Solution **Homework 5** ISyE 6420 November 11, 2020

1 Paddy Soil Adhession.

We run the OpenBUGS code and obtain the results shown in Figure 1. The code is attached in Appendix A.

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
BR2	0.6748	0.07591	2.728E-4	0.4961	0.6857	0.7907	1001	100000
beta0	0.3991	0.1294	0.001104	0.1443	0.3981	0.6552	1001	100000
beta1	0.7682	0.08246	7.089E-4	0.605	0.7686	0.9304	1001	100000
pH1.beta0	1.0E-5	0.003162	9.976E-6	0.0	0.0	0.0	1001	100000
pH1.beta1	0.9971	0.05396	2.369E-4	1.0	1.0	1.0	1001	100000
rubber.pred	1.933	0.4227	0.00127	1.101	1.932	2.766	1001	100000

Figure 1: OpenBUGS result for problem 1

(a) As the mean value of β_0 and β_1 is 0.3991 and 0.7682, respectively, the fitted model is

$$y = 0.3991 + 0.7682x,$$

where the response variable y is adhesion to rubber and the variable x is adhesion to steel. The Bayesian R^2 is 0.6748.

(b) Based on the output, the predictive response has a mean of 1.933 and a 95% credible set as $[1.101, 2.766]$.

2 Third-degree Burns.

By running the OpenBUGS code (attached in Appendix B), we obtain the results shown in Figure 2.

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
beta0	25.41	11.3	0.6239	18.77	23.1	67.72	1001	100000
beta1	-11.94	5.348	0.2951	-31.76	-10.85	-8.78	1001	100000
deviance	44.69	47.79	2.688	33.15	34.6	207.8	1001	100000
p.star	0.8095	0.04549	0.002108	0.7474	0.8032	0.9767	1001	100000

Figure 2: OpenBUGS result for problem 2

(a) Based on the estimated parameters, we obtain the logistic regression model as

$$y[i] \sim \mathcal{B}in(n_i, p_i), \text{ where } \mathrm{logit}(p_i) = 25.41 - 11.94x_i, i = 1, \dots, 9,$$

where y is the number of survivals, p is the probability of survival rate and n is total number patients.

We see that the deviance of the model is 44.69.

(b) When `log(area+1` equals 2, the posterior probability of survival is 0.8095.

3 SO_2, NO_2, and Hospital Admissions.

By running the OpenBUGS code (attached in Appendix C), we obtain the results shown in Figure 3.

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
beta0	5.451	0.03964	0.002229	5.336	5.461	5.474	1001	100000
beta1	-0.001851	2.949E-4	1.513E-5	-0.002374	-0.001829	-0.001409	1001	100000
beta2	0.00267	7.352E-4	4.133E-5	0.002256	0.002497	0.005056	1001	100000
lambda.star	280.6	7.993	0.4434	275.3	278.7	305.4	1001	100000

Figure 3: OpenBUGS result for problem 3

(a) We obtain the following model with the estimated parameters.

$$y_i \sim \mathcal{P}oi(\lambda_i), \text{ where } \lambda_i = \exp\{5.451 - 0.001851 x_{i1} + 0.00267 x_{i2}\}, i = 1, \ldots, n.$$

Based on the estimated coefficients, we see that the level of SO_2 has a negative coefficient (-0.001851) and the level of NO_2 has a positive coefficient (0.00267). This means that the level of SO_2 has a negaive impact on the hospital admission, while the level of NO_2 has a positive impact on the hospital admission.

(b) Based on the results, the expected number of hospital admissions is 280.6 and the 95% credible set is $[275.3, 305.4]$.

A OpenBUGS code for Problem 1

```
model {

for (i in 1:n) {
rubber[i] ~ dnorm(mu[i], tau)
mu[i] <- beta0 + beta1 * steel[i]
}

beta0 ~ dnorm(0, 0.0001)
beta1 ~ dnorm(0, 0.0001)
tau ~ dgamma(0.0001, 0.0001)
sigma2 <- 1/tau

p <- 2
# Bayesian R2
nminusp <- n-p
sse <- nminusp * sigma2
for (i in 1:n) {
crubber[i] <- rubber[i] - mean(rubber[])
}
sst <- inprod(crubber[], crubber[])
BR2 <- 1 - sse/sst

# test posterior probability of hypothesis beta0>1
pH1.beta0 <- step(-1+beta0)

# test posterior probability of hypothesis beta1<1
pH1.beta1 <- step(1-beta1)

# prediction
steel.new <- 2
mu.new <- beta0 + beta1 * steel.new
rubber.pred ~ dnorm(mu.new, tau)

}
```

B OpenBUGS code for Problem 2

```
model {
```

```
for (i in 1:n) {

total[i] <- survived[i]+died[i]
survived[i] ~ dbin(p[i], total[i])
logit(p[i]) <- beta0 + beta1*x[i]

}

beta0 ~ dnorm(0, 0.0001)
beta1 ~ dnorm(0, 0.0001)

# posterior probability when x=2
x.star <- 2
logit(p.star) <- beta0 + beta1*x.star

}

# DATA
list(n=9)
x[] survived[] died[]
1.35 13 0
1.60 19 0
1.75 67 2
1.85 45 5
1.95 71 8
2.05 50 20
2.15 35 31
2.25 7 49
2.35 1 12
END
```

C OpenBUGS code for Problem 3

```
    model {

for (i in 1:n) {
admission[i] ~ dpois(lambda[i])
lambda[i] <- exp(beta0 + beta1*SO2[i] + beta2*NO2[i])
}

beta0 ~ dnorm(0, 0.0001)
```

```
beta1 ~ dnorm(0, 0.0001)
beta2 ~ dnorm(0, 0.0001)

# prediction
SO2.star <- 44
NO2.star <- 100
lambda.star <- exp(beta0 + beta1*SO2.star + beta2*NO2.star)

}

# DATA
list(n=730)


# INIT
list(beta0=0, beta1=0, beta2=0)
```