Bayesian Modelling – Problem Sheet 5 Part B

Please hand in your solutions for Problems 1-4 by 6pm on Monday 29/04/2019

Problem 1

1. Brouwer's fixed-point theorem states that if f is a continuous mapping from a compact set $\mathcal{K}$ into itself then there exists an $x_0 \in \mathcal{K}$ such that $x_0 = f(x_0)$.

   Using this result show that every transition matrix P on $\mathcal{Y} := \{1, \dots, m\}$ admits an invariant distribution.

2. Let
$$P = \begin{pmatrix} 1 & 0 & 0 \\ 1/3 & 1/3 & 1/3 \\ 0 & 0 & 1 \end{pmatrix}.$$
   Show that P has infinitely many invariant distributions. Is P an irreducible and aperiodic transition matrix?

3. Let
$$P = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}.$$
   Show that P has a unique invariant distribution μ and that P does not satisfy the detailed balance condition for μ.

4. Let $P = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$.
   a) Show that P is irreducible and periodic.
   b) Show that P has a unique invariant distribution μ.
   c) Show that, for all $(i, j) \in \mathcal{Y}^2$, $\lim_{t \to +\infty} p_{ij}^{(t)}$ does not exist.

Problem 2 (Invariant distribution of the Gibbs sampler)

Let $\mathcal{Y} \subset \mathbb{R}^d$ be a state space and $\mu \in \mathcal{P}(\mathcal{Y})$.

1. Let $i \in \{1, \dots, d\}$ be arbitrary and consider the Metropolis-Hastings algorithm (Algorithm (A2) in the lecture notes) with proposal distribution $q_i(\tilde{y}|y)$ defined by
$$q_i(\tilde{y}|y) = \mu^{(i)}(\tilde{y}^{(i)} | \, y^{(-i)}) \mathbb{1}_{\{y^{(-i)}\}}(\tilde{y}^{(-i)}), \quad \tilde{y}, y \in \mathcal{Y}.$$
   Show that $\mathbb{P}\big(Y_t = \tilde{Y}_t\big) = 1$.

2. Let $k_1(\tilde{y}|y)$ and $k_2(\tilde{y}|y)$ be two transition kernels on $\mathcal{Y}$ that admit μ as invariant distribution. Show that the transition kernel

$$k(\tilde{y}|y) := \int_{\mathcal{Y}} k_2(\tilde{y}|y')k_1(y'|y)\mathrm{d}y'$$

has μ as invariant distribution.

3. Using the results in part 1 and in part 2, show that the Gibbs sampler (Algorithm (A3) in the lecture notes) has μ as invariant distribution.

Problem 3 (ARCH models)

Let $p_1, \ldots, p_{101}$ be the daily closing prices of the S&P 500 index between 18/10/2018 and 15/03/2019, and let $x_t = \log(p_{t+1}/p_t)$, for $t = 1, \ldots, 100$, be the corresponding daily log-returns. In this problem we use an ARCH (Generalized Auto-Regressive Conditionally Heteroscedastic) model to model the observation $x := (x_1, \ldots, x_{100})$. An ARCH(q) model, which is a popular model for the log-returns of a financial asset, is defined by

$$X_t = \sigma_t \epsilon_t, \quad \sigma_t^2 = \omega + \sum_{i=1}^{q} \alpha_i X_{t-i}^2, \quad t \geq 1 \tag{1}$$

where the ϵ_t's are i.i.d. $\mathcal{N}_1(0,1)$ random variables,

$$\theta_q := (\omega, \alpha_1, \ldots, \alpha_q) \in \Theta_q := \mathbb{R}_{>0}^{q+1},$$

and where $X_{t-i}^2 = 0$ whenever $t - i \leq 0$. Let $f_q(x|\theta_q)$ be the likelihood function of the model. We make the standard assumption that $\sum_{i=1}^{q} \alpha_i < 1$, which implies that the stochastic process $(X_t)_{t\geq 1}$ defined in (1) is stationary. To incorporate this prior information we take as prior distribution for $(\alpha_1, \ldots, \alpha_q)$ the uniform distribution on the set

$$\Big\{ y \in \mathbb{R}_{>0}^{q} : \sum_{i=1}^{q} y_i < 1 \Big\}.$$

We let the Gamma$(1, 1/2)$ distribution be the prior distribution for ω and use the notation $\pi_q(\theta_q)$ for the so-defined prior distribution on Θ_q. It can be shown that, for $i = 1, \ldots, q$, the marginal distribution of α_i under $\pi_q(\theta)$ is the Beta(1,q) distribution.

The data[1] are in the file SP500.txt and can be loaded in R as follows (assuming that your .R file is in the same folder as the DataSheet5 folder):

```
# Load SP500 data
p<-read.table('DataSheet5/ARCH/SP500.txt')
# Compute log-returns
x<-diff(log(p[,5]))
```

[1] Available here: `https://finance.yahoo.com/quote/^GSPC/history/`

1. Plot all the marginal distributions of $\pi_q(\theta_q)$ for $q = 1, 2$. Is the prior distribution for α_1 non-informative in the sense of Laplace for $q \in \{1, 2\}$?

2. Consider first an ARCH(1) model.
   a) Implement in R a Metropolis-Hastings algorithm having
   $$\pi_1(\theta_1|x) \propto f_1(x|\theta_1)\pi_1(\theta_1)$$
   as invariant distribution, starting value $\theta_{1,0} = (0.01, 0.1)$ and where the proposal distribution $Q(\theta_1, \mathrm{d}\tilde{\theta}_1)$ is the $\mathcal{N}_2(\theta_1, \Sigma_1)$ distribution. Propose a matrix Σ_1, a burn-in period B and a length T for the simulated trajectory (burn-in period non-included). Justify your choices.
   b) Plot the estimated marginal distributions of $\pi_1(\theta_1|x)$ and give a 99% confidence interval for each of the corresponding posterior means.

3. Consider now an ARCH(2) model.
   a) Implement in R a Metropolis-Hastings algorithm having
   $$\pi_2(\theta_2|x) \propto f_2(x|\theta_2)\pi_2(\theta_2)$$
   as invariant distribution, starting value $\theta_{2,0} = (0.01, 0.1, 0.1)$ and where the proposal distribution $Q(\theta_2, \mathrm{d}\tilde{\theta}_2)$ is the $\mathcal{N}_3(\theta, \Sigma_2)$ distribution, with Σ_2 the matrix given in the file Sigma_2.txt:
   ```
   Sigma<-as.matrix(read.table('DataSheet5/ARCH/Sigma_2.txt'))
   ```
   Propose a burn-in period B and a length T for the simulated trajectory (burn-in period non-included). Justify your choices.
   b) Plot the estimated marginal distributions of $\pi_2(\theta_2|x)$ and give a 99% confidence interval for each of the corresponding posterior means.

4. Consider now the test, in the ARCH(2) model, of the hypothesis $H_0 : \alpha_2 \leq 0.1$ against the alternative $H_1 : \alpha_2 > 0.1$.
   a) Provide an estimate and a 99% confidence interval for $\pi_2\big(\{\theta_2 : \alpha_2 \leq 0.1\}|x\big)$. Do you accept H_0 under the $a_0 - a_1$ loss function when $a_0 = a_1$?
   b) Compute an estimate of $B_{01}^{\pi_2}(x)$, the Bayes factor for the test we are considering.
   c) Comment the results obtained in parts 4.a) and 4.b).

5. Based on the results of part 4, choose between the ARCH(1) model and the ARCH(2) model.

Problem 4 (Spam filtering)

The goal of this problem is to build a classifier that can be used to classify an email as a spam or as a non-spam. In this context, a classifier is a function $\gamma : \mathbb{R}^d \to \{0, 1\}$ that takes as input a d-dimensional vector $z \in \mathbb{R}^d$ containing information about the email (such as the occurrence of certain characters) and that returns as output the value 1 if the email is classified as a spam and the value zero otherwise.

To this aim, the file spambase.data contains $n = 4\,601$ observations $\{\tilde{z}_i, x_i\}_{i=1}^n$, where $x_i = 1$ if email i is a spam and $x_i = 0$ otherwise while $\tilde{z}_i \in \mathbb{R}^{57}$ is a vector containing 57 attributes (or features). The description of the 57 features are given in the file spambase.names.txt while the file spambase.DOCUMENTATION.txt contains some summary statistics for each feature. In this dataset, about 40% of the emails are spam emails.

A simple model that can be used for classification is the probit regression model which assumes that the observations are independent and such that

$$X_i \sim \text{Bernoulli}\Big(\Psi\big((1, \tilde{z}_i)^T \theta\big)\Big), \quad i = 1, \dots, n \tag{2}$$

where Ψ is the cumulative density function of the $\mathcal{N}_1(0, 1)$ distribution and $\theta \in \mathbb{R}^{58}$.

Using machine learning terminology, we partition the dataset into a training set $\{\tilde{z}_i, x_i\}_{i=1}^{n_{\text{train}}}$ and a test set $\{\tilde{z}_i, x_i\}_{i=n_{\text{train}}+1}^{n}$. The training set, which contains about 80% of the observations ($n_{\text{train}} = 3680$), will be used to build and estimate the model, while the test set will be used to estimate its classification error. (To simplify the notation here we do as if the training set contains the first n_{train} observations but in fact the elements of this set have been chosen randomly.)

The training set[2] can be loaded in R as follows (assuming that your .R file is in the same folder than the DataSheet5 folder):

```
train_set<-read.table('DataSheet5/spam/spambase_train.txt')
```

The vector $x_{\text{train}} := \{x_i\}_{i=1}^{n_{\text{train}}}$ and the $(n_{\text{train}} \times 57)$ matrix $\tilde{Z}_{\text{train}} := \{\tilde{z}_i\}_{i=1}^{n_{\text{train}}}$ can be obtained as follows:

```
x_train<-as.matrix(train_set[,ncol(train_set)])
Z_train<-as.matrix(train_set[,1:(ncol(train_set)-1)])
```

If we use the whole matrix $\tilde{Z}_{\text{train}}$ to build our classifier then we have to estimate 58 parameters (one per feature plus one intercept) which may be challenging both from a computational point of view (i.e. running a Markov chain on a 58 dimensional space is complicated) and from a statistical point of view (i.e. the larger is the number of parameters we estimate the less precise the estimation is). Moreover, if too many parameters are introduced in the model there is a risk of overfitting, namely that the estimated model corresponds too closely to the training data and thus provides poor out-of-sample predictions (i.e. a large classification error on the test set).

[2]The complete dataset is available here: `https://archive.ics.uci.edu/ml/datasets/spambase`

The typical machine learning problem in this context is to choose the combination $(j_1, \ldots, j_k)$ of $k \in \{1, \ldots, 57\}$ columns of $\tilde{Z}_{\text{train}}$ that minimizes the prediction error of the classifier. In other words, for different values of k and column indices $(j_1, \ldots, j_k)$, the model is estimated on the training set, its classification error on the test set is computed, and we retain the model that minimizes this latter.

Here, we propose to keep columns (5,6,7,8,9,16,17,18,19,20,21,23,24, 45, 57) of $\tilde{Z}_{\text{train}}$ (again, see the file spambase.names.txt to figure out what features these columns correspond to):

```
keep<-c(5,6,7,8,9,16,17,18,19,20,21,23,24,45,57)
Z_train<-Z_train[,keep]
```

We add an intercept in the model:

```
Z_train<-cbind(rep(1,nrow(Z_train)),Z_train)
```

Let Z_{train} denote the resulting $n_{\text{train}} \times d$ matrix, with $d = 16$, and let $z_i \in \mathbb{R}^d$ be the i-th row of Z_{train}. For $\theta \in \Theta := \mathbb{R}^d$ let

$$\tilde{f}_z(x|\theta) = \Phi(z^T\theta)^{x_i}\big(1 - \Phi(z^T\theta)\big)^{1-x}, \quad z \in \mathbb{R}^d, \quad x \in \{0, 1\} \tag{3}$$

so that, for $i = 1, \ldots, n_{\text{train}}$, the likelihood of observation x_i given θ is $\tilde{f}_{z_i}(x_i|\theta)$. To complete the Bayesian model we let $\pi(\theta)$ be the density of the $\mathcal{N}_d(\mu_0, \Sigma_0)$ distribution.

We now show that this particular choice for $\pi(\theta)$ yields a Bayesian model such that the posterior distribution

$$\pi(\theta|x_{\text{train}}) \propto \pi(\theta) \prod_{i=1}^{n_{\text{train}}} \tilde{f}_{z_i}(x_i|\theta)$$

can be efficiently approximated using a Gibbs sampler. To this aim remark first that assuming (2) (with $(1, \tilde{z}_i) \in \mathbb{R}^{58}$ replaced by $z_i \in \mathbb{R}^d$) is equivalent to assuming that

$$Y_i \sim \mathcal{N}_1(z_i^T\theta, 1), \quad X_i|Y_i = \begin{cases} 1, & Y_i > 0 \\ 0, & \text{otherwise} \end{cases}, \qquad i = 1, \ldots, n_{\text{train}} \tag{4}$$

where $Y := (Y_1, \ldots, Y_{n_{\text{train}}})$ is a vector of auxiliary variables.

Using the result of Problem Sheet 1, Problem 2, it is easily checked that the full conditional distributions of the extended posterior distribution $\pi(\theta, y|x_{\text{train}})$ are given by

$$\theta|Y, x_{\text{train}} \sim \mathcal{N}_d\Big((Z_{\text{train}}^T Z_{\text{train}} + \Sigma_0^{-1})^{-1}\big(\Sigma_0^{-1}\mu_0 + Z_{\text{train}}^T Y\big), (Z_{\text{train}}^T Z_{\text{train}} + \Sigma_0^{-1})^{-1}\Big)$$

$$Y_i|\theta, Y_{-i}, x_{\text{train}} \sim \begin{cases} \mathcal{TN}_{(0,+\infty)}\big(z_i^T\theta, 1\big), & x_i = 1 \\ \mathcal{TN}_{(-\infty,0]}\big(z_i^T\theta, 1\big), & x_i = 0, \end{cases} \qquad i = 1, \ldots, n_{\text{train}}$$

where $\mathcal{TN}_A(\mu, 1)$ denotes the $\mathcal{N}_1(\mu, 1)$ distribution truncated on $A \subset \mathbb{R}$. Sampling from $\theta|Y, x_{\text{train}}$ is therefore trivial while efficient methods exist for sampling from a truncated normal distribution. This auxiliary variables approach has been proposed by

Holmes, C. C., & Held, L. (2006). Bayesian auxiliary variable models for binary and multinomial regression. Bayesian analysis, 1(1), 145-168.

1. Taking $\mu_0 = (0, \ldots, 0)$ and $\Sigma_0 = 100 I_d$ (vague prior), use JAGS to get an approximation of $\pi(\theta|x_{\text{train}})$. Propose a burn-in period B and a length T for the simulated trajectory (burn-in period non-included). Justify your choices.

   To tell JAGS to use the auxiliary variables approach described above you need to load, before running the jags..model() command, the module glm:

   ```
   load.module('glm')
   ```

   Remark: In JAGS you only need to implement the model (2), that is you do not need to implement the model (4) with the auxiliary variables $Y_1, \ldots, Y_{n_{\text{train}}}$ (the module glm does it for you).

   Warning: If you do not use the latest version of JAGS (i.e. JAGS 4.3.0) the module glm may not work correctly. To check that JAGS is indeed going to use the auxiliary variables approach described above you can type the command:

   ```
   list.samplers(mymodel)
   ```

   which should return glm::Holmes-Held if everything works well.

2. We now construct our classifier $\gamma : \mathbb{R}^d \to \{0, 1\}$.

   a) Let $z \in \mathbb{R}^d$, $X'|(\theta, x_{\text{train}}) \sim \tilde{f}_z(x'|\theta)$ (i.e. X' is conditionally independent of X_{train} given θ) and $\theta \sim \pi(\theta|x_{\text{train}})$, with $\tilde{f}_z(\cdot|\theta)$ as in (3). Show that the posterior distribution of X' given x_{train} is

   $$\pi(x'|x_{\text{train}}) = \int_\Theta \tilde{f}_z(x'|\theta)\pi(\theta|x_{\text{train}})\mathrm{d}\theta.$$

   b) Let $z \in \mathbb{R}^d$ and X' be as in part 2.a). Using the $a_0 - a_1$ loss function with $a_0 = a_1$ consider the Bayesian test $H_0 : X' = 1$ against the alternative $H_1 : X' = 0$. Write in R the function $\delta^\pi_{x_{\text{train}}} : \mathbb{R}^d \to \{0, 1\}$ which is such that $\delta^\pi_{x_{\text{train}}}(z) = 1$ if H_0 is accepted and zero otherwise. Let $\gamma = \delta^\pi_{x_{\text{train}}}$ be our classifier.

3. We now want to use the test set to asses our model and the performance of the classifier γ defined in part 2.b).

   The test set can be loaded in R as follows:

   ```
   test_set<-read.table('DataSheet5/spam/spambase_test.txt')
   ```

   The vector $x_{\text{test}} := \{x_i\}_{n_{\text{train}}+1}^n$ and the matrix $\tilde{Z}_{\text{test}} := \{\tilde{z}_i\}_{n_{\text{train}}+1}^n$ can be obtained as follows:

```
x_test<-as.matrix(test_set[,ncol(test_set)])
Z_test<-as.matrix(test_set[,1:(ncol(test_set)-1)])
```

We keep the same columns as for the training set:

```
Z_test<-Z_test[,keep]
```

Finally, we add an intercept:

```
Z_test<-cbind(rep(1,nrow(Z_test)),Z_test)
```

To simplify the notation let $I_{\text{test}} = \{n_{\text{train}} + 1, \ldots, n\}$ and, for $j = 0, 1$, let $I_{j,\text{test}} = \{i \in I_{\text{test}} : x_i = j\}$.

a) Compute the estimated value of $\pi(x_i|x_{\text{train}})$ for all $i \in I_{\text{test}}$. Using two separate boxplots, plot $\pi(x_i|x_{\text{train}})$ for $i \in I_{0,\text{test}}$ and $\pi(x_i|x_{\text{train}})$ for $i \in I_{1,\text{test}}$. Comment the results.

b) Compute the following quantities:

$$\frac{\sum_{i\in I_{\text{test}}} |\gamma(z_i) - x_i|}{\#I_{\text{test}}}, \quad \frac{\sum_{i\in I_{0,\text{test}}} |\gamma(z_i) - x_i|}{\#I_{0,\text{test}}}, \quad \frac{\sum_{i\in I_{1,\text{test}}} |\gamma(z_i) - x_i|}{\#I_{1,\text{test}}}$$

where $\#I$ denotes the cardinality of the set $I \subset \mathbb{N}$. Based on these results discuss the performance of the classifier γ.

4. (**Optional**) Try to find another list of features (i.e. another vector keep$\subset \{1, \ldots, 57\}$) that yields a smaller classification error than the one obtained in part 3.b).