UNIVERSITEIT VAN DIE VRYSTAAT
UNIVERSITY OF THE FREE STATE

STSB 6816

WISKUNDIGE STATISTIEK & AKTUARIËLE WETENSKAP/
MATHEMATICAL STATISTICS & ACTUARIAL SCIENCE

Special Test — 13 June 2019

MEMORANDUM

TYD/TIME: 100 Minutes PUNTE/MARKS: 38

INSTRUCTIONS:

- Answer all questions in a single Word document. Please convert to .pdf at the end.
- Label questions clearly, as it is done on this question paper.
- All results accurate to 2 decimal places.
- Show all derivations, formulas, code, sources and reasoning.
- Intervals should cover 95% probability unless stated otherwise.
- No communication software, devices or websites may be accessed prior to submission.

Question 1

This test is based on a supplied data set given on your LMS as 'Bayes2019Test3Data.csv'. It is a set of measurements taken on breast biopsies to test for cancer.

In this dataset the final column (*class*) has a value of 1 for cases where cancer was present, and a value of zero where cancer was not present.

Your goal is to build a Bayesian model that can predict cancer presence according to these measurements.

(a) Read in the data and produce a summary. All measurements should go from 1 to 10. The class should be half zeros and half ones. [2]

```
mydata <- read.csv('Bayes2019Test3Data.csv')
summary(mydata)
```

✓✓

(b) Attempt to fit a binomial GLM to predict the class, based on the 9 measurements plus intercept. Discuss and try to explain any problems you encounter. [4]

```
y <- mydata$class
n <- nrow(mydata)
k <- ncol(mydata)
Xdf <- mydata[,1:9]
Xmat <- as.matrix(Xdf)
X1 <- cbind(rep(1,n),Xmat)
summary(model1 <- glm(y~.,data = Xdf,family = binomial()))
cormat <- cor(Xmat)
cat('\nHighest correlation within X:',max(cormat[cormat<1]),'\n')
```

✓ ✓for glm line correct

```
 glm.fit:  algorithm did not converge
glm.fit:  fitted probabilities numerically 0 or 1 occurred
.   .   .
Highest correlation within X: 0.9138933
```

✓

✓for attempted explanation such as that there is multicolinearity in the data.

(c) Now fit the conservative Bayesian model as given below. Give the posterior mean estimates of the parameters from at least 20000 samples. [8]

$$\begin{aligned} class_i &\sim Bernoulli(p_i) \\ p_i &= \mathbf{x}_i\boldsymbol{\beta} \\ \beta_j &\sim N(0, 10^2)\ \forall\ j = 1, \ldots, 10 \\ x_{i1} &= 1\ \forall\ i = 1, \ldots, 100 \end{aligned}$$

```
library(R2OpenBUGS)
logitmodel <- function() {
  for (i in 1:n) {
    y[i] ~ dbern(p[i])
    p[i] <- ilogit(inprod(beta[1:k],X[i,1:k]))
  }
  for (j in 1:k) {
    beta[j] ~ dnorm(0,0.01)
  }
}
write.model(logitmodel,'logitmodel.txt')
BUGSdata <- list(n=n, k=k, X=X1, y=y)
inits <- function() { list(beta=rnorm(k,0,0.1)) }
bugsoutput <- bugs(BUGSdata,inits,c('beta'),22000, 'logitmodel.txt', 3, 2000, 2,
    debug=TRUE)
bugsoutput$mean
```

✓ ✓ ✓ ✓ ✓ ✓

-29.121 2.635 -1.067 1.264 2.006 0.507 1.104 0.768 -0.399 0.677 ✓ ✓

(d) What type of prior is being used for β_i? Also explain why the model can be seen as conservative. [3]

Because it pulls the betas to zero, it is technically a subjective or informative prior. Nevertheless, it is still a neutral prior. It can be seen as a Lasso-style model because it tries to penalise deviations from zero, and thus it is less likely to indicate a beta as different from zero, making it a conservative model. ✔ ✔ ✔[for any three valid points]

(e) Use your Bayes model output to predict each class for the observed sample. Make a table of counts of predictions vs actual classes (confusion matrix). Give this 2 by 2 table and explain what the numbers mean. [5]

```
nsim <- bugsoutput$n.sims
yprobhat <- plogis(bugsoutput$sims.list$beta%*%t(X1))
yhat <- (colMeans(yprobhat)>0.5)
table(yhat,y)
```

✔ ✔ ✔

The table will probably have 2 predictions where the model says cancer when it actually wasn’t. ✔ ✔

(f) Calculate a Bayesian equivalent of p-values for $\beta_1, \beta_2, \ldots, \beta_{10}$ (to attempt to see how they differ from 0). [3]

```
pvalfunc <- function(sims) {2*min(mean(sims<0),mean(sims>0))}
pvals <- apply(bugsoutput$sims.list$beta,2,pvalfunc)
names(pvals) <- colnames(X1)
print(pvals)
```

✔ ✔

✔

intercept	clump	ucellsize	ucellshape	mgadhesion
0.000	0.000	0.266	0.321	0.003
sepics	bnuclei	bchromatin	normnucl	mitoses
0.348	0.000	0.273	0.349	0.636

(g) Now refit the model using only the explanatory variables that your initial model indicated as having less than 5% chance of being insignificant. Compare the DIC of the initial model and the DIC of the new model, with interpretation. [6]

```
Xred <- X1[,pvals<0.05]
kred <- ncol(Xred)
BUGSdata <- list(n=n, k=kred, X=Xred, y=y)
inits <- function() { list(beta=rnorm(kred,0,0.1)) }
bugsoutputred <- bugs(BUGSdata,inits,c(’beta’),22000, ’logitmodel.txt’, 3, 2000, 2,
    debug=TRUE)
cat(’\nThe DIC of the full model is’,bugsoutput$DIC,’; while the DIC of the reduced
    model is’,bugsoutputred$DIC,’.\n’)
```

✔ ✔ ✔ ✔

The DIC of the full model is 17.37; while the DIC of the reduced model is 14.67. This implies that the reduced model is more parsimonious. ✔ ✔

(h) Import the extra sample of data provided in ‘Bayes2019Test3ExtraSample.csv’. Use your already fitted Bayes model output to predict each class for the extra sample. Make a table of counts of predictions vs actual classes (confusion matrix). Give this 2 by 2 table and explain what the numbers mean. [7]

```
newsample <- read.csv(’Bayes2019Test3ExtraSample.csv’)
ynew <- newsample$class
nnew <- nrow(newsample)
knew <- ncol(newsample)
Xdfnew <- newsample[,1:9]
Xmatnew <- as.matrix(Xdfnew)
Xnew <- cbind(rep(1,nnew),Xmatnew)
yprobhatnew <- plogis(bugsoutput$sims.list$beta%*%t(Xnew))
yhatnew <- (colMeans(yprobhatnew)>0.5)
table(yhatnew,ynew)
```

✓✓✓✓✓

✓

yhatnew\ynew	0	1
FALSE	98	6
TRUE	2	94

In this sample there are 6 cases of actual cancer that were screened as fine by the model, which is worrying, and shows over-fitting. ✓

Total for Question 1: 38

Total half marks on memo = 76 vs. 76 = Double total margin points (=38).