PROBLEM 15

SHOW THAT $E[\hat{g}(\vec{x})] = E[g(x)]$

DEMONSTRATION: $E[\hat{g}(\vec{x})] = E[\frac{1}{N}\sum_i^N g(x_i)] \overset{\text{LINEARITY OF EXPECTATION}}{=}$

$$= \frac{1}{N} E[\sum_i^N g(x_i)] \overset{-\,''\,-}{=} \frac{1}{N}\sum_i^N E[g(x_i)] \overset{\substack{X_i \text{ ARE FROM} \\ \text{THE SAME DISTRIBUTION}}}{=}$$

$$= \frac{1}{N}\sum_i^N E[g(x)] = \frac{1}{\cancel{N}} \cdot \cancel{N} E[g(x)] = E[g(x)] \quad \square$$

SHOW THAT $Var[\hat{g}(\vec{x})] = \frac{1}{N} Var(g(x))$

DEMONSTRATION: $Var[\hat{g}(\vec{x})] = Var[\frac{1}{N}\sum_i^N g(x_i)] =$

$$= \frac{1}{N^2} Var[\sum_i^N g(x_i)] = \frac{1}{N^2}\sum_i^N\sum_j^N \Sigma_{ij} \quad (*)$$

Σ - COVARIANCE MATRIX OF $\forall g(x_i)$ — VARIANCE OF $g(x_i)$ IS ON THE i^{th} ELEMENT OF DIAGONAL. OTHER ELEMENTS OF THE MATRIX ARE $Cov(g(x_i), g(x_j))$. SINCE $g(x_i)$ & $g(x_j)$ ARE INDEPENDENT, THEY ARE ZEROS.

$(*)$ CAN BE REWRITTEN AS

$$= \frac{1}{N^2}\left(\sum_i^N Var[g(x_i)]\right) = \frac{1}{N^2}\sum_i^N Var[g(x)] =$$

$$= \frac{1}{N^{\cancel{2}}} \cancel{N} Var[g(x)] = \frac{1}{N} Var[g(x)] \quad \square$$

THESE RESULTS ARE NOT APPLIED TO THE CASE WHEN X_i ARE GENERATED FROM MCMC SAMPLER BECAUSE X_i DEPEND ON X_{i-1} (IF $i > 0$).

PROBLEM 16

a) $\underline{P(C=T \mid R=T, S=T, W=T)} = \dfrac{P(R=T, S=T, W=T \mid C=T)}{P(R=T, S=T, W=T)}$

- $P(R=T, S=T, W=T) = \sum_c P(R=T, S=T, W=T \mid C=c) =$

$= \sum_c P(R=T, S=T \mid C=c) = \sum_c P(R=T \mid C=c) P(S=T \mid C=c) P(C=c) =$

$= P(R=T \mid C=T) P(S=T \mid C=T) P(C=T) + P(R=T \mid C=F) P(S=T \mid C=F) P(C=F) =$

$= \frac{\overset{4}{\cancel{8}}}{10} \cdot \frac{1}{10} \cdot \frac{1}{\cancel{2}} + \frac{5}{10} \cdot \frac{\cancel{2}}{10} \cdot \frac{1}{\cancel{2}} = \frac{4}{100} + \frac{5}{100} = \frac{9}{100}$

- $P(R=T, S=T, W=T \mid C=T) = P(R=T, S=T \mid C=T) =$

$= P(R=T \mid C=T) P(S=T \mid C=T) P(C=T) = \frac{\overset{4}{\cancel{8}}}{10} \cdot \frac{1}{10} \cdot \frac{1}{\cancel{2}} = \frac{4}{100}$

- $P(C=T \mid R=T, S=T, W=T) = \frac{4}{100} \cdot \frac{100}{9} = \underline{\frac{4}{9} \approx 0.44}$

$\underline{P(C=T \mid R=F, S=T, W=T)} = \dfrac{P(R=F, S=T, W=T \mid C=T)}{P(R=F, S=T, W=T)}$

- $P(R=F, S=T, W=T) = \sum_c P(R=F, S=T \mid C=c) =$

$= P(R=F \mid C=T) P(S=T \mid C=T) P(C=T) + P(R=F \mid C=F) P(S=T \mid C=F) P(C=F) =$

$= \frac{\cancel{2}}{10} \cdot \frac{1}{10} \cdot \frac{1}{\cancel{2}} + \frac{\overset{4}{\cancel{8}}}{10} \cdot \frac{5}{10} \cdot \frac{1}{\cancel{2}} = \frac{1}{100} + \frac{20}{100} = \frac{21}{100}$

- $P(R=F, S=T, W=T \mid C=T) = P(R=F, S=T \mid C=T) =$

$= P(R=F \mid C=T) P(S=T \mid C=T) P(C=T) = \frac{1}{100}$

- $P(C=T \mid R=F, S=T, W=T) = \frac{1}{\cancel{100}} \cdot \frac{\cancel{100}}{21} = \underline{\frac{1}{21} \approx 0.48}$

$$\underline{P(R=T \mid C=T, S=T, W=T)} = \frac{P(R=T, C=T, S=T, W=T)}{P(C=T, S=T, W=T)}$$

- $P(C=T, S=T, W=T) = P(W=T \mid C=T, S=T) =$

$$= \sum_r P(W=T \mid R=r, S=T)\, P(R=r \mid C=T) =$$

$$= P(W=T \mid R=T, S=T)\, P(R=T \mid C=T) + P(W=T \mid R=F, S=T)\, P(R=F \mid C=T) =$$

$$= \frac{99}{100} \cdot \frac{8}{10} + \frac{90}{100} \cdot \frac{2}{10} = \frac{792}{1000} + \frac{180}{1000} = \frac{972}{1000}$$

- $P(R=T, C=T, S=T, W=T) = P(W=T \mid R=T, C=T, S=T)\, P(R=T \mid C=T, S=T) =$

$$= P(W=T \mid R=T, S=T)\, P(R=T \mid C=T) =$$

$$= \frac{99}{100} \cdot \frac{8}{10} = \frac{792}{1000}$$

- $P(R=T \mid C=T, S=T, W=T) = \frac{792}{\cancel{1000}} \cdot \frac{\cancel{1000}}{972} = \underline{\frac{792}{972} \approx 0.82}$

$$\underline{P(R=T \mid C=F, S=T, W=T)} = \frac{P(R=T, C=F, S=T, W=T)}{P(C=F, S=T, W=T)}$$

- $P(C=F, S=T, W=T) = P(W=T \mid C=F, S=T) =$

$$= \sum_r P(W=T \mid R=r, S=T)\, P(R=r \mid C=F) =$$

$$= P(W=T \mid R=T, S=T)\, P(R=T \mid C=F) + P(W=T \mid R=F, S=T)\, P(R=F \mid C=F) =$$

$$= \frac{99}{100} \cdot \frac{2}{10} + \frac{90}{100} \cdot \frac{8}{10} = \frac{198}{1000} + \frac{720}{1000} = \frac{918}{1000}$$

- $P(R=T, C=F, S=T, W=T) = P(W=T \mid R=T, C=F, S=T)\, P(R=T \mid C=F, S=T) =$

$$= P(W=T \mid R=T, S=T)\, P(R=T \mid C=F) =$$

$$= \frac{99}{100} \cdot \frac{2}{10} = \frac{198}{1000}$$

- $P(R=T \mid C=F, S=T, W=T) = \frac{198}{\cancel{1000}} \cdot \frac{\cancel{1000}}{918} = \underline{\frac{198}{918} \approx 0.22}$

i) ANALYTICAL COMPUTATION OF MARGINAL PROBABILITY

$$P(R=T \mid S=T, W=T) = \frac{P(S=T, W=T, R=T)}{P(S=T, W=T)}$$

- $P(S=T, W=T) = \sum_c \sum_r P(S=T, W=T, C=c, R=r) =$

$$= \sum_c \sum_r P(W=T \mid S=T, R=r) P(S=T \mid C=c) P(R=r \mid C=c) P(C=c)$$

0	0	0	0	0
0	1	0	1	1
1	0	1	0	0
1	1	1	1	1

- $P(S=T, W=T, R=T) = \sum_c P(S=T, W=T, R=T, C=c) =$

$$= \sum_c P(W=T \mid S=T, R=T) P(S=T \mid C=c) P(R=T \mid C=c) P(C=c)$$

Project 6

Contents

Problem 16

Gibbs sampler (b)

```r
# Encoding conditional probabilities
p <- list(C=0.5, S.C1=0.1, S.C0=0.5, R.C1=0.8, R.C0=0.2,
          W.S1_R1=0.99, W.S1_R0=0.9, W.S0_R1=0.9, W.S0_R0=0.01)
p$C.R1_S1_W1 <- p$R.C1*p$S.C1*p$C/(p$R.C1*p$S.C1*p$C+p$R.C0*p$S.C0*(1-p$C))
p$C.R0_S1_W1 <- (1-p$R.C1)*p$S.C1*p$C/((1-p$R.C1)*p$S.C1*p$C+(1-p$R.C0)*p$S.C0*(1-p$C))
p$R.C1_S1_W1 <- p$W.S1_R1*p$R.C1/(p$W.S1_R1*p$R.C1+p$W.S1_R0*(1-p$R.C1))
p$R.C0_S1_W1 <- p$W.S1_R1*p$R.C0/(p$W.S1_R1*p$R.C0+p$W.S1_R0*(1-p$R.C0))
```

```r
Gibbs_sampler <- function(N=100) {
    samples <- matrix(NA, ncol=2, nrow=N, dimnames=list(NULL, c("R", "C")))
    # Initialisation (beginning state)
    samples[1,] <- c(TRUE, TRUE)

    for(n in 2:N) {
        # Random sampling to choose which variable (R or C) is updated
        indicator <- sample(c("R", "C"), size=1, prob=c(0.5, 0.5))
        if(indicator == "R") {
            # Sampling from P(R_{n+1}|C_{n},S=T,W=T)
            if(samples[(n-1), "C"]) sampling_prob <- p$R.C1_S1_W1 else sampling_prob <- p$R.C0_S1_W1

            sampled_R <- sample(c(TRUE, FALSE), size=1, prob=c(sampling_prob, (1-sampling_prob)))
            samples[n,] <- c(sampled_R, samples[(n-1), "C"])
        } else {
            # Sampling from P(C_{n+1}|R_{n},S=T,W=T)
            if(samples[(n-1), "R"]) sampling_prob <- p$C.R1_S1_W1 else sampling_prob <- p$C.R0_S1_W1

            sampled_C <- sample(c(TRUE, FALSE), size=1, prob=c(sampling_prob, (1-sampling_prob)))
            samples[n,] <- c(samples[(n-1), "R"], sampled_C)
        }
```

```r
    }

    return(samples)
}
```

```r
set.seed(42)
N <- 100
samples <- Gibbs_sampler(N)
```

```r
result <- table(samples[, "R"], samples[, "C"])/N
rownames(result) <- c("R0", "R1")
colnames(result) <- c("C0", "C1")
result
```

```
##
##        C0   C1
##   R0 0.59 0.05
##   R1 0.18 0.18
```

Marginal probability (c)

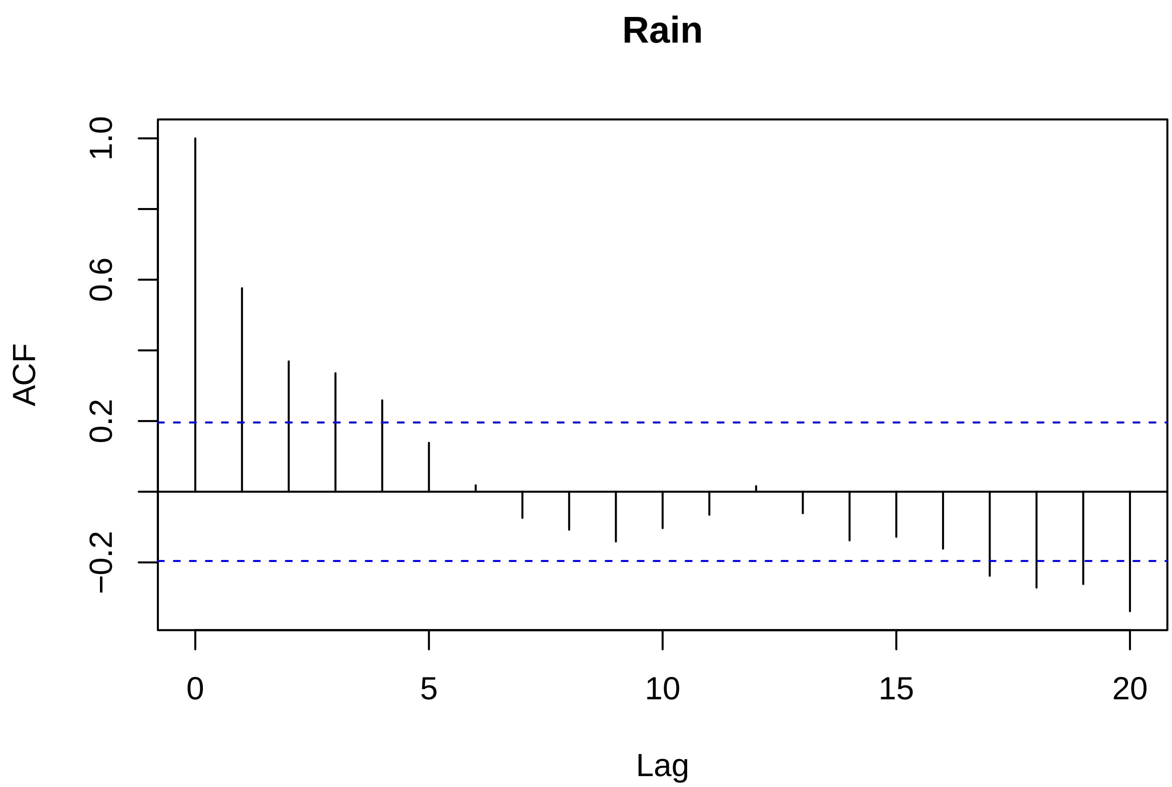
Computing marginal probability P(R=T|S=T,W=T).

```r
marginal.R.S1_W1 <- sum(result[2,])
marginal.R.S1_W1
```

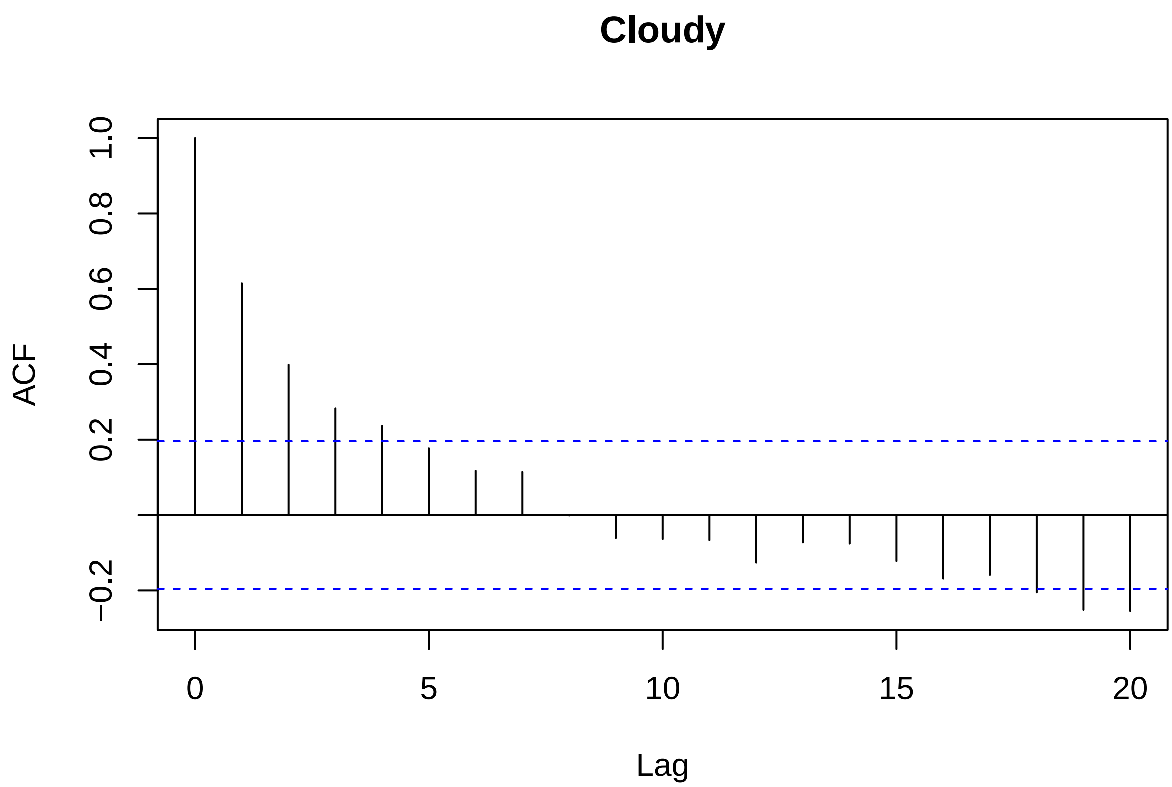
```
## [1] 0.36
```

Auto-correlation and effective sample size (ESS) (d)

```r
acf_output_R <- acf(as.numeric(samples[, "R"]), main="Rain")
```




```
acf_output_C <- acf(as.numeric(samples[, "C"]), main="Cloudy")
```




```
ESS_R <- acf_output_R$n.used/(1+2*sum(acf_output_R$acf))
ESS_C <- acf_output_C$n.used/(1+2*sum(acf_output_C$acf))
print(paste("Effective sample size for 'rain':", ESS_R))
```

```
## [1] "Effective sample size for 'rain': 44.4718962322423"
```

```
print(paste("Effective sample size for 'cloudy':", ESS_C))
```

```
## [1] "Effective sample size for 'cloudy': 27.5547672392332"
```

Gibbs sampling of 50000 samples (e)

```
set.seed(42)
N <- 50000
samples1 <- Gibbs_sampler(N)

result1 <- table(samples1[, "R"], samples1[, "C"])/N
rownames(result1) <- c("R0", "R1")
colnames(result1) <- c("C0", "C1")

samples2 <- Gibbs_sampler(N)

result2 <- table(samples2[, "R"], samples2[, "C"])/N
rownames(result2) <- c("R0", "R1")
colnames(result2) <- c("C0", "C1")
```

```r
result1
```

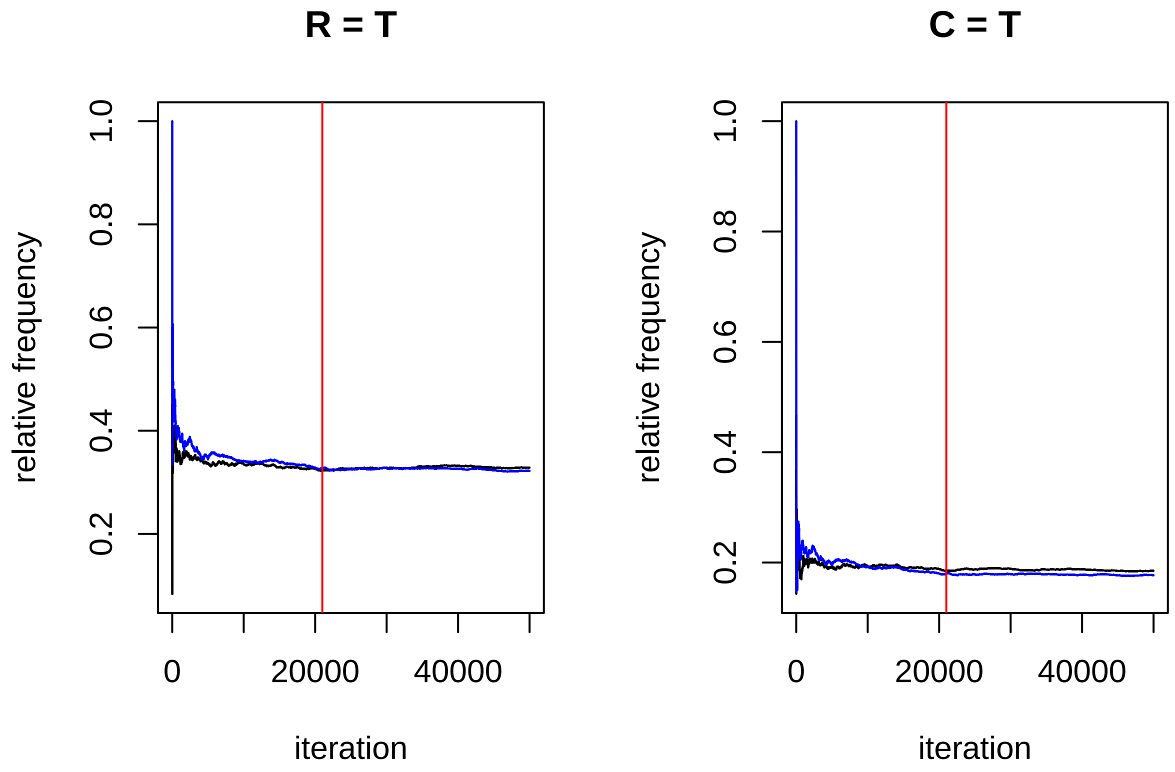
```
##
##          C0      C1
##   R0 0.63588 0.03570
##   R1 0.17904 0.14938
```

```r
result2
```

```
##
##          C0      C1
##   R0 0.64450 0.03334
##   R1 0.17826 0.14390
```

Relative frequencies to detect burn-in phase (f)

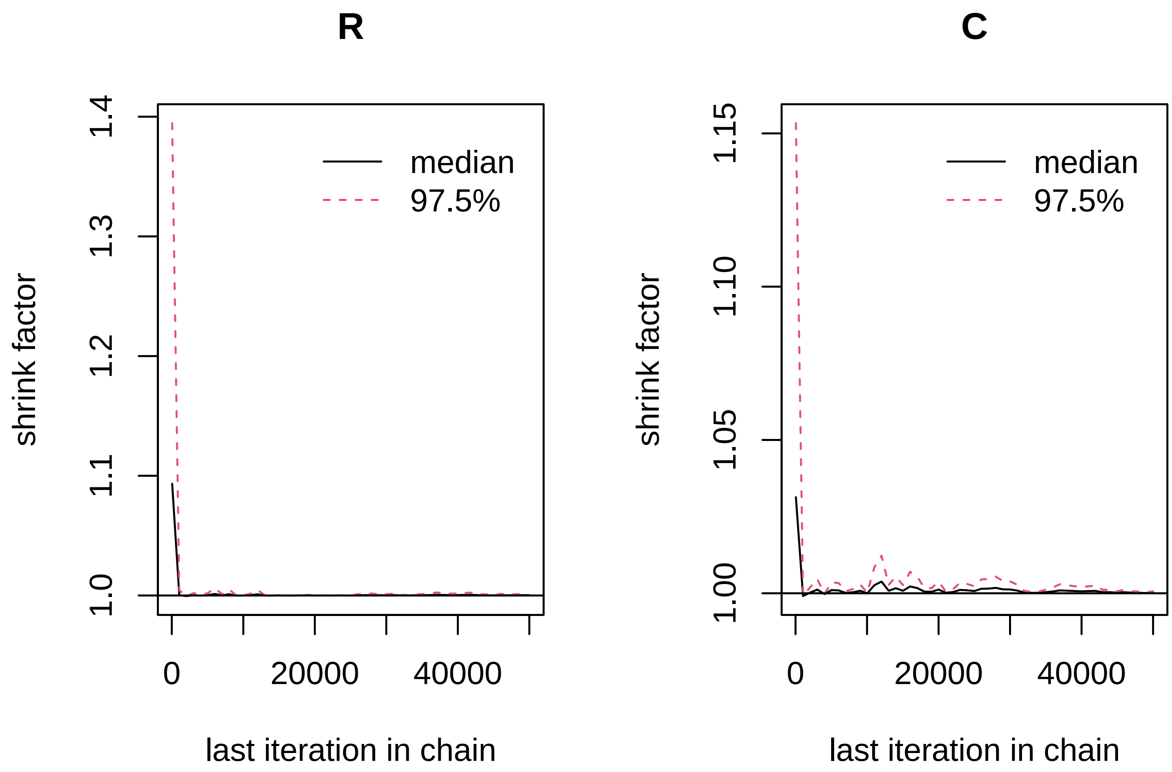
```r
# Plotting relative frequencies of R = T and C = T
burn_in_phase <- 21000
par(mfrow=c(1,2))
plot(1:N, cumsum(samples1[, "R"])/seq_len(N), type="l",
    xlab="iteration", ylab="relative frequency", main="R = T")
lines(cumsum(samples2[, "R"])/seq_len(N), type='l', col='blue')
abline(v=burn_in_phase, col="red")
plot(1:N, cumsum(samples1[, "C"])/seq_len(N), type="l",
    xlab="iteration", ylab="relative frequency", main="C = T")
lines(cumsum(samples2[, "C"])/seq_len(N), type='l', col='blue')
abline(v=burn_in_phase, col="red")
```



Based on this plot, burn-in time could be suggested to be 21000. After this time point curves of relative frequency for both runs and both variables seem to be converged.

Potential scale reduction factor (g)

```
library(coda)
mcmcs <- mcmc.list(mcmc(data=samples1, start=2, end=N),
                   mcmc(data=samples2, start=2, end=N))
gelman.plot(mcmcs)
```




```
gelman.diag(mcmcs)
```

```
## Potential scale reduction factors:
##
##   Point est. Upper C.I.
## R          1          1
## C          1          1
##
## Multivariate psrf
##
## 1
```

After around 20000 iterations, the fluctuations around the potential scale reduction factor are negligible for both variables, therefore a suggested burn-in time is 20000 iterations.

Re-estimation of marginal probability (h)

Computing marginal probability P(R=T|S=T,W=T) excluding the samples from the burn-in phase. The length of the burn-in phase was chosen to be the average between the burn-in phase estimate from relative frequency plot and plot of potential scale reduction factor.

```
burn_in_phase <- 20500
result1 <- table(samples1[(burn_in_phase+1):N, "R"],
    samples1[(burn_in_phase+1):N, "C"])/(N-burn_in_phase)
rownames(result1) <- c("R0", "R1")
colnames(result1) <- c("C0", "C1")
result1
```

```
##
##              C0         C1
##   R0 0.63315254 0.03471186
##   R1 0.18230508 0.14983051
```

```
result2 <- table(samples2[(burn_in_phase+1):N, "R"],
    samples2[(burn_in_phase+1):N, "C"])/(N-burn_in_phase)
rownames(result2) <- c("R0", "R1")
colnames(result2) <- c("C0", "C1")
result2
```

```
##
##              C0         C1
##   R0 0.64752542 0.03288136
##   R1 0.17630508 0.14328814
```

```
marginal1.R.S1_W1 <- sum(result1[2,])
marginal2.R.S1_W1 <- sum(result2[2,])
print("Marginal probability using the first run of 50000 iterations and excluded burn-in phase:")
```

```
## [1] "Marginal probability using the first run of 50000 iterations and excluded burn-in phase:"
```

```
print(marginal1.R.S1_W1)
```

```
## [1] 0.3321356
```

```
print("Marginal probability using the second run of 50000 iterations and excluded burn-in phase:")
```

```
## [1] "Marginal probability using the second run of 50000 iterations and excluded burn-in phase:"
```

```
print(marginal2.R.S1_W1)
```

```
## [1] 0.3195932
```

Analytical solution of marginal probability (i)

```
S1_W1 <- p$W.S1_R0*p$S.C0*(1-p$R.C0)*(1-p$C) +
         p$W.S1_R0*p$S.C1*(1-p$R.C1)*p$C +
         p$W.S1_R1*p$S.C0*p$R.C0*(1-p$C) +
         p$W.S1_R1*p$S.C1*p$R.C1*p$C

R1_S1_W1 <- p$W.S1_R1*p$S.C0*p$R.C0*(1-p$C) +
            p$W.S1_R1*p$S.C1*p$R.C1*p$C

analytical_marginal.R.S1_W1 <- R1_S1_W1 / S1_W1
```

```r
print("Analytical solution for the marginal probability:")
```

```
## [1] "Analytical solution for the marginal probability:"
```

```r
print(analytical_marginal.R.S1_W1)
```

```
## [1] 0.3203883
```

The marginal probabilities of the (h) part are closer to the analytical solution, meanwhile the probability of (c) part differs more. It is intuitive, since the sample size in (c) was small (100).

```r
library(rmarkdown)
render("Gibbs.Rmd", pdf_document(TRUE), "Indilewitsch_Toidze_Houhamdi_Pudziuvelyte_Project6.pdf")
```