Research Article

Comparison of multimedia communications QoE models by Bayesian networks and Bayesian statistics: a case study

Shuji Tasaka[1]

Abstract
This paper presents a comparison of *Bayesian Network* (*BN*) and *Bayesian Statistics* (*BS*) modeling for *QoE* (*Quality of Experience*) estimation and prediction in multimedia communications, with special attention to prediction. As an example of the comparison, we employ a haptic-audiovisual interactive communication system with guaranteed bandwidth. The QoE measure adopted here is subjects' overall satisfaction (average score) of performing an interactive task under conditions specified by combinations of the video guaranteed bandwidth, video encoding bit rate, receiver's playout buffering time and gender of each subject. For BN modeling, we utilize an R package **bnlearn** and create a discrete BN model of a *directed acyclic graph* with four nodes corresponding to the four parameters. For BS modeling, we build (1) a Bayesian hierarchical regression model with covariates of the four parameters and random effect terms reflecting users' individualities and gender, and (2) a Bayesian regression model without the random effect terms. The two BS models are analyzed by *Markov chain Monte Carlo* (*MCMC*) simulation with the software *OpenBUGS*. We then find that the BN and BS models provide approximately the same estimates of the QoE measure. Regarding the prediction, however, the BS model with random effect terms outperforms the BN model and BS model without random effect terms. We thus learn that the random effect terms enhance the ability of Bayesian approaches in QoE prediction.

Keywords Quality of Experience (QoE) · Bayesian network · Bayesian statistics · bnlearn · OpenBUGS · Haptic-audiovisual communications

Mathematics Subject Classification 62F15 · 62M20 · 65C05 · 68M11 · 68M12 · 68Q32 · 68T05 · 68T37 · 90B18 · 91E10 · 91E45 · 94A08

1 Introduction

Quality of Experience (*QoE*) in multimedia communications is a truly end-to-end quality of multimedia services offered to the users since it includes the users' perceived quality of the service; it is influenced by not only *Quality of Service* (*QoS*) of the network but also surrounding environments and human factors such as gender, age and other demographic properties [1–3]. This complicates quantitative assessment of QoE.

To cope with the problem, many methodologies have been proposed according to situations to be assessed; e.g., [4–11] Among them, *Bayesian* methodologies provide powerful approaches to the problem; they have also been widely utilized as probabilistic modeling tools of *uncertainty* in a variety of applications [12–17].

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s42452-019-0983-5) contains supplementary material, which is available to authorized users.

✉ Shuji Tasaka, tasaka@nisri.jp; s.tasaka@nitech.jp | [1]Nagoya Industrial Science Research Institute, Nagoya 464-0819, Japan.

SN Applied Sciences (2019) 1:975 | https://doi.org/10.1007/s42452-019-0983-5

Received: 14 January 2019 / Accepted: 24 July 2019 / Published online: 3 August 2019

Two main methods in this category are *Bayesian Network* (*BN*) [12–14] and *Bayesian Statistics* (*BS*) [15–17]. The BN method is a probabilistic inference tool (a *machine learning* technique) in artificial intelligence; it is based on *graphical modeling* using *directed acyclic graphs* (*DAGs*), which encode the *conditional independence* between nodes, a key concept in BNs. On the other hand, modern Bayesian statistics utilize computer-simulation based techniques for deriving *posterior probabilities* of unknown parameters of interest; we suppose this type as the BS method in this paper.

A noticeable advantage of the Bayesian approaches is the capability of explicitly incorporating influencing factors on QoE and *prior information* into models; this enable us to investigate the effects of the factors on QoE in systematic and efficient ways using priors.

The two methods utilize a common basic rule, *Bayes' theorem*, for computing posterior probabilities. So, some parts of each implementation share the same techniques, but there are many different methodological details between the two.

Studies on assessment of *multimedia communications QoE* by the BN method appeared in literature earlier than those by the BS method; e.g., see [4, 18, 19, 23, 24, 26] for BN, and [6, 20, 21] for BS. Despite the similarity in name, however, no comparative study on multimedia communications QoE assessment between the two methods can be found in publications.

There seems to be even some confusion between the two methods in this area of QoE study. Thus, advantages and disadvantages of each method have not yet been clarified. *The clarification of the relationship can provide information on how we should effectively use each method as a multimedia tool according to a given problem*; this contributes to promoting research on quantitative assessment of QoE and is the aim of this paper.

This paper is a first trial of comparing the BN and BS as methodologies for estimation and prediction of QoE in multimedia communications, paying special attention to prediction.

As a material of the comparison, this paper employs a *haptic-audiovisual* interactive communication system, since multisensory communications of this type are regarded as offering one of the most promising multimedia services in near future [22]. Haptic-audiovisual communications also exhibit diversified traffic characteristics of component media, which bring an appropriate problem for an approach by the Bayesian methods. The *overall satisfaction* in performing an interactive task is defined as QoE, which is represented as the average of five-point scores given by users.

It should be noted that the methodology proposed in this paper for comparison between BN and BS is not restricted to the haptic-audiovisual interactive communication system but applicable to any other multimedia communication systems.

This paper aims at clarification of the relationship between BN and BS from an inferential methodological point of view by demonstrating a case study. We do not intend holistic QoE assessment of the haptic-audiovisual interactive communication system itself, which is outside the scope of the current paper and should be treated as a separate research subject with multidimensional QoE measures whose components include haptic measures, audiovisual ones and interstream ones as well as overall satisfaction as in [21].

The remainder of the paper is organized as follows. Section 2 overviews related work in multimedia communications QoE modeling by BN and BS. Section 3 outlines methodologies adopted for a comparison of BN and BS modeling. Section 4 describes the haptic-audiovisual interactive communication system treated in this paper, along with an interactive task to be performed. It also presents subjective experimental data of scores for QoE and introduces data formats for Bayesian modeling. Section 5 proposes a BN model and carries out QoE estimation and prediction. Section 6 presents two BS models. Section 7 compares results of QoE estimation and prediction by the BS models with those by the BN model. Section 8 concludes the paper.

This paper is accompanied by electronic supplementary material which consists of 10 files (four dataset csv files, five R codes, and an OpenBUGS code) in addition to a README file (42452_2019_983_MOESM11_ESM.pdf).

2 Related work

2.1 Fundamentals of BS and BN

First of all, we briefly review fundamentals of Bayesian statistics (BS) and Bayesian network (BN). Both methods utilize *Bayes' theorem*, which evaluates the *posterior probability distribution* of unknown parameter $\boldsymbol{\theta}$ conditional on the observed data (i.e., samples) $\boldsymbol{y}$ by a formula

$$p(\boldsymbol{\theta}|\boldsymbol{y}) = f(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})/p(\boldsymbol{y}) \propto f(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \quad (1)$$

where $p(\cdot)$ stands for the probability density function, $f(\boldsymbol{y}|\boldsymbol{\theta})$ is the *likelihood* (the *sampling model*) and $\pi(\boldsymbol{\theta})$ the *prior distribution* of $\boldsymbol{\theta}$ [16, 17]. The BS method always specifies the prior in addition to the sampling model [16, 17]; i.e, fully Bayesian. The BN method, on the other hand, is not inherently Bayesian at all; "probabilities represented by this method can be interpreted in any number of ways, including as some form of frequency" [13].

The BN method consists of *structure learning* and *parameter learning*. Given a set of observations of random variables, a structure learning algorithm is utilized to specify the topology of a directed acyclic graph (DAG) which captures dependence relationships among *nodes* each corresponding to a random variable. DAGs, however, are often built not by using any structure learning algorithms but with technical knowledge (expert knowledge) of the target problem. The parameter learning is performed to estimate a *conditional probability table* (*CPT*) of each node; as an algorithm for the estimation, the *maximum likelihood* can be used as well as Bayesian [13, 14].

2.2 Individual researches

We next glance through research papers on multimedia communications QoE modeling by BN and BS. QoE studies by the BN method have been published in [4, 5, 18, 19, 23, 24, 26], while those by the BS appear in [6, 20, 21]. We can find no QoE study in which the same multimedia communication system is analyzed by both BN and BS with an aim of comparing the two analytic results.

As for the BN category, Ullah et al. [18] gave a model of the user behavior in Peer-to-Peer (P2P) live video streaming systems. They proposed a BN model with 12 nodes each of which represents a user behavior metric or an impacting factor like streaming quality, delay or bandwidth contribution ratio. However, QoE was not evaluated quantitatively.

Note that a QoE value in a BN model is usually calculated as a function of conditional probabilities associated with a node; e.g., the average of scores conditional on *instantiations* (all possible combinations of values of parent nodes [13]), as we will see in Eq. (2). However, ordinary BN models do not support such mechanisms.

In order to quantify QoE of a VoIP application as a BN model, Mitra et al. [4, 23] resorted to *Bayesian decision networks* (*influence diagrams*), which add *utility nodes* as well as *decision nodes* to ordinary nodes (*chance nodes*) of BNs. Each utility node has an associated *utility table* with one entry for each possible instantiation of its parent nodes and provides the *expected utility* value [13]. The utility table can be made so that the expected utility value indicates QoE. Furthermore, Mitra et al. extended their BN model to dynamic Bayesian networks for sequential QoE modeling (i.e., measuring QoE over time) [24]. The BN software *GeNIe* [25], which has a *GUI* (*Graphical User Interface*), is used in [4, 23, 24].

Carvalho et al. [26] proposed a methodology of designing video transmission over wireless LANs, combining measurement and simulation with the aid of Bayesian networks. They built a BN with six nodes (metrics) consisting of distance, delay, jitter, PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural Similarity) and RSSI (Receiver Signal Strength Indicator). The BN model provides inferences of the metrics that are used to find the requirements of specified standard values. Thus, QoE itself was not evaluated in the paper.

Mok et al. [5] applied a multiclass *Naïve Bayes classifier* to predict the quality of workers (i.e., subjects) in *crowdsourcing* for video QoE assessment, though its application is not to the evaluation of QoE itself.

Karadimce and Davcev [19] proposed a BN model for a classification of cloud-based services based on objective and subjective characteristics for perception of quality. They use a utility node to evaluate QoE, which is calculated as the expected utility value, with the software GeNIe.

From the observations so far, we have seen that the previous BN studies evaluating multimedia communications QoE employ utility nodes and that BN models without utility nodes need some specific technique for quantifying QoE.

Regarding the BS method, Tasaka [6, 20] presented a BS framework for QoE (overall satisfaction) estimation and prediction in a bandwidth guaranteed interactive *audiovisual* communication system. He built Bayesian regression models with covariates of channel bandwidth, contents of tasks, customization of playout buffering control for QoE enhancement in addition to random effect terms having hierarchical priors for the users' attributes (individualities and gender). The models are analyzed by *Markov chain Monte Carlo* (*MCMC*) simulations with the software *WinBUGS* [17, 27]. Furthermore, Tasaka [21] proposed a methodology for assessing multidimensional QoE by Bayesian analysis of *Structural Equation Models* (*SEMs*), employing a *haptic-audiovisual* interactive communication system as an example. The QoE measures are 13 subjective ones and a single objective measure of efficiency.

3 Modeling methodologies for comparison

This section elaborates on methodologies adopted by this paper for the QoE comparison between BN and BS. Because of a first trial of the comparison, this paper takes simple approaches to the modeling so that they can demonstrate the differences in estimation and prediction abilities between the two methods, if any. We pay more attention to prediction than estimation; this is because the previous studies on QoE modeling, especially BN models, have not elucidated the prediction issue in any systematic manner.

The haptic-audiovisual communication system treated in this paper has a *bandwidth guarantee* mechanism[1] [28]. It is characterized by the video guaranteed bandwidth B, the video encoding bit rate R and the playout buffering time P, while the other system parameters are kept constant.

3.1 BN modeling

For BN modeling, we develop an ordinary discrete BN model (i.e., without utility nodes), which is the most popular and simplest one among the BN models. The DAG consists of five nodes: B, R, P, G and S, where G denotes the subject's gender and S the score variable on a five-level quality scale. Since this BN model does not use any utility node, we evaluate QoE by utilizing the *conditional probability table* (*CPT*) of node S; this is one of the novelties of the current paper.

Many BN software packages are available commercially and freely. This paper has selected an R package **bnlearn** [14, 29, 30]; this is because it is a free package and provides a *command-line interface*. Although many BN software packages with GUI (Graphical User Interface) are available [13, 14], they can be used without specifying details of individual inference processes; this could prevent us from proper understanding of internal operations for the inference. On the other hand, a command-line interface requires the user to specify individual inference steps by commands; this is suitable for the current purpose of comparing the two different types of modeling, BN and BS.

3.2 BS modeling

For BS modeling, we formulate *Bayesian regression* models which have the mean of a continuous *latent variable* underlying S as the *response variable* and B, R, P and G as *explanatory variables*. We build two models: a regression model with *random effect* terms having *hierarchical priors* for the users' attribute, and a regression model without the random effect terms. We analyze the models by MCMC simulation with the software *OpenBUGS* [17, 31]. BUGS [17], which includes WinBUGS, OpenBUG and *JAGS*, exploits the conditional independence encoded in DAGs to simplify the expression of the *full conditional distributions* for MCMC, though the DAGs do not appear explicitly in the BS formulation.

3.3 Prediction

In both BN and BS methods, this paper pays special attention to the prediction problem. Given a dataset of observations of a random variable, the first step for the prediction is to divide the dataset into a *training set* and a *test set*. A model for the prediction is built by fitting it on the training set. The model provides predicted values of data in the test set; then, the accuracy is checked by comparison between the predicted values and the corresponding observed ones in the test set. The four steps (the dataset partition, model fitting, prediction and accuracy check) are repeated by changing the training and test sets. Overall accuracy is evaluated as the average of individual ones. It is apparent that the accuracy depends on how the original dataset is partitioned into the two parts.

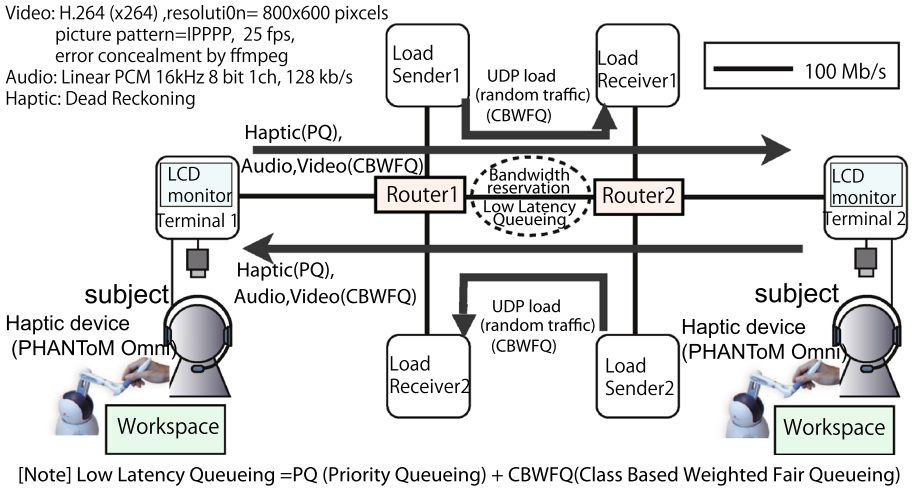
A widely used way of the partition and accuracy check is *k-fold cross-validation*, where the data are *randomly* partitioned into k subsets, and each subset is used in turn to validate the model fitted on the remaining $k-1$ subsets [13, 29, 30].

The prediction can also be considered a *missing data problem* [17], since the data in the test set are regarded as missing in the original whole dataset. *MCAR* (*missing completely at random*) and *MAR* (*missing at random*) are often assumed in prediction problems as in k-fold cross-validation. This assumption may be appropriate when we want to remove scores from the original dataset for some reasons (e.g., data are outliers, or subjects are unqualified). However, this is not necessarily an appropriate setting in real situations of QoE assessment. Data we wish to predict are not considered to be dispersed randomly over the dataset but rather to be located in blocks, since a QoE value is calculated by taking an average over a data block (i.e., a set of scores given by subjects under a certain condition). This corresponds to *MNAR* (*missing not at random*), which is a much harder situation than MCAR and MAR in the missing data problem. This paper supposes MNAR. We will see more details in Sect. 5.4.

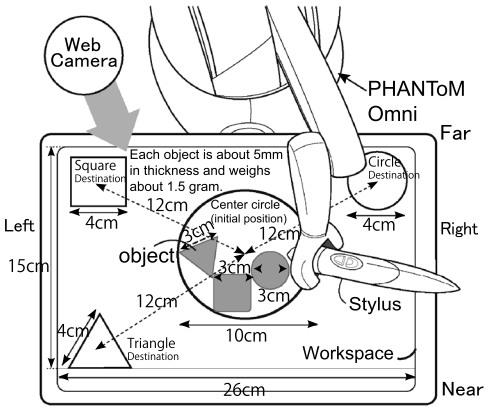
4 Haptic-audiovisual interactive communications with guaranteed bandwidth

This section illustrates the experimental system constructed for haptic-audiovisual interactive communications and a task which a pair of subjects perform for QoE evaluation [28]. The data of scores collected in the experiment are formatted into a style convenient for Bayesian modeling.

[1] The system in [21] has no such a mechanism; it is a best-effort network



(a) Experimental system



(b) Workspace for the task of object movement

Fig. 1 Experimental system configuration and the task

4.1 Experimental system

Figure 1 shows the experimental system configuration.

A pair of subjects carry out an interactive task, which is referred to as *object movement*. The task is performed in the workspace of Fig. 1b (this is not a virtual space but a *real space*). One of the subjects plays a role of the *instructor*, and the other is the *manipulator*; the procedure for the task will be described later in the next subsection.

Each terminal is equipped with a haptic device (*PHANToM Omni*; hereafter, *PHANToM* for simplicity), a video camera, an LCD monitor and a headset, and has an identical workspace.

As depicted in Fig. 1, PHANToM has a stylus with which the user can take mechanical actions on physical objects. The two terminals are located in different rooms. Each terminal transmits/receives haptic media, audio and video to/from the other terminal as *three separate UDP streams*.

PHANToM gives the user a reaction force that is proportional to the difference in the coordinate data of the stylus between its own device and the other one; thus, the two haptic devices work so that they can decrease the positional difference of the styli. This forms a force feedback loop between the two devices and allows the user to manipulate the stylus of the other side remotely by his/her own stylus.

The receiver at each terminal exerts *playout buffering control* at the application layer in order to absorb network delay jitter. The transmission unit at the application layer is referred to as a *media unit* (*MU*) in this paper. A video MU corresponds to a video frame, an audio MU is a constant number (320) of audio samples, and a haptic MU is current positional information (320 bit coordinate data) of the stylus.

Load Senders 1 and 2 transmit UDP load traffic to Load Receivers 1 and 2, respectively; they generate UDP datagrams of 1480 bytes each at exponentially distributed intervals.

Routers 1 and 2 are bandwidth controllable routers (Cisco 2811), which are connected through a 100 Mb/s full duplex Ethernet channel; the other links have the same transmission rate (100 Mb/s). The routers make bandwidth

Table 1 Specification of video, audio and haptic

		Video (H.264)	Audio (PCM)	Haptic (DR)
Packet scheduler		CBWFQ	CB-WFQ	PQ
Guaranteed band-width B (Mb/s)		$B = 1.0, 1.5, 2.0, 2.5$	0.160	0.20
Encoding bit rate R (kb/s)	$B = 1.0$	$R = 600, 750, 900$	128 (linear PCM)	Variable (DR)
	$B = 1.5$	$R = 900, 1125, 1350$		
	$B = 2.0$	$R = 1200, 1500, 1800$		
	$B = 2.5$	$R = 1500, 1875, 2250$		
Playout buffering time (ms)		$P = 40, 60, 80, 100, 150, 300$		10

reservation between the two with the *Low Latency Queueing* packet scheduling algorithm, which classifies the traffic into two kinds of classes: the *Priority Queueing* (*PQ*) class and the *Class Based Weighted Fair Queueing* (*CBWFQ*) classes. Each class has a dedicated buffer. Packets in the PQ class are served with high priority until its buffer becomes empty; then, the server goes down to the CBWFQ classes. CBWFQ can specify the minimum guaranteed bandwidth for each class. The PQ class is assigned to the haptic media, while the video, audio and UDP load are treated as three separate CBWFQ classes.

Table 1 shows specification of the video, audio and haptic media. The video guaranteed bandwidth, which is denoted by B, is set to 1.0, 1.5, 2.0, or 2.5 Mb/s. The video encoding bit rate R takes three values according to B. The audio is a 128 kb/s linear PCM with the guaranteed bandwidth of 160 kb/s.

The original haptic MU rate of PHANToM is 1 kHz; this leads to a bit rate of 320 kb/s. However, in order to reduce the bit rate, we have adopted *Position History-Based DR* (*Dead-Reckoning*) [32, 33], which is a sort of predictive encoding scheme; the DR scheme adopted here sends a haptic MU only when the difference between the predicted position and the real one exceeds a threshold value. This paper performs a linear prediction using the position and velocity; the threshold value is set to 1.0 mm. Prediction error of its own predicted position upon reception of a haptic MU is corrected gradually in ten steps. Note that the rate of updating prediction for DR and decision on MU transmission is kept at 1 kHz. The guaranteed bandwidth for the haptic media as the PQ class is 200 kb/s, which can accommodate the reduced bit rate.

The remaining bandwidth is allocated to the load traffic, whose average bit rate is set to the guaranteed bandwidth.

In this system, the *playout buffering time* at the receiver plays an important role in enhancing the performance. We have employed the *Media Adaptive Buffering* (*MAB*) scheme [21], where the playout buffering time for the haptic media is set to a shorter value than that for the audio and video streams. The playout buffering time often dominates end-to-end delay in interactive communications, and the haptic is vulnerable to long end-to-end delay, which usually increases the coordinate difference; a large difference generates a stronger reaction force, which degrades the operability of PHANToM. In this experiment, the buffering time of the haptic media is kept at 10 ms, while the audio and video streams adopt the same buffering time P, which is set to 40, 60, 80, 100, 150, or 300 ms (see Table 1).

4.2 Task of object movement

Referring to Fig. 1b, we explain the procedure for the task of object movement. At the beginning of a task, one of the two subjects is assigned to the *instructor*, and the other subject to the *manipulator*.

The following three steps are repeated during 30 s .

1. The instructor randomly selects an object out of the three in the center circle on his/her own side and tells an instruction (e.g., move the circle object to the square destination) to the manipulator using the microphone.
2. The manipulator acknowledges the instruction; then, to move the specified object on the instructor's side, he/she manipulates the instructor's PHANToM through his/her own PHANToM over the network, while watching the instructor's workspace displayed on his/her LCD monitor.
3. Once the manipulator has delivered the object to the destination, the two subjects alternate the roles.

Note that the subject at a terminal can operate not only his/her own stylus but also the stylus at the other terminal remotely. During an experiment run (i.e., 30 s), each subject holds his/her own stylus in a hand; however, the instructor's subject surrenders him/herself to the manipulator's movement not so as to impede it. In a sense, the instructor looks like a marionette remotely operated by the manipulator.

4.3 Subjective experiment

We conducted subjective experiment for collecting five-point scores for the task by recruiting 38 subjects, who are composed of 24 females and 14 males in their teens

Table 2 Definition of stimulus ID (1 through 72) and stimuli group numbers (1 through 12)

Stimulus ID	1	2	3	4	5	6	7	8	9	10	11	12
Stm grp No.	1						2					
B (bandwidth)	1.0						1.0					
R (rate)	600						750					
P (playout)	40	60	80	100	150	300	40	60	80	100	150	300
Stimulus ID	13	14	15	16	17	18	19	20	21	22	23	24
Stm grp No.	3						4					
B (bandwidth)	1.0						1.5					
R (rate)	900						900					
P (playout)	40	60	80	100	150	300	40	60	80	100	150	300
Stimulus ID	25	26	27	28	29	30	31	32	33	34	35	36
Stm grp No.	5						6					
B (bandwidth)	1.5						1.5					
R (rate)	1125						1350					
P (playout)	40	60	80	100	150	300	40	60	80	100	150	300
Stimulus ID	37	38	39	40	41	42	43	44	45	46	47	48
Stm grp No.	7						8					
B (bandwidth)	2.0						2.0					
R (rate)	1200						1500					
P (playout)	40	60	80	100	150	300	40	60	80	100	150	300
Stimulus ID	49	50	51	52	53	54	55	56	57	58	59	60
Stm grp No.	9						10					
B (bandwidth)	2.0						2.5					
R (rate)	1800						1500					
P (playout)	40	60	80	100	150	300	40	60	80	100	150	300
Stimulus ID	61	62	63	64	65	66	67	68	69	70	71	72
Stm grp No.	11						12					
B (bandwidth)	2.5						2.5					
R (rate)	1875						2250					
P (playout)	40	60	80	100	150	300	40	60	80	100	150	300

B (video guaranteed Bandwidth) = 1.0, 1.5, 2.0, 2.5 [Mb/s];
R (video encoding bit Rate) = 600, 750, 900, 1125, 1350, 1200, 1500, 1800, 1875, 2250 [kb/s];
P (playout buffering time) = 40, 60, 80, 100, 150, 300 [ms]; stimulus ID = j ($j = 1, \ldots, 72$),
Stm grp No. = stimuli group number = n ($n = 1, \ldots, 12$);
in stm grp n, $j = 6(n-1)+1, \ldots, 6n$

and twenties; they were university students. We have given each of the 38 people a subject ID with a gender mark, F (Female) or M (Male); the ID's for female are 1 through 12 and 20 through 31, while those for male are 13 through 19 and 32 through 38.

In the experiment, each subject was asked to evaluate 11 QoE measures on a *five-point quality* scale; they are three video-related ones, two haptic ones, an audio one, interstream synchronization quality between video and haptic, interactivity, communication naturalness, work difficulty, and overall satisfaction.

For simplicity of discussion, we deal only with the *overall satisfaction* in this paper. Other measures can be modeled in the same way as proposed here. The *Absolute Category Rating* (*ACR*) with the five-level quality scale was used: "excellent" = 5, "good" = 4, "fair" = 3, "poor" = 2 and "bad" = 1.

For the convenience of Bayesian modeling in the following sections, we now introduce the *stimulus ID* and *stimuli group number* which are defined in Table 2. A *stimulus* is the unit of system parameters for which QoE is assessed; it is prescribed by a combination of *B*, *R* and *P*, which gives

Table 3 Full dataset $\boldsymbol{S}_F$ of collected scores

Subject ID (gender)	Stimulus ID					
	1	2	3	…	71	72
1 (F)	5	4	4	…	4	5
2 (F)	4	3	3	…	3	3
⋮ (F)	⋮	⋮	⋮	⋱	⋮	⋮
12 (F)	4	5	4	…	5	5
13 (M)	3	3	2	…	4	2
⋮ (M)	⋮	⋮	⋮	⋱	⋮	⋮
19 (M)	2	2	3	…	4	4
20 (F)	1	2	1		3	3
⋮ (F)	⋮	⋮	⋮	⋱	⋮	⋮
31(F)	5	5	5	…	5	5
32(M)	2	4	3	…	3	3
⋮ (M)	⋮	⋮	⋮	⋱	⋮	⋮
38 (M)	4	3	1	…	5	4

1 = bad, 2 = poor, 3 = fair, 4 = good, 5 = excellent

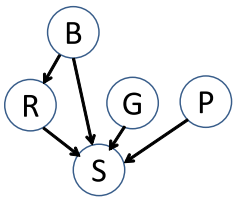


Fig. 2 *dag*: a directed acyclic graph for Bayesian network modeling

$4 \times 3 \times 6 = 72$ stimuli in total. A *stimuli group* is a set of stimuli whose values of the (B, R) pair are the same (the total number is $4 \times 3 = 12$); within a stimuli group, we can examine the effect of the playout buffering time P on QoE under a fair condition.

Each subject gives a score to every stimulus; therefore, a stimulus has 38 scores. Table 3 displays a framework of the collected scores as a 38×72 matrix, which is called the *full dataset* and is denoted by $\boldsymbol{S}_F$ in this paper. The subject ID is shown with a gender mark, F (Female) or M (Male). The full set of the scores is provided in the supplementary material as the "Full-dataset.csv" file (42452_2019_983_MOESM1_ESM.csv).

5 Bayesian network modeling

In this section, we formulate a discrete Bayesian Network (BN) model, using an R package **bnlearn** [14, 29, 30]. We also make estimation and prediction of QoE.

5.1 DAG and CPT

The five variables B, R, P, G and S are treated as nodes of a BN model. We did not utilize any *structure learning* algorithm for creating a directed acyclic graph (DAG), but we set a graph shown in Fig. 2, which we call *dag* here, considering dependence relationships among the variables.

Table 4 dhav: a data frame for parameter learning in the BN model

Data No.	Node				
	B	R	P	S	G
1	$\boldsymbol{v}_1$			5	F
2	$\boldsymbol{v}_1$			4	F
⋮	⋮			⋮	⋮
38	$\boldsymbol{v}_1$			4	M
⋮	⋮			⋮	⋮
$38(j-1)+1$	$\boldsymbol{v}_j$			s_{1j}	F
⋮	⋮			⋮	⋮
$38(j-1)+38$	$\boldsymbol{v}_j$			s_{38j}	M
⋮	⋮			⋮	⋮
2699	$\boldsymbol{v}_{72}$			5	F
2700	$\boldsymbol{v}_{72}$			3	F
⋮	⋮			⋮	⋮
2736	$\boldsymbol{v}_{72}$			4	M

We will calculate QoE by using the conditional probability table (CPT) of node S as we will see in Sect. 5.3. As already mentioned in Sect. 3, unlike [4, 19, 23, 24], this paper does not introduce utility nodes.

For estimation of CPT in each node (i.e., *parameter learning*), we make a data frame with five columns each representing the node variable as illustrated in Table 4; we refer to the data frame as *dhav* in this paper.[2] The S column of *dhav* has been constructed by reshaping the 38×72

[2] *dhav* is given in the supplementary material as the "B-R-P-S-G.csv" file (42452_2019_983_MOESM2_ESM.csv).

Table 5 Definition of notations representing QoE measures

Notation	Definition
$Q_{BN.j}$	QoE estimated by a Bayesian Network model for stimulus j, using the full dataset (no score is missing)
$Q_{BN.j}^{(\alpha,\beta)}$	QoE estimated/predicted by a Bayesian Network model for stimulus j when β scores are not available for stimulus α ($\alpha = 1, \ldots, J$; $\beta = 1, \ldots, N$; $J = 72, N = 38$ in this paper). The β subjects are specified by their ID's
$Q_{BS.j}$	QoE estimated by a Bayesian Statistical model for stimulus j, using the full dataset (no score is missing)
$Q_{BS.j}^{(\alpha,\beta)}$	QoE estimated/predicted by a Bayesian Statistical model for stimulus j when β scores are not available for stimulus α ($\alpha = 1, \ldots, J$; $\beta = 1, \ldots, N$). The β subjects are specified by their ID's
$Q_{BSNR.j}$	QoE estimated by a Bayesian Statistical model with No Random effect terms (i.e., without hierarchical priors) for stimulus j, using the full dataset
$Q_{BSNR.j}^{(\alpha,\beta)}$	QoE estimated/predicted by a Bayesian Statistical model with No Random effect terms for stimulus j when β scores are not available for stimulus α ($\alpha = 1, \ldots, J; \beta = 1, \ldots, N$)

We define a J-dimensional QoE vector whose j-th component is given in the above table;

e.g., $\boldsymbol{Q}_{BN} \triangleq (Q_{BN.1}, \ldots, Q_{BN.J})$, $\boldsymbol{Q}_{BN}^{(\alpha,\beta)} \triangleq (Q_{BN.1}^{(\alpha,\beta)}, \ldots, Q_{BN.J}^{(\alpha,\beta)})$,

$\boldsymbol{Q}_{BS} \triangleq (Q_{BS.1}, \ldots, Q_{BS.J})$, $\boldsymbol{Q}_{BS}^{(\alpha,\beta)} \triangleq (Q_{BS.1}^{(\alpha,\beta)}, \ldots, Q_{BS.J}^{(\alpha,\beta)})$

matrix of the full daset into a 2736 dimensional column vector. The notation $\boldsymbol{v}_j \triangleq (B_j, R_j, P_j)$ $(j = 1, \ldots, 72)$ denotes a vector of (B, R, P) corresponding to stimulus j; the values are given by the column of stimulus j in Table 2; e.g, $\boldsymbol{v}_{23} = (1.5, 900, 150)$. All the elements in *dhav* are regarded as discrete and non-ordered states (*levels*).

For parameter learning, **bnlearn** provides the **bn.fit** function, which supports the *maximum likelihood estimation* ("**mle**") as well as *Bayesian estimation* ("**bayes**"). The **method** argument in the **bn.fit** function determines which estimator will be used. The **mle** method calculates the estimate as an *empirical frequency* in the dataset (*dhav*). When the **method** argument is set to "**bayes**" with an optional argument **iss** (*imaginary sample size*), posterior probabilities are computed from a uniform prior over each CPT; the likelihood function of the *multinomial* distribution and its conjugate prior (the *Dirichlet* distribution) are supposed. The output of the **bn.fit** function becomes an object of class **bn.fit**.

This paper utilizes only "**bayes**", though "**mle**" will be employed for a comparison purpose of estimation accuracy. We denote the output of the **bn.fit** function by *bn.bayes* here and can produce it by a command:

bn.bayes = bn.fit(dag, data=dhav, method="bayes", iss=10)

where *dag* is the object to which the function is applied, and we set iss = 10. In the current case, *bn.bayes* is not sensitive to the value of iss.

Note that the number of conditions (stimulus+gender) in CPT of node S (i.e., the number of configurations of the categories of the parents of S) is not $4 \times 3 \times 6 \times 2 = 144$ but $4 \times 10 \times 6 \times 2 = 480$, since we have assumed non-ordered states, and R takes 10 different values; therefore, the conditions include states of (B, R) pairs that are not implemented, such as $B < R$ (e.g., B = 1.0 Mb/s and R = 1125 kb/s). The **mle** method does not assign probabilities to such avoidable states, while the **bayes** method sets uniform probabilities of 1 / 5 to each element.

5.2 Definitions of QoE measures

Before delving into assessing QoE, we define notations representing QoE measures for BN modeling as well as BS modeling, which will be discussed in the next section.

Table 5 lists the notations. Although their meanings are self-explanatory, we give three examples of $Q_{BN.j}^{(\alpha,\beta)}$ in order to make them more understandable. First, $Q_{BN.55}^{(57,1:38)}$ means QoE estimated by a Bayesian Network model for stimulus 55 when scores of subjects 1 through 38 are not available for stimulus 57 (i.e., scores of all the subjects for stimulus 57 are missing). Similarly, $Q_{BN.57}^{(57,1:38)}$ means QoE for stimulus 57 which is predicted by the Bayesian Network model fitted to the dataset where all the scores of stimulus 57 are missing. Also, $Q_{BN.57}^{(57,1:19)}$ represents QoE for stimulus 57 which is predicted by the Bayesian Network model when scores of subjects 1 through 19 are not available for stimulus 57. If we want to express QoE conditional on both stimulus and gender, we put a subscript of the stimulus ID (say j) followed by the gender mark (say F) like $Q_{BN.jF}$, though this is not defined in Table 5 for simplicity.

The counterparts of BS such as $Q_{BS.j}^{(\alpha,\beta)}$ are interpreted in the same way.

We also define a J-dimensional QoE vector to express the whole set of QoE as

$\boldsymbol{Q}_{BN} \triangleq (Q_{BN.1}, \ldots, Q_{BN.J})$ and its subset for stimuli group n as

$$\boldsymbol{Q}_{BN(n)} \triangleq (Q_{BN.6(n-1)+1}, \ldots, Q_{BN.6n})$$

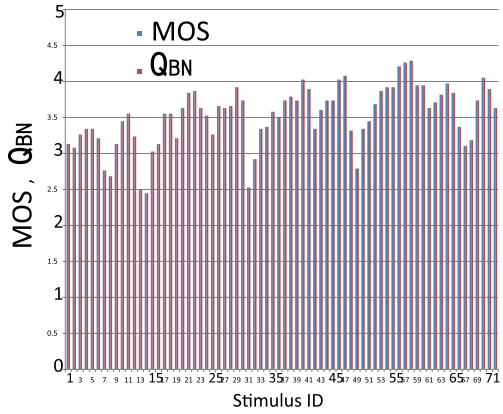


Fig. 3 MOS and $\boldsymbol{Q}_{BN}$ versus stimulus ID

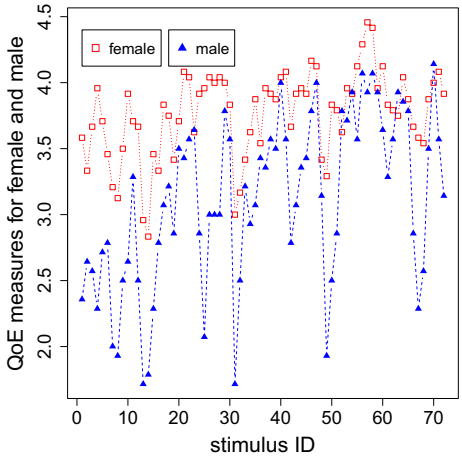


Fig. 4 $Q_{BN.jF}$ and $Q_{BN.jM}$ versus stimulus ID $= j\,(j = 1, \ldots, 72)$

5.3 QoE estimation

Let $\pi_j(k) \triangleq \Pr(S = k|j)$ denote the probability that $S = k$, conditional on stimulus j; then, $Q_{BN.j}$ is calculated as

$$Q_{BN.j} = \sum_{k=1}^{K} k \cdot \pi_j(k) \tag{2}$$

where $K = 5$ in this paper. Similarly, QoE conditional on both stimulus ID $= j$ and gender $G = g$ is expressed as

$$Q_{BN.jg} = \sum_{k=1}^{K} k \cdot \pi_{jg}(k) \tag{3}$$

where $\pi_{jg}(k) \triangleq \Pr(S = k|j, g)$.

The probabilities $\pi_j(k)$ and $\pi_{jg}(k)$ can be obtained from the CPT of node S. Detail of the derivation is described in "Appendix 1".

Note that Eqs. (2) and (3) become exactly equal to *Mean Opinion Score* (*MOS*) when the **mle** method is employed, since the conditional probability is calculated as an empirical frequency.

Figure 3 displays a barplot of MOS and $\boldsymbol{Q}_{BN}$ versus stimulus ID;[3] we see that the two kinds of the QoE measures are almost equal. The *Pearson correlation coefficient* (*PCC*) between the two is 1.0, and the *p-value* is less than 2.2×10^{-16} for the null hypothesis that the true correlation is equal to 0. The *mean square error* (*MSE*) is 7.6605×10^{-7}.

In Fig. 3, we also notice that as the stimulus ID increases, MOS and $\boldsymbol{Q}_{BN}$ rise and fall periodically at intervals of six, which comes from the six kinds of the playout buffering time P in a stimuli group (see Table 2). This is due to a tradeoff relationship between *fidelity* and *latency* in interactive communications [6].Focusing on a single stimuli group, we can recognize this feature more clearly as we will see later in Figs. 5 and 6.

We also plot $Q_{BN.jF}$ and $Q_{BN.jM} (j = 1, \ldots, 72)$ in Fig. 4; we then find that the female subjects tend to give higher scores than the male subjects. The PCC (p-value) and MSE between the estimate and MOS calculated separately for each gender are 1.0 ($p < 2.2 \times 10^{-16}$) and 5.2085×10^{-7}, respectively, for female, and 1.0 ($p < 2.2 \times 10^{-16}$) and 9.3699×10^{-7}, respectively, for male.

5.4 QoE prediction

This paper supposes two cases in which $Q_{BN.j}$ should be predicted: (a) the whole scores for stimulus j are missing; i.e., the predicted value is $Q_{BN.j}^{(j,1:38)}$, and (b) the front 19 scores out of 38 (i.e., the first half) are not available for stimulus j, i.e., $Q_{BN.j}^{(j,1:19)}$. Other cases will also be discussed in Sect. 7.2.

Case (a) is a natural and often encountered situation for QoE prediction; it is usually the case that the whole scores for a stimulus is not observed and that we want to predict the average of the scores. On the other hand, Case (b) is just a decrease of the sample size, which imposes a weaker condition on prediction. In the context of the *missing data* problem [17], Cases (a) and (b) correspond to *MNAR* (*missing not at random*). Recall that MNAR is a more difficult setting than MCAR and MAR, which the k-fold cross-validation is based on.

In the BN modeling environment settled in this paper, two methods for QoE prediction are available: (1) the **querygrain** function, which is *exact inference* we utilized

[3] R codes for plotting Figs. 3 and 4 are given in the supplementary material as File 3 (42452_2019_983_MOESM3_ESM.pdf) (see the README.pdf file (42452_2019_983_MOESM11_ESM.pdf) for the definition of the file numbers).

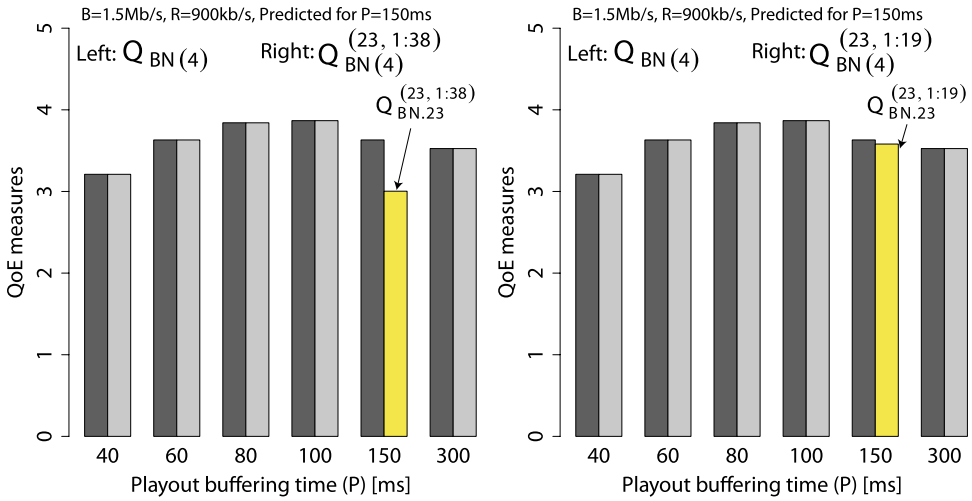


Fig. 5 QoE BN-prediction of stimulus 23 in stimuli group 4

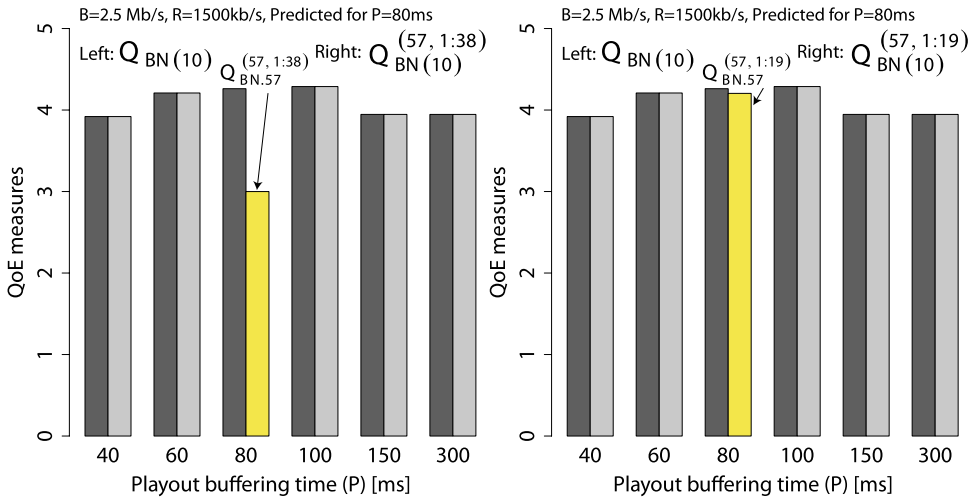


Fig. 6 QoE BN-prediction of stimulus 57 in stimuli group 10

for QoE estimation in the previous subsection, and (2) the **predict** function in **bnlearn**, which has alternatives of *method = "parents"* and *method = "bayes-lw"*. These three ways of the prediction usually provide very close values to each other. In this paper, we mainly employ the predict function with method = "bayes-lw", which does *approximate inference*, for the sake of consistency with later discussions on *cross-validation*, where brief comparison with prediction by the querygrain will be made.

The usage of the predict function is as follows:

predict(object, node, data, method = "bayes-lw", prob = TRUE)

where the argument **object** is an object of class **bn.fit**, **node** is the label of the target node (node = "S" in the current case), **data** is a data frame containing the data to be predicted, i.e., a *test set* (complete observation will be ignored), **method** instructs that the predicted values are computed by averaging *likelihood weighting* simulations [13, 14] performed using all the available nodes as evidence, and **prob** is a boolean value; if **prob** = TRUE and **object** is a discrete network, the probabilities used for prediction are attached to the predicted values as an attribute called **prob** [29].

The **predict** function returns a *data frame with the same structure* as **data**. It should be noted that it returns the predicted values, which we denote by *predicted* here, each with the highest conditional probability. More precisely in the current case, it returns a single score value (from among five possible values "1" through "5") for each variable being predicted and ignores score values with lower probabilities; e.g., predicted = "5", which has a probability of 0.35, while "1" with 0.0436, "2" with 0.2601, "3" with 0.1729 and "4" with 0.1734 are ignored. Consequently, we cannot calculate the predicted value of QoE (the average of scores) with the returned value by itself; for the calculation of the QoE measure, we have to extract the probabilities of scores as an attribute by a command **attributes(predicted)$prob**.

Table 6 Custom-folds cross-validation: posterior classification error (PCError), Pearson correlation coefficient (PCC) along with the *p-value* and mean square error (MSE)

Subset size (unit of split)	PCError	PCC (*p*-value)		MSE
38 (a stimulus)	0.79569	"predict"	0.14633 ($p = 0.22$)	0.44537
		"querygrain"	Not applicable	0.44572
19 (1 / 2 stimulus)	0.70249	"predict"	0.95632 ($p < 2.2 \times 10^{-16}$)	0.01588
		"querygrain'	0.95663 ($p < 2.2 \times 10^{-16}$)	0.01579
38 (72-fold c.v.) (random split of scores)	0.71294	"predict"	0.87497	0.05456
		"querygrain"	0.99874	0.000444

We now exteriorize the way of predicting $Q_{BN.j}^{(j,1:m)}$ where $m = 19$ or 38.

As shown in "Appendix 2", $Q_{BN.j}^{(j,1:m)}$ is calculated as

$$Q_{BN.j}^{(j,1:m)} = \sum_{i=1}^{m} \sum_{k=1}^{5} k \cdot \phi_{i.j}^{(j,1:m)}(k)/m \tag{4}$$

where $\phi_{i.j}^{(j,1:m)}(k)$ is defined as the predicted probability that $S = k$ for stimulus j by subject i ($i = 1, \ldots, m$)

We can evaluate $Q_{BN.n}^{(j,1:m)}$ ($n \neq j$) by using bn.bayes.j, which is defined in "Appendix 2", in the same way as that of QoE estimation.

In the following numerical examples, we pick up stimulus ID's of 23 and 57. This selection is arbitrary and has no special meaning. Other stimuli can be treated in the same way by referring to R codes in the supplementary material (File 4 (42452_2019_983_MOESM4_ESM.pdf) for the left panels of Figs. 5 and 6, and File 5 (42452_2019_983_MOESM5_ESM.pdf) for the right panels of the two figures).

5.4.1 Case (a) for stimulus 23

This is a problem of predicting the QoE measure for stimulus 23 when all the subjects' scores for the stimulus are not available: $Q_{BN.23}^{(23,1:38)}$.

The left panel of Fig. 5 plots $\boldsymbol{Q}_{BN(4)}^{(23,1:38)}$, which contains $Q_{BN.23}^{(23,1:38)}$ as a member of the stimuli group 4, along with $\boldsymbol{Q}_{BN(4)}$ for a comparison purpose. We find that the predictive accuracy is low. This is because $\phi_{i.23}^{(23,1:38)}(k)$ has been predicted to be uniform over $K = 5$, namely, a probability around 0.2, which produces an average score of about 3.0 for all subjects.

Also, note that there exists the optimum value of P in the sense that it maximizes QoE; $P = 100$ ms in Fig. 5. This is a feature of interactive communications.

5.4.2 Case (b) for stimulus 23

We can improve the predictive accuracy if we increase the sample size for stimulus 23 from zero to 19. The right panel of Fig. 5 plots $\boldsymbol{Q}_{BN(4)}^{(23,1:19)}$, to which $Q_{BN.23}^{(23,1:19)}$ belongs. We can confirm the improvement in the predictive accuracy.

5.4.3 Cases (a) and (b) for stimulus 57

$Q_{BN.57}^{(57,1:38)}$ and $Q_{BN.57}^{(57,1:19)}$ are displayed in Fig. 6, which demonstrates the same property as that of stimulus 23.

From the examples above, we see that the BN model requires some *nonzero* number of observations for good prediction; otherwise, the prediction is bad (always about 3.0).

5.5 Custom folds cross-validation

We next extend the discussions in Cases (a) and (b) in the previous subsection by means of *custom folds cross-validation* [29], where the data are manually partitioned by the user into subsets, which are then used as in k-fold cross-validation. Case (a) corresponds to a subset size of 38, which means $k = 2736/38 = 72$ folds. Similarly, Case (b) has a size of 19 and therefore $k = 144$.

For cross-validation, **bnlearn** provides the **bn.cv** function, which can perform the Bayesian custom-folds cross-validation by setting **method** = "custom-folds", **fit** = "bayes" and **loss** = "pred-lw".

The **bn.cv** function makes prediction of scores in each *test subset* by utilizing the model fitted to the remaining subsets which collectively constitute the *training set*: 71 subsets in Case (a) and 143 ones in Case (b). It then returns the *Posterior Classification Error* (*PCError*) of scores by comparing predicted scores and the corresponding observed ones.

We have carried out the two kinds of custom folds cross-validation: subset sizes of 38 and 19.[4] The splits were performed in units of a stimulus or half a stimulus. Table 6 shows the Pearson correlation coefficient (PCC) along with the *p*-value and the mean square error (MSE) between predicted QoE values and estimated QoE values (i.e., without missing scores) in addition to the Posterior Classification Error (PCError) returned by the **bn.cv** function. Note that the classification error is a criterion for individual scores,

[4] The R codes for these two kinds of custom folds are given in the supplementary material as Files 4 and 5 (42452_2019_983_MOESM4_ESM.pdf, and 42452_2019_983_MOESM5_ESM.pdf).

while the PCC and MSE are the ones for the average of scores (i.e., QoE).

As PCC and MSE, Table 6 presents two kinds of the values for each subset size: "predict" and "querygrain". The former means prediction by the predict function with method = "bayes-lw", which we have utilized so far in this paper; the latter is the one by the querygrain function.

The number of predicted QoE values in the case of subset size 38 is 72, each of which has been evaluated for each fold. On the other hand, the number of prediction for the subset size of 19 is 144; therefore, in order to ensure one-to-one correspondence between predicted QoE and estimated QoE (namely, QoE estimation without missing scores), we used the estimated QoE values twice in the calculation of PCC and MSE.

In Table 6, we first find that although both subset sizes produce large classification errors, the size of 19 achieves good correlation coefficients of about 0.956 in stark contrast to the PCC's for size 38: 0.14633 by "predict" and *Not Applicable* by "querygrain". The result of "querygrain" comes from the zero standard deviations since its predicted values are always 3.0. Furthermore, the PCC value of 0.14633 by "predict" is not statistically significant at a significance level of 0.05 (i.e, the null hypothesis that the true correlation is equal to 0 is not rejected), whereas the one for size 19 is significant at a significance level of 0.05. The observation above is consistent with Figs. 5 and 6.

Furthermore, we have examined 72-fold cross-validation[5] in which the data are randomly split into 72 subsets (of size 38 each) in units of scores; this implies that the missing data number scatters over 2736 rows of *dhav* in Table 4.

The bottom row of Table 6 demonstrates the result of the 72-fold cross validation. The predicted QoE value of a stimulus for "predict" has been calculated in a similar way to that in Sect. 5.4 if the stimulus belongs to the test subset; otherwise, we evaluated it by utilizing conditional probabilities obtained by the querygrain function applied to the training set consisting of the remaining 71 subsets. We have thus collected 72 predicted QoE values in total. In the case of "querygrain", all the p™redicted QoE values have been derived by the querygrain function regardless of the stimulus affiliations with the test subset. The process was repeated ten times. The result is the average over them.

As expected, the 72-fold cross validation realizes high correlation coefficients of 0.87497 for "predict" and 0.99874 for "querygrain" in spite of a large PCError of 0.71294; this is due to the MCAR property.

Note that the **bn.cv** function does not work well for checking the accuracy of predicted QoE values because it returns only a single score value with the highest probability, while ignoring the other scores, as already mentioned. Scores given by the same subject even under a constant condition usually fluctuate and therefore comparison between observed and predicted score values has no intrinsic meaning. We have noticed that *k*-fold cross-validation (i.e., randomly split), which is often used for checking predictive accuracy in BN modeling, is a slack condition for QoE prediction.

6 Bayesian statistical modeling

We now turn our attention to Bayesian Statistical (BS) modeling. In this section, we build an *ordinal regression model* having the mean of a continuous *latent variable* underlying *S* as the *response variable* and *B*, *R*, *P* and *G* as *explanatory variables* (*covariates*). The key here is that although the ordinal data of scores are discrete by nature, we represent them as indicators of a continuous variable underlying the scores. The QoE measure for stimulus *j* is expressed such as $Q_{BS,j}$ as already defined in Table 5.

We could formulate other BS models, especially ones without resort to a latent continuous variable, in the current case. However, we have decided to adopt the above model since it can effectively utilize the information on the subjects' individualities in the given dataset $\boldsymbol{S}_F$. If we used $\boldsymbol{S}_F$ only as a score frequency table by aggregating scores in a stimulus (i.e., counting the frequency of each score in a stimulus), which produces five rows without the "G" column instead of 38 rows per stimulus in Table 4, the information on the subjects' individualities would be lost; this incurs degradation of QoE estimation accuracy.

6.1 BS model

The model is built in a similar way to the one proposed in [6]. Let s_{ij} denote the random variable representing score *S* for stimulus *j* by subject *i* with probability $\pi_{ij}(k) \triangleq \Pr(s_{ij} = k|i,j)$ $(i = 1,\ldots,N; j = 1,\ldots,J; k = 1,\ldots,K;$ $N = 38, J = 72, K = 5)$. We then assume that it has a categorical distribution:

$$s_{ij} \sim \text{Categorical}\,(\pi_{ij}(1),\ldots,\pi_{ij}(K)) \tag{5}$$

This implies that s_{ij} takes one of *K* mutually exclusive outcomes $1,\ldots,K$ with probabilities $\pi_{ij}(1),\ldots,\pi_{ij}(K)$, respectively, where $\pi_{ij}(1) + \cdots + \pi_{ij}(K) = 1$.

The data $s_{i1},\ldots,s_{iJ}$ are supposed to be independent outcomes on an ordinal scale $1, 2, \ldots, K$. Even if subject *i* selects the same score *k* (say) for two different stimuli j_1 and j_2 (i.e., $s_{ij_1} = s_{ij_2} = k$), he/she can have slightly different satisfaction with the experience between the two stimuli because of

[5] The R code is presented as File 6 (42452_2019_983_MOESM6_ESM.pdf) in the supplementary material.

many reasons (e.g., his/her mental and physical conditions at the measurement time). This is also true when two subjects i_1 and i_2 select the same score k owing to their individualities. This suggests that the same score does not imply an exactly equal satisfaction but it has some fluctuations. Therefore, it is reasonable to consider that scores $1, 2, \ldots, K$ indicate K *categories* of satisfaction each of which has an interval with some probability distribution.

We can incorporate the above observation into the model by introducing a continuous *latent variable* z_{ij}^* underlying S, which can take any real value. Then, the K categories for z_{ij}^* are regarded as K intervals $[\kappa_{j0}, \kappa_{j1}]$, $(\kappa_{j1}, \kappa_{j2}]$, …, $(\kappa_{j,K-1}, \kappa_{jK}]$, where $\kappa_{j0} = -\infty$ and $\kappa_{jK} = \infty$. The k-th *cut point* for stimulus j, κ_{jk}, is a random variable with the constraint of the ordinal relations specified above. Thus, s_{ij} is set to k if $\kappa_{j,k-1} < z_{ij}^* \le \kappa_{jk}$; $k = 1, \ldots, K$.

Furthermore, we suppose that z_{ij}^* follows a *logistic* distribution with mean μ_{ij} and the residual error ε_{ij} with zero mean (i.e., $z_{ij}^* = \mu_{ij} + \varepsilon_{ij}$). Then, the *probability distribution function* of ε_{ij} is expressed as $F(\varepsilon_{ij}) = \text{logistic}(\varepsilon_{ij}) = 1/\{1 + \exp(-\varepsilon_{ij})\}$. We thus have

$$p_{ij}(k) = 1/\{1 + \exp(\mu_{ij} - \kappa_{jk})\}; \quad k = 1, \ldots, K-1 \tag{6}$$

where $p_{ij}(k)$ is the *cumulative probability* of s_{ij} being k or lower such that $p_{ij}(k) \triangleq \Pr(s_{ij} \le k | i, j)$ $k = 1, \ldots, K-1$ and $p_{ij}(K) = 1$

The probability $\pi_{ij}(k)$ is given as $\pi_{ij}(1) = p_{ij}(1)$ and $\pi_{ij}(k) = p_{ij}(k) - p_{ij}(k-1)$ for $k = 2, \ldots, K$.

We next regress μ_{ij} on B, R, P and G in order to reflect the effects of these covariates on μ_{ij}. We perform *standardization* of covariates B, R and P, i.e., each covariate minus its own mean and divided by its own standard deviation [16], in order to suppress the variation in the magnitude among the covariates.

Letting *mean*(X) denote the *empirical mean* of covariate X and *sd*(X) the *unbiased standard deviation* (square root of *unbiased variance*) of X, we assume

$$\begin{aligned}\mu_{ij} =& b_1 \cdot (B_j - mean(B))/sd(B) \\ &+ b_2 \cdot (R_j - mean(R))/sd(R) \\ &+ b_3 \cdot (P_j - mean(P))/sd(P) \\ &+ b_4 \cdot G_i \cdot (B_j - mean(B))/sd(B) \\ &+ b_5 \cdot G_i \cdot (P_j - mean(P))/sd(P) \\ &+ u_i + v_{G(i)}\end{aligned} \tag{7}$$

where B_j, R_j and P_j are the values of B, R and P, respectively, for stimulus j, G_i is the gender of subject i (1 for female and 2 for male), and $\{b_1, b_2, b_3, b_4, b_5\}$ are regression coefficients, u_i expresses a *random effect* due to subject i, and $v_{G(i)}$ a random effect of subject i at the gender-level.

As in [6], this paper takes *noninformative priors* of the normal distribution with zero mean: $b_n \sim \text{N}(0, 10^2)$ (the variance $= 10^2$, $n = 1, \ldots, 5$), $u_i \sim \text{N}(0, s_1^2)$ and $s_1 \sim \text{Uniform}(0, 10^4)$, which is a *hierarchical prior*. We further specify $v_{G(i)} \sim \text{N}(0, s_2^2)$ and $s_2 \sim \text{Uniform}(0, 10^4)$. Priors of the cut points are set as in [6]: $\kappa_{j1} \sim \text{N}(0, 1)$, $\kappa_{jk} = \kappa_{j,k-1} + \Delta_{jk}$, and $\Delta_{jk} \sim \text{Exponential}(1)$ for $k = 2, \ldots, K-1$, where the mean is 1. Note that this setting assures the cut points satisfying the ordinal relations.

In order to estimate the joint posterior probability density of $(\kappa_{j1}, \kappa_{j2}, \kappa_{j3}, \kappa_{j4})$ and $(b_1, b_2, b_3, b_4, b_5)$ for given i and j, we fit Eqs. (5)–(7) to the full dataset $\boldsymbol{S}_F$. We can then obtain its marginal posterior probability density of each parameter, which provides estimates of $\{p_{ij}(k)\}$ and $\{\pi_{ij}(k)\}$; thus, we get $Q_{BS,j}$ as

$$Q_{BS,j} = \sum_{i=1}^{N} \sum_{k=1}^{K} k \cdot \pi_{ij}(k)/N \tag{8}$$

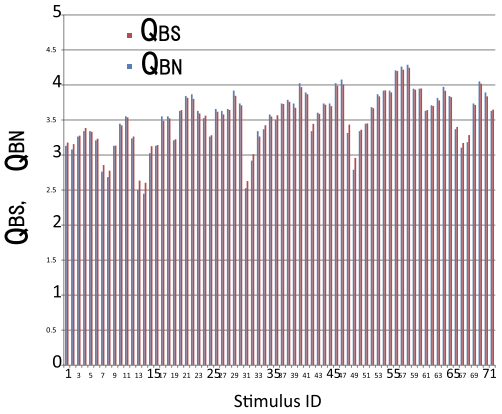


Fig. 7 $\boldsymbol{Q}_{BS}$ and $\boldsymbol{Q}_{BN}$ versus stimulus ID

6.2 QoE estimation

We carried out *Markov chain Monte Carlo* (*MCMC*) simulation of the BS model by the software *OpenBUGS* [17, 31]. Setting thinning to 5, we run three chains for a 21,000-iteration period following a 1000-iteration burn-in and confirmed convergence of the simulations (the *Brooks–Gelman–Rubin* (*BGR*) *diagnostic* $\hat{R}$ is very close to 1). The *DIC* (*Deviance Information Criterion*) is 6953.0, and the effective number of parameters p_D is 285.8.

In addition, we performed MCMC simulation of a Bayesian model without the interaction terms in Eq. (7) (i.e., $b_4 = 0$, $b_5 = 0$) and found DIC = 6978.0, which is larger than 6953.0; therefore, the model with the interaction terms is better.

Figure 7 shows a barplot of $\boldsymbol{Q}_{BS}$ and $\boldsymbol{Q}_{BN}$ versus stimulus ID. We see that the BS model provides QoE estimates very close to those by the BN model; the Pearson correlation coefficient between the two is 0.995763, and the p-value

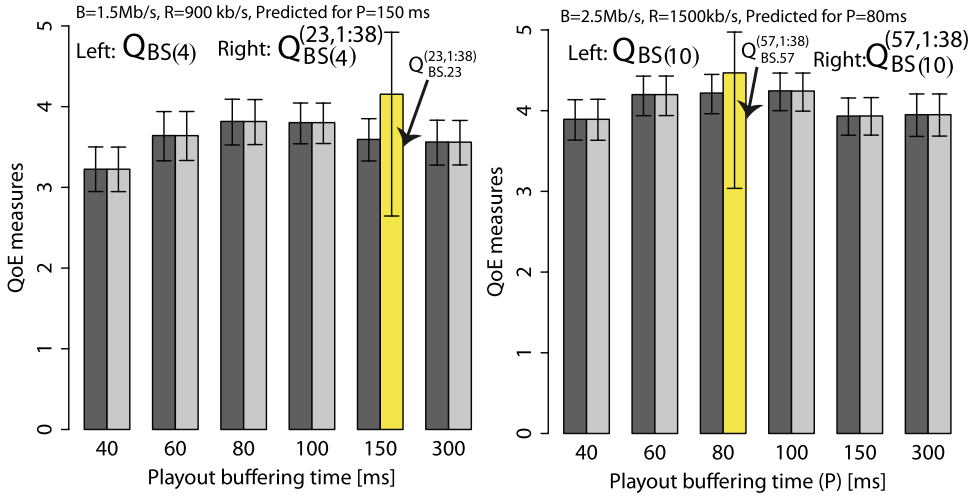


Fig. 8 QoE BS-prediction of stimuli 23 and 57 for 38 missing scores

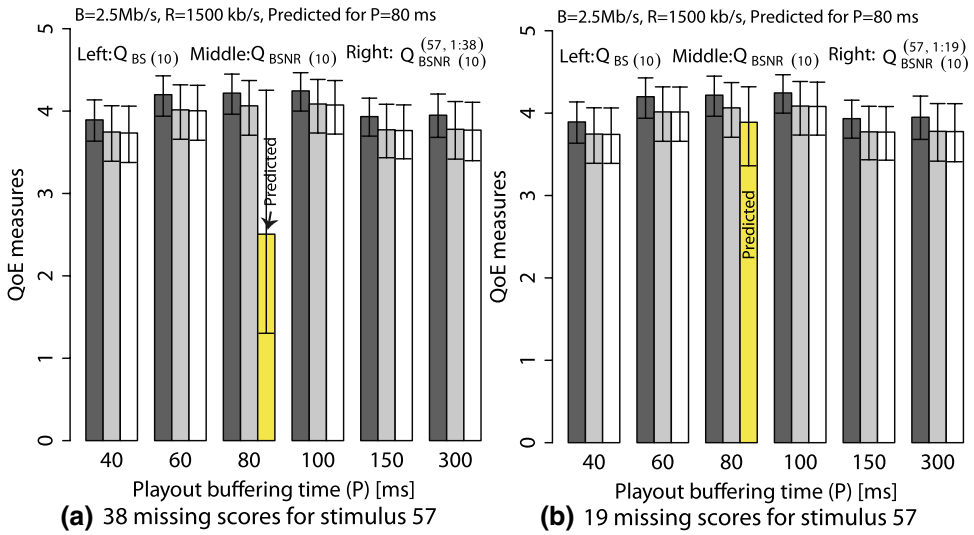


Fig. 9 QoE BS-prediction with No Random effect terms for stimulus 57

(a) 38 missing scores for stimulus 57 **(b)** 19 missing scores for stimulus 57

is less than 2.2×10^{-16} for the null hypothesis that the true correlation is equal to 0.

6.3 QoE prediction

In MCMC simulation of BS models, the prediction of score is easily made by forming a *fitting dataset* (i.e., the training set), which is obtained by replacing the values to be predicted with "NA" in the full dataset $\boldsymbol{S}_F$ [6]. We can make the prediction just by fitting the BS model of Eqs. (5)–(7) to the fitting dataset. As an example, the OpenBUGS code for evaluating $\boldsymbol{Q}^{(57,1:38)}_{BS.57}$ is given as File 7 (42452_2019_983_MOESM7_ESM.pdf) in the supplementary material.

In this way, we evaluated $\boldsymbol{Q}^{(23,1:38)}_{BS.23}$ and $\boldsymbol{Q}^{(57,1:38)}_{BS.57}$. Figure 8 illustrates the predicted values with the *95% credible intervals* along with the estimated QoE values for stimuli groups 4 and 10. Comparing this figure with the left panels of Figs. 5 and 6, we notice that the BS model provides better prediction than the BN model under the condition of the 38 missing scores. Recall that the BN model always provides about 3.0 as the predicted QoE value for a stimulus with 38 missing scores.

Now let us consider how we have achieved such better prediction with the BS model. Comparing the BN and BS models, we easily notice that a salient difference is the random effect terms u_i and $v_{G(i)}$ in Eq. (7). Removing these terms from Eq. (7), we can confirm the effectiveness of the terms in prediction; we will see this in the next subsection.

6.4 BS model without random effect terms: BSNR

We modify the regression equation of μ_{ij} by removing u_i and $v_{G(i)}$ in Eq. (7). For this BSNR model, the DIC under the same simulation condition as that in Sect. 6.2 is 8373.0, which is much larger than 6953.0. We can make estimation and prediction of the QoE measure $Q_{BSNR.j}$ in the same way as that in the previous subsection.

Figure 9 plots $Q^{(57,1:38)}_{BSNR.57}$ and $Q^{(57,1:19)}_{BSNR.57}$ as parts of $\boldsymbol{Q}^{(57,1:38)}_{BSNR(10)}$ and $\boldsymbol{Q}^{(57,1:19)}_{BSNR(10)}$, respectively. As in the BN model, no observa-

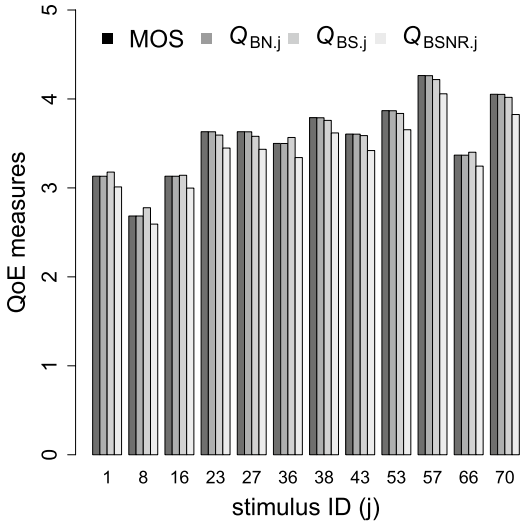


Fig. 10 MOS and estimated QoE values with no missing score: BN, BS and BSNR models

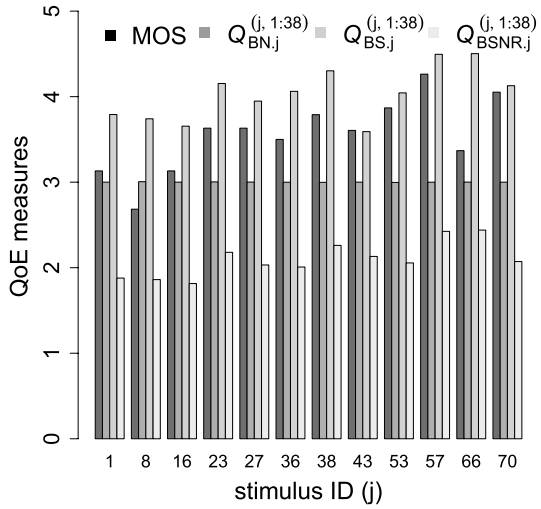


Fig. 11 MOS and predicted QoE values with 38 missing scores for each stimulus: BN, BS and BSNR models

Table 7 Pearson correlation coefficient (PCC) and mean square error (MSE) between MOS and estimates by BN, BS, and BSNR models

Pairs compared	PCC (p-value)	MSE
MOS versus BN	1.0 ($p < 2.2 \times 10^{-16}$)	7.660×10^{-7}
MOS versus BS	0.9957622 ($p < 2.2 \times 10^{-16}$)	0.003407
MOS versus BSNR	0.9990471 ($p < 2.2 \times 10^{-16}$)	0.028088
BN versus BS	0.995763 ($p < 2.2 \times 10^{-16}$)	0.003385

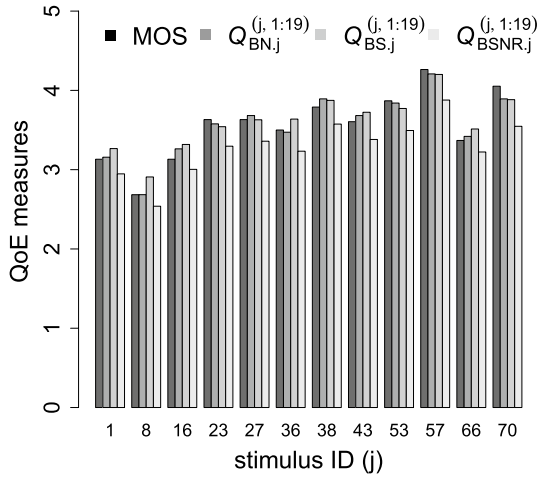


Fig. 12 MOS and predicted QoE values with 19 missing scores for each stimulus: BN, BS and BSNR models

tion of score at all for the stimulus largely degrades the prediction, while the increase of the observation size for the prediction from 0 to 19 (the decrease of the missing data from 38 to 19) has improved the predictive accuracy. This change from the BS model is clearly due to the lack of the random effect terms. This inadequacy of the model appears in the DIC value.

7 Comparison of BN and BS models

7.1 QoE estimation and prediction

We now compare the BN, BS and BSNR models in terms of estimation and prediction accuracy.[6]

We first plot MOS and QoE values with no missing score estimated by the BN, BS and BSNR models as a function of the stimulus ID in Fig. 10. As expected, we observe that MOS and QoE values by BN and BS are very close to each other, while QoE values by BSNR are slightly lower than the others. Table 7 presents the Pearson correlation coefficient (PCC) along with the p-value and mean square error (MSE) between MOS and the estimate; this supports the observations mentioned above.

In addition, we have performed a *Wilcoxon signed rank test* [34] between MOS and BSNR under the null hypothesis that the median of the difference (MOS minus the BSNR's QoE) equals zero, using R's *wilcox.exact* with paired = TRUE. We then found that the p-value is 1.659×10^{-13} (the alternative hypothesis is two-sided); thus, the null hypothesis is rejected. The value of the test statistic (denoted as V)

[6] File 8 (42452_2019_983_MOESM8_ESM.csv) in the supplementary material is a csv file that summarizes QoE values for the comparison; this is used for plotting Figs. 10, 11 and 12 by the R code of File 10 (42452_2019_983_MOESM10_ESM.pdf).

Table 8 Pearson correlation coefficient (PCC) and mean square error (MSE) of 12 predictions by BN, BS and BSNR models (stimulus ID = 1, 8, 16, 23, 27, 36, 38, 43, 53, 57, 66, 70)

Pairs compared	PCC (p-value)	MSE
MOS versus BN.NA38	-0.05919 ($p = 0.6214$)	0.4471
MOS versus BS.NA38	0.5777 ($p = 0.04918$)	0.3453
MOS versus BSNR.NA38	0.6233 ($p = 0.03036$)	2.2356
MOS versus BN.NA19	0.9641 ($p < 2.2 \times 10^{-16}$)	0.01578
MOS versus BS.NA19	0.9756 ($p = 6.48 \times 10^{-8}$)	0.01784
MOS versus BSNR.NA19	0.9809 ($p = 1.922 \times 10^{-8}$)	0.08261
BN.NA19 versus BS.NA19	0.9854 ($p = 5.14 \times 10^{-9}$)	0.009456

is 2628, which is equal to the sum of 1 through 72; this means that the MOS is larger than the BSNR's QoE value for all the stimuli.

The p-value by the Wilcoxon signed rank test between MOS and BS is calculated to be 0.9262 (therefore, zero median of the difference), while the one between MOS and BN is 2.742×10^{-13}. The MOS is slightly larger than the BN estimate for many stimuli; the value of the test statistic is 2616.

We next compare MOS and QoE predicted by BN, BS and BSNR. In the comparison, we were faced with a difficulty: a large amount of computational time of the BS and BSNR models because of MCMC simulation. For instance, it took about 9 h 16 min to get a set of 72 predictions of the $Q_{BS.j}^{(j,1:38)}$ and $Q_{BS.n}^{(j,1:38)} (n \neq j)$ by a PC with an Intel Core i7-4770 CPU at 3.40 GHz and a 32 GB main memory.[7] Therefore, we did not compare all the stimuli (i.e., the 72 stimuli) but picked up a stimulus from each stimuli-group, i.e., 12 stimuli (samples) in total, so that each value of the playout buffering time P is selected twice in the samples; the ID's of the stimuli thus picked up are 1, 8, 16, 23, 27, 36, 38, 43, 53, 57, 66 and 70.

Figure 11 displays MOS and the three kinds of QoE prediction in the case of 38 missing scores (i.e., when the whole scores for a stimulus are missing). Note that although $Q_{BS.j}^{(j,1:38)}$ varies according to stimulus ID as in MOS, $Q_{BN.j}^{(j,1:38)}$ is always very close to 3.0 for all the stimuli; this implies that the BN model does not make QoE prediction for a stimulus when no score for the stimulus is available at all.

The case of 19 missing scores is demonstrated in Fig. 12, where we notice that $Q_{BN.j}^{(j,1:19)}$ and $Q_{BS.j}^{(j,1:19)}$ can provide good predictions close to MOS.

In order to examine accuracy of the predictions in Figs. 11 and 12 quantitatively, we have calculated the PCC and MSE between MOS and prediction, which are shown in Table 8, where ".NA38" attached to a model name like "BS.NA38" means that 38 scores are missing for each stimulus in the model; ".NA19" has the implication of 19 missing scores.

In Table 8, we first notice that the PCC between MOS and BN.NA38 is very small and that the null hypothesis (the true correlation is zero) cannot be rejected at a significance level of 0.05. We also see that in the case of 38 missing scores, BS.NA38 is the best predictor among the three (BN, BS and BSNR) since its MSE is the smallest, though its PCC is slightly smaller than that of BSNR. Also, the PCC of BS.NA38 is statistically significant; the null hypothesis is rejected at a significance level of 0.05 because of $p = 0.04918$.

In the case of 19 missing scores, on the other hand, all the three models provide high correlation with MOS. In particular, the BN and BS models achieve much smaller values of MSE than BSNR; therefore, the two models can be regarded as comparable. In order to confirm this finding, we performed a Wilcoxon signed rank test between BN.NA19 and BS.NA19 under the null hypothesis that the distributions of the two kinds of predicted values differ by a location shift of zero. As a result, we obtained $p = 0.3013$; therefore, the median of the difference can be regarded as zero (i.e., the null hypothesis cannot be rejected).

7.2 Effect of the number of missing scores

So far, we have studied only two cases of missing scores: the whole scores of a stimulus (namely, 38) and the first half (19). We now examine the effect of the number of missing scores on the prediction in more detail, which is illustrated in Fig. 13 by utilizing the R code of File 10 (42452_2019_983_MOESM10_ESM.pdf) with the dataset of File 9 (42452_2019_983_MOESM9_ESM.csv) in the supplementary material.

The left panel of Fig. 13 illustrates $Q_{BN.23}^{(23,1:m)}$ and $Q_{BS.23}^{(23,1:m)}$ versus the number of missing scores m (scores from subjects 1 through m) for stimulus 23. For a comparison purpose, we also plot MOS for stimulus 23, which is equal to 3.6316. We then notice that the BS provides better approximations to MOS than the BN. In fact, the MSE between $Q_{BS.23}^{(23,1:m)}$ and MOS is calculated to be 0.06153, while the one between $Q_{BN.23}^{(23,1:m)}$ and MOS is 0.1892. Furthermore, we find that $Q_{BN.23}^{(23,1:m)}$ becomes much lower than MOS for m being 31 or larger, in which case scores only by male are available (see Table 3), and male QoE values are generally lower than the female ones (a sort of *selection bias*) as already seen in Fig. 4. This decrease in QoE values does not occur in the BS model owing to the random effect terms.

The case of stimulus 57 (MOS = 4.2632) is shown in the right panel of Fig. 13, where we see a very similar result

[7] The corresponding time by the BN model is less than 30 s.

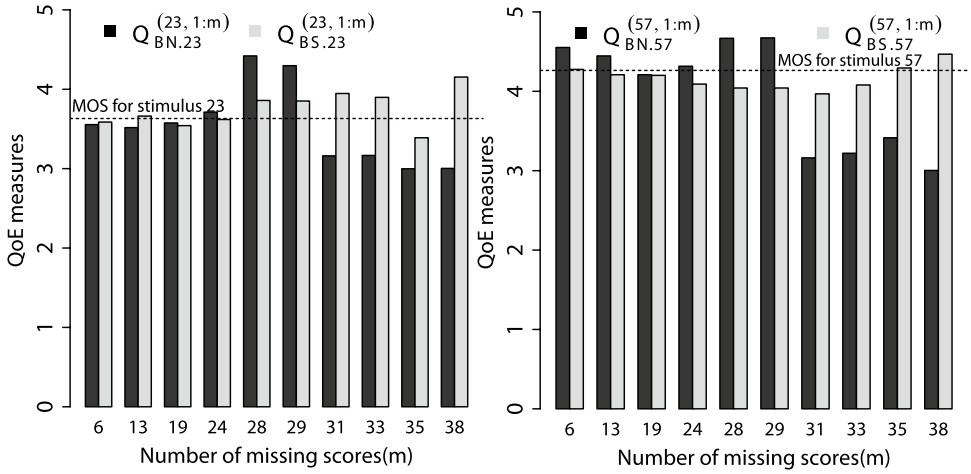


Fig. 13 Effect of the number of missing scores (from subjects 1 through *m*) on predicted QoE for stimuli 23 (left) and 57 (right)

to that of stimulus 23. The MSE is 0.4261 between BN and MOS , and 0.02980 between BS and MOS.

7.3 Considerations on comparison results

We have thus found that the BS model with the random effect terms is superior in predictive capability to the BN model as well as the BS model without the terms. The predictive power of the hierarchical model (the BS model with the random effect terms) comes from *borrowing of strength* across stimuli [16, 17]. This suggests that the incorporation of the random effects into the BN model can improve its predictive ability; this is left as future work.

Furthermore, it is fair and necessary to emphasize the main disadvantage of the BS model in this paper: a large amount of computational time compared with the BN model. The BN model has utilized a *closed form* posterior distribution with the multimonial likelihood and its conjugate prior (Dirichlet), which largely alleviates the computational burden. On the other hand, the BS model employed nonconjugate priors and MCMC simulations of *full conditional distributions*, which spend much time. Also, as the probability prediction algorithm, the BN model resorted to *likelihood weighting*, which works very fast compared with MCMC. The difference in computational time is not necessarily due to the difference in principle between BN and BS models but has stemmed from the two ways of computing the posterior distributions.

It is rather difficult to draw a clear-cut solution to the usage problem of the BN and BS methods only from the results found so far in this paper. As far as the BN and BS models built in this paper are concerned, it is a good idea to employ the BN model when we want only estimates of QoE because of short computational time; if we desire to predict QoE, the BS model with the random effect terms is indispensable.

It should be emphasized again that the BN and BS models built here are just simple examples because of a first trial on comparative study of the two Bayesian approaches. More elaborate models could be constructed, especially for BN. For example, *hybrid Bayesian networks*, which accommodate both discrete and continuous variables, can be handled with the use of MCMC (e.g., by JAGS) [14]; in this case, the difference between BN and BS is not clear.

8 Conclusions

We built a discrete Bayesian Network (BN) model and Bayesian Statistical (BS) regression models with/without random effect terms due to subjects. The BN and BS models were analyzed by an R package **bnlearn** and an MCMC software OpenBUGS, respectively. We compared the three models from a viewpoint of QoE estimation and prediction. As a result, we first found that the three models are comparable with respect to estimation accuracy. Regarding the QoE prediction ability, however, we noticed that the BS model with the random effect terms, which is a hierarchical model, is the best, while the other two models provide poor accuracy, when scores for a stimulus to be predicted are not available at all. When some scores for the stimulus are made available, the predictive accuracy of the two models improves.

The predictive accuracy is deeply concerned with the *missing data problem*, which presents difficult issues to be solved [15, 17]. We should treat this problem for QoE prediction in realistic and useful settings such as MNAR.

Since this paper presented a first-step trial on BN and BS model comparison as a case study, many issues are left unsolved. Future work includes *leave-one-out* full cross-validation of the BS models.

Acknowledgements The author acknowledges Eiichi Isomura and Prof. Toshiro Nunome of Nagoya Institute of Technology for their assistance in obtaining the experimental data.

Funding This work was supported by JSPS KAKENHI (Grant-In-Aid for Scientific Research of Japan Society for the Promotion of Science) Grant No. 17K06454.

Compliance with ethical standards

Conflict of interest The author declares that he has no conflict of interest.

Research involving human participants The subjective experiment in this paper was conducted in 2012 according to the column of "Protection of Human Rights and Compliance with Laws and Regulations" in the author's research proposal for JSPS KAKENHI which had been funded as Grant No. 23656253, and related rules and regulations in Nagoya Institute of Technology, Japan.

Informed consent The 38 participants in the subjective experiment have given consents to the usage of score data and gender information in the author's publications of this experiment in compliance with personnel and accounting rules and regulations in Nagoya Institute of Technology.

Appendix 1: Derivation of $\pi_j(k)$ and $\pi_{jg}(k)$

We can do it in a systematic way by exerting the **querygrain** function on node *S* of a *junction tree* into which the DAG (*dag*) is transformed. Utilizing package **gRain** [35], we can build the junction tree by the **as.grain** function and compute its probability tables by the **compile** function. The condition of interest is specified by the **setEvidence** function with arguments **nodes** and **states** [14]; e.g., nodes = c("*B*","*R*","*P*") and states = "$\boldsymbol{v}_j$" for evaluation of $\pi_j(k)$, and nodes = c("*B*","*R*","*P*","*G*") and states = c("$\boldsymbol{v}_j$","*F*") for evaluation of $\pi_{jF}(k)$. We can get the conditional probabilities $\pi_j(k)$ and $\pi_{jF}(k)$ by applying the querygrain function to the junction tree whose nodes and states have been specified as evidence. Defining a column vector $\boldsymbol{\pi}_j \triangleq (\pi_j(1), \ldots, \pi_j(K))^T$ and a $J \times K$ matrix $\boldsymbol{\Pi} \triangleq (\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_J)^T$, we can compute $\boldsymbol{\Pi}$ as

jCGTbayes $<-$ *setEvidence(compile(as.grain(bn.bayes))*, *nodes*=c("*B*","*R*","*P*"), *states*=c(B_j,R_j,P_j))

$\boldsymbol{\Pi}$ $<-$ *querygrain(jCGTbayes*, *node* = "*S*")$S

Appendix 2: Derivation of $Q_{BN.j}^{(j,1:m)}$

Defining $m_b = (j-1) \times 38 + 1$ and $m_e = (j-1) \times 38 + m$, which are the beginning number of missing data and ending one, respectively (see Table 4), we first make two data frames in R format:

dhav.training.j = *dhav*$[-m_b : -m_e,]$

dhav.test.j = *dhav*$[m_b : m_e,]$

The former is the *training set* for model fitting and is obtained by deleting the m_b-th through m_e-th rows, which contain $\boldsymbol{v}_j$, from *dhav* given by Table 4. The latter is the *test set* for the prediction, which is the part of *dhav* left by the former; it is a data frame consisting of the m_b-th through m_e-th rows of *dhav*. We then fit the model *dag* to *dhav.training.j* and predict the scores for stimulus *j* using *dhav.test.j* by the following two commands:

bn.bayes.j = *bn.fit(dag*, *data* = *dhav.training.j*, *method* = *"bayes"*, *iss* = *10)*,

predicted.bayes.j = *predict(bn.bayes.j*, *node* = *"S"*, *data* = *dhav.test.j*, *method* = *"bayes-lw"*, *n* = *10000*, *prob* = *TRUE)*

where the argument *n* is the number of random samples. Note that the predict function ignores the values of node *S* in *dhav.test.j*, which are to be predicted; we may replace the values with any factors (say, all "100").

Let $\phi_{i.j}^{(j,1:m)}(k)$ denote the predicted probability that $S = k$ for stimulus *j* by subject *i* ($i = 1, \ldots, m$) and define a 5-dimensional column vector

$$\boldsymbol{\phi}_{i.j}^{(j,1:m)} \triangleq (\phi_{i.j}^{(j,1:m)}(1), \ldots, \phi_{i.j}^{(j,1:m)}(5))^T$$

and a $5 \times m$ matrix $\boldsymbol{\Phi}^{(j,1:m)} \triangleq (\boldsymbol{\phi}_{1.j}^{(j,1:m)}, \ldots, \boldsymbol{\phi}_{m.j}^{(j,1:m)})$, which is obtained by

$\boldsymbol{\Phi}^{(j,1:m)}$ $<-$ *attributes(predicted.bayes.j)$prob*.

We can calculate $Q_{BN.j}^{(j,1:m)}$ by using $\boldsymbol{\Phi}^{(j,1:m)}$ in Eq. (4).

References

1. Brooks P, Hestness B (2010) User measures of quality of experience: why being objective and quantitative is important. IEEE Netw 24(2):8–13
2. ur Laghari KR, Crespi N, Connelly K (2012) Toward total of quality of experience: a QoE model in a communication ecosystem. IEEE Commun Mag 50(4):58–65
3. Le Callet P, Möller S, Perkis A (eds) (2013) Qualinet whitepaper on definitions of quality of experience (2012) European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003), Lausanne, Switzerland, Ver. 1.2
4. Mitra K, Zaslavsky A, Åhlund C (2015) Context-aware QoE modelling, measurement and prediction in mobile computing systems. IEEE Trans Mob Comput 14(5):920–936. https://doi.org/10.1109/TMC.2013.155
5. Mok RKP, Chang RKC, Li W (2017) Detecting low-quality workers in QoE crowdtesting: a worker behavior-based approach. IEEE Trans Multimed 19(3):530–543
6. Tasaka S (2017) Bayesian hierarchical regression models for QoE estimation and prediction in audiovisual communications. IEEE Trans Multimed 19(6):1195–1208. https://doi.org/10.1109/TMM.2017.2652064
7. Fan Z, Jiang T, Huang T (2017) Active sampling exploiting reliable informativeness for subjective image quality assessment based on pairwise comparison. IEEE Trans Multimed 19(12):2720–2735
8. Ribeiro FML et al (2018) Quality of experience in a stereoscopic multiview environment. IEEE Trans Multimed 20(1):1–14

9. Jana S, Chan A, Pande A, Mohapatra P (2016) QoE prediction model for mobile video telephony. Multimed Tools Appl 75(13):7957–7980
10. Robitza W, Ahmad A, Kara PA, Atzori L, Martini MG, Raake A, Sun L (2017) Challenges of future multimedia QoE monitoring for internet service providers. Multimed Tools Appl 76:22243–22266
11. Yue T, Wang H, Cheng S (2018) Learning from users: a data-driven method of QoE evaluation for Internet video. Multimed Tools Appl 77(20):27269–27300
12. Pearl J (1988) Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann, San Francisco
13. Korb KB, Nicholson AE (2011) Bayesian artificial intelligence, 2nd edn. CRC Press, Boca Raton
14. Scutari M, Denis J-B (2015) Bayesian networks with examples in R. CRC Press, Boca Raton
15. Congdon P (2006) Bayesian statistical modelling, 2nd edn. Wiley, Chichester
16. Carlin BP, Louis TA (2009) Bayesian methods for data analysis, 3rd edn. CRC Press, Boca Raton
17. Lunn D, Jackson C, Best N, Thomas A, Spiegelhalter D (2013) The BUGS book. CRC Press, Boca Raton
18. Ullah I, Bonnet G, Doyen G, Gaïti D (2010) Modeling user behavior in P2P live video streaming systems through a Bayesian network. In: Proceedings of IFIP2010, AIMS 2010, LNCS 6155. Springer, pp 2–13
19. Karadimce A, Davcev D (2016) Perception of quality in cloud computing based services. In: Proceedings of IEEE QoMEX2016, pp 1–6, Lisbon. https://doi.org/10.1109/QoMEX.2016.7498925
20. Tasaka S (2015) A Bayesian hierarchical model of QoE in interactive audiovisual communications. In: Proceedings of IEEE ICC2015, pp 8611–8617, London. https://doi.org/10.1109/ICC.2015.7249439
21. Tasaka S (2016) Bayesian structural equation modeling of multidimensional QoE in haptic-audiovisual interactive communications. In: Proceedings of IEEE ICC2016, pp 3345–3350, Kuala Lumpur. https://doi.org/10.1109/ICC.2016.7511202
22. Steinbach E, Hirche S, Ernst M, Brandi F, Chaudhari R, Kammerl J, Vittorias I (2012) Haptic communications. Proc IEEE 100(4):937–956. https://doi.org/10.1109/JPROC.2011.2182100
23. Mitra K, Åhlund C, Zaslavsky A (2011) A decision-theoretic approach for quality-of-experience measurement and prediction. In: Proceedings of IEEE ICME 2011, pp 563–566, Barcelona. https://doi.org/10.1109/ICME.2011.6012098
24. Mitra K, Zaslavsky A, Åhlund C (2011) Dynamic Bayesian networks for sequential quality of experience modelling and measurement. In: Smart spaces and next generation wired/wireless networking, LNCS 6869, Springer, pp 135–146. https://doi.org/10.1007/978-3-642-22875-9_12
25. GeNIe—Graphical interface for SMILE. BayesFusion, LLC. https://download.bayesfusion.com/. Accessed 16 Nov 2018
26. Carvalho A, Fraiha S, Carmona J, Gomes H, Araujo J, Cavalcante G (2015) Prediction metrics for QoE/QoS in wireless video networks for indoor environmental planning: a Bayesian approach. In: Proceedings of the 14th international conference on networks (ICN2015), pp 171–176
27. WinBUGS. MRC Biostatistics Unit, University of Cambridge. https://www.mrc-bsu.cam.ac.uk/software/bugs/the-bugs-project-winbugs/. Accessed 16 Nov 2018
28. Isomura E, Tasaka S, Nunome T (2013) QoE assessment of audio-visual and haptic interactive communications over bandwidth guaranteed IP networks. Technical Report of IEICE, CQ2012-76, vol 112, pp 15–20 **(in Japanese)**
29. bnlearn—an R package for Bayesian network learning and inference. http://www.bnlearn.com/. Accessed 16 Nov 2018
30. Nagarajan R, Scutari M, Lèbre S (2013) Bayesian networks in R with applications in systems biology. Springer, Berlin
31. OpenBUGS. http://www.openbugs.net/w/Downloads. Accessed 16 Nov 2018
32. Singhal SK, Cheriton DR (1994) Using a position history-based protocol for distributed object visualization. Technical Report CS–TR–94–1505, Department of Computer Science, Stanford University
33. Hikichi K, Morino H, Fukuda I, Matsumoto S, Yasuda Y, Arimoto I, Iijima M, Sezaki K (2001) Architecture of haptics communication system for adaptation to network environments. In: Proceedings of IEEE ICME2001, pp 563–566
34. Hollander M, Wolfe DA, Chicken E (2013) Nonparametric statistical methods, 3rd edn. Wiley, New York
35. Højsgaard S (2012) Graphical independence networks with the gRain package for R. J Stat Softw 46(10):1–26