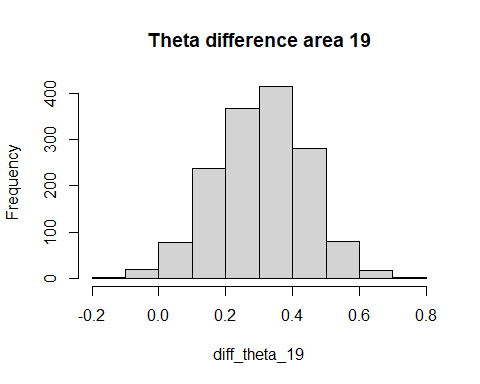
theta1\_19\_samp <- rbinom(Nsamp[,"theta[1,19]"], 10, Nsamp[,"theta[1,19]"])  
theta2\_19\_samp <- rbinom(Nsamp[,"theta[2,19]"], 10, Nsamp[,"theta[2,19]"])  
hist(theta1\_19\_samp, main="New sample from area 19, vegetation")



hist(theta2\_19\_samp, main="New sample from area 19, non-vegetation")

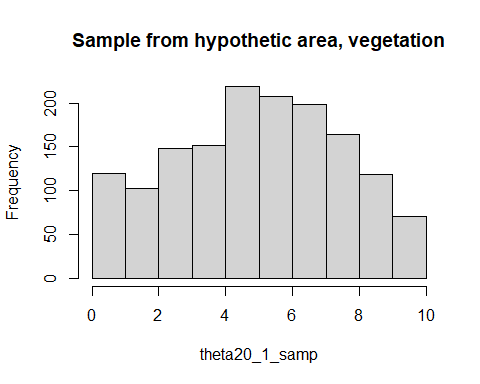


diff\_theta\_19 <- Nsamp[,"theta[2,19]"] - Nsamp[, "theta[1,19]"]  
hist(diff\_theta\_19, main="Theta difference area 19")

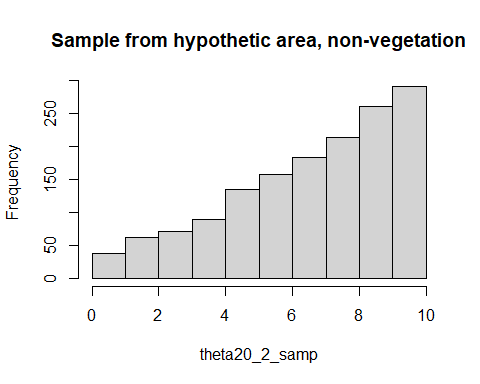
 (…)

### Task 4

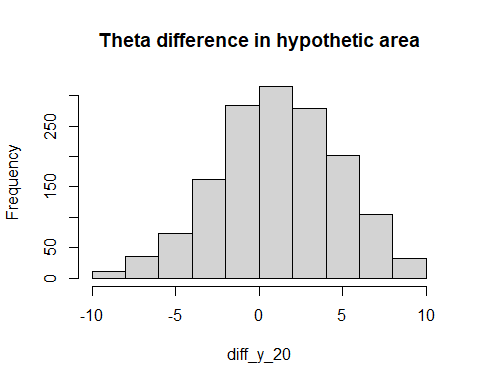
y\_tilde\_1 <- rbeta(1500, Nsamp[,"mu[1]"]\*Nsamp[,"s[1]"], Nsamp[,"s[1]"] - Nsamp[,"mu[1]"]\*Nsamp[,"s[1]"])  
theta20\_1\_samp <- rbinom(y\_tilde\_1, 10, y\_tilde\_1)  
hist(theta20\_1\_samp, breaks = 10, main="Sample from hypothetic area, vegetation")



y\_tilde\_2 <- rbeta(1500, Nsamp[,"mu[2]"]\*Nsamp[,"s[2]"], Nsamp[,"s[2]"] - Nsamp[,"mu[2]"]\*Nsamp[,"s[2]"])  
theta20\_2\_samp <- rbinom(y\_tilde\_2, 10, y\_tilde\_2)  
hist(theta20\_2\_samp, breaks = 10, main="Sample from hypothetic area, non-vegetation")



diff\_y\_20 = theta20\_2\_samp - theta20\_1\_samp  
hist(diff\_y\_20, main="Theta difference in hypothetic area")

 (…)

### Task 5

#mean for mu:  
mean(Nsamp[,'mu[1]'])

## [1] 0.5386341

mean(Nsamp[, 'mu[2]'])

## [1] 0.7170965

sd(Nsamp[,'mu[1]'])

## [1] 0.06024598

sd(Nsamp[, 'mu[2]'])

## [1] 0.0556165

If we compare the attained here to what we saw in the in week 2 exercise 3, we see that the results are quite close to each other, with being a bit lower (in week 2 i calculated a . The standard deviation is approximately double as big, perhaps to allow for the bigger variation within the individual sample sites.

Also, the logic of these parameters is somewhat different. The mean estimated in the hierarchical model tries to capture a true mean of white fish larvae in general, regardless of location within the Gulf of Bothnia. While the individual site-specific show variation that possibly is not exclusively a measure of variation within an identical distribution, but also a sign of a de facto difference in the distribution at different sites.

The Gulf of Bothnia is not of a negligible size, with a length of 725 km. The coastal line is quite different, with mountains in parts of the west side of the gulf and flatlands in other places. Perhaps it is a good idea to assume that the true distribution might differ between different sites within this area, making a hierarchical model somewhat more useful for modelling the prevalence of white fish larvae.

# References

Lari Veneranta, Richard Hudd and Jarno Vanhatalo (2013). Reproduction areas of sea-spawning Coregonids reflect the environment in shallow coastal waters. Marine Ecology Progress Series, 477:231-250. <http://www.int-res.com/abstracts/meps/v477/p231-250/>