DOES PEOPLE'S ABILITY TO MAKE PROBABILITY FORECASTS IMPLY THAT THEY HAVE STATISTICAL MODELS?

A PREPRINT

Adarsh Vijayaraghavan
Department of Information Technology
Rutgers Business School
Newark, NJ 08854
adarsh.vijayaraghavan@rutgers.edu

December 20, 2018

1 Introduction

Humans and other animals have some ability to forecast the future by incorporating information from the past and the present. This ability is essential for our survival in a chaotic, and possibly random, world. The psychology and neuroscience literature on forecasting attempts to answer two questions : is the mind optimal at prediction tasks; and is the mind Bayesian. In many cases, the answers to the two questions overlap. Researchers who believe that the mind is optimal often believe that the mind is Bayesian too (Kording (2014), Knill and Pouget (2004), Nassar et al. (2010), Edwards et al. (2012)).

Kahneman and Tversky (1979) showed that the human mind is sub-optimal in many specific statistical tasks, leading to many studies on systematic cognitive biases. On the other hand, a slew of studies has shown that the mind is in fact surprisingly good at some prediction tasks (Peterson and Beach (1967), Anderson (1991), Bogacz (2007), Griffiths and Tenenbaum (2006)). More that fifty years ago, Peterson and Beach (1967) attempted to reconcile a similar discord in literature with a distinction between what they termed *"descriptive"* and *"inferential"* tasks. In a descriptive task, subjects are directly asked to estimate the parameter of a stochastic process. Consider an example an extension of which is central to this paper. Suppose there is an urn containing green and red rings. Subjects are shown one ring at a time sequentially from this urn, and their task is to estimate the proportion of green balls in the urn. This is a descriptive task. In an inferential task, rather than being asked to directly estimate the proportion of green balls, subjects are tasked with a different question, such as *"there is an urn A with 70 % green rings, and another urn B with 50 % green rings. How likely is it that the sequence of rings were drawn from urn A?"*. Answering such a question would invole the calculation of *inverse probabilities*, i.e., subjects would have to weigh two different hypothesis based on the sequence of rings as evidence. Peterson and Beach (1967) observed that subjects are generally good at descriptive tasks, but not as good at inference tasks. Similar views have been advanced by some recent authors too (Brown and Steyvers (2009), Gallistel et al. (2014)).

Assuming that the mind is optimal, at least in certain circumstances, it is still not clear if the mind performs a Bayesian calculation to achieve this optimality. Bowers and Davis (2012) used a Bayesian argument against such a proposition, which we discuss further in Section 6. They also use the terminologies introduced by Marr (1982). Marr distinguished between *computational models* and *algorithmic models*. Computational models explore how a certain task can be completed without considering the possibility that the mind implements a similar model. An algorithmic model, on the other hand, considers a model from the perspective of how the mind can implements it neurobiologically. Such a model has to factor in how the mind stores the data required (the *data structures* used) and the mechanism by which the mind performs these calculations. There is a potential source of confusion with the terminologies *"algorithmic"* and *"model"* here. I have already described the term *"algorithmic"* above. The term *"model"* here is used to indicate that we are trying to model the subject's behavior. From a statistician's point of view, a model could mean something different, and we make this distinction clearer a little later in this section and in section 2. Section 2 also explains my use of the term *"algorithms"*. To avoid this confusion, I will explicitly contextualize these two terms whenever I use them. Bowers and Davis (2012) extend Marr's terminology to classify Bayesian researchers in psychology and

neuroscience as *algorithmic*, *methodological*, and *theoretical*. Their use of the term *"algorithmic"* has a completely different meaning adding to the confusion with the terminologies. It suffices to remark that Bowers and Davis (2012) claim that the algorithmic Bayesians are very rare, and I will not be referring to this usage anymore in this paper. The methodological Bayesians propose computational models of how a task is performed by the mind. However, they do not explicitly claim that that the mind faithfully implements an identical model. Methodological Bayesians suggest that the mind may be using certain heuristics and approximations to perform calculations that are similar to their computational models. Theoretical Bayesians, on the other hand, claim that their models are algorithmic. Bowers and Davis (2012) contend that such claims must also demonstrate how the mind stores priors, calculates likelihoods, multiplies both, and when required, normalizes the product with the normalizing factor. However, there is little evidence that the mind can perform all these steps. Besides Bowers and Davis (2012), other researchers have questioned the claims that the mind is optimal (Colombo et al. (2018), Tauber et al. (2017), Elqayam and Evans (2011)).

In this paper, we would like to ask a third question about the mind's forecasting ability - does people's ability to make forecasts imply that they have *statistical models*? The term *model* as used in the statistics literature represents a stochastic data-generating process. A statistical model encapsulates the assumptions we make about data. In Breiman et al. (2001)'s terminology, these models are "*data models*". For instance, an assumption usually made in various models is that each observation is independent of the other observations. Another assumption could be that the errors in measurement of each observation follow a Gaussian distribution. However, there are also methods from the forecasting literature that do not make use of statistical models. These methods predict the next outcome in a sequence by minimizing some learning function. In Breiman et al. (2001)'s terminology, these are "*algorithmic models*". Note that this is different from the algorithmic model of the mind described in the previous paragraph.

We are going to focus on the data generated from one experiment in the psychology literature. The study we focus on is by Gallistel et al. (2014), which is very similar to the descriptive experiment we saw earlier, except for the fact that the proportion of green rings in the urn can potentially vary from trial-to-trial. This is a non-stationary Bernoulli process - a Bernoulli process because if the probability of seeing a green ring in any given trial is p, the probability of seeing a red ring is $(1\text{-}p)$; and non-stationary because p is not fixed. In this study titled *"The perception of probability"*, Gallistel et al. (2014) propose a semi-Bayesian changepoint model to estimate the trial-by-trial probability of our non-stationary Bernoulli process. We refer to their model as the GKLML model based on the initials of the authors. In this experiment, subjects are sequentially shown 1000 rings with replacement from an urn filled with red and green balls, and tasked with estimating the proportion of green rings in the urn before each trial. The subjects are informed that this proportion of green rings is not constant. The subjects can indicate their current estimate of the proportion of the green balls in the urn by moving a slider between 0 to 1 (inclusive on both sides). Note that this proportion of green rings in the urn can be interpretted as the hidden true probability of observing a green ring in the next trial. Gallistel et al. (2014) show that the subjects' behavior closely resembles the behavior exhibited by the GKLML model. One of the prominent characteristic using which they demonstrate this resemblance is the distribution of *"step heights"* and *"step widths"*. Step height is the magnitude by which the estimated probability changes from one trial to another. Step width is the number of trials for which the estimated probability is kept constant before determining that there needs to be a change in the estimate.

My contribution in this paper is as follows. I have replicated the GKLML model programmatically to ensure that I have a thorough understanding of it. Further, I attempt to analyze other non-Bayesian methods that lead to predictions with the same characteristics as the subjects' predictions. Apart from possessing the advantages of being conceptually and computationally simple, these methods do not make use of any statistical models. I then argue that the fact that the mind makes optimal predictions in some very specific tasks do not necessarily imply that the mind makes use of Bayesian methods, or that the mind has statistical models to draw upon.

The rest of this article is structured as follows. In section (2), I will briefly contrast making predictions through statistical models with algorithmic models. The loss function for these algorithmic models is a proper scoring rule, and I provide a brief description of proper scoring rules in the same section. In section (3), I describe the experiment performed in Gallistel et al. (2014). I then describe their examination of simple trial-by-trial non-Bayesian methods which do not fit the step heights and step widths generated by the actual subjects in the experiment. Finally, I provide a brief description of the GKLML model.

In section 4, I suggest some methods that explain this data without making use of statistical models. In section 5, I compare the GKLML model with our methods in terms of the step heights and step widths. I also compare these methods in terms of the proper scoring rules I introduce in section (2). The proper scoring rules help measure the precision of our estimates. I will conclude with a discussion in section (6).

2 Probability forecasting and statistical modeling

In this section, I am going to contrast two different ways of making predictions. The two different ways I discuss are somewhat in the lines of what Breiman et al. (2001) called the two different cultures - the *Data Modeling* culture and the *Algorithmic Modeling* culture. To avoid confusion with some of the terminologies from the psychology and neuroscience literature, I refer to these two cultures as *predictions using statistical models* and *predictions without statistical models*. I use the term *algorithm* in its broader sense representing a step-by-step computational procedure.

A statistical model encapsulates our assumptions about the data generating process. In the case of our example with rings drawn from an urn, one assumption that could be made is that the rings are drawn from a binomial distribution with a parameter p. A Bayesian statistical model may include assumptions about the *a priori* probability distribution that is in a mathematically convenient form. On the other hand, methods of prediction without statistical models do not make any assumption about the underlying data generating process. As Cesa-Bianchi and Lugosi (2006) note in their book, such an approach can be applied for causal mechanisms that are deterministic, stochastic, or even adversarial. A few such methods are described below.

2.1 Exponential smoothing method

Exponential smoothing methods are very common in the forecasting literature, and an early explanation of the method was provided by Holt (1957). This method is conceptually very simple, easy to implement, and tail-recursive. The exponential smoothing method takes a weighted average of all previous observations to predict the outcome of the next trial. The weights, however, are unequal. The more recent observations have more weight, while the less recent ones have exponentially lower weights. This paper deals only with Bernoulli trials with 2 outcomes, so we can consider a simplified formulation :

$$\hat{y_t} = \lambda * x_{\text{t-1}} + (1 - \lambda)\hat{y_{\text{t-1}}}$$

Where $\hat{y_t}$ is the Bernoulli parameter of the next trial which we are attempting to predict. $x_{\text{t-1}} \in \{0, 1\}$ and depends on the latest observation. λ is the memory parameter that controls how rapidly the weight of older observations decreases. In a situation of high volatility where the underlying true probability we are trying to estimate changes rapidly, λ needs to be low. If the volatility is low, λ needs to be higher. The formula is recursive, and makes use of all past observations. We usually start by setting $\hat{y_1}$ with an arbitrary value, usually 1/2, and then use the formula recursively for subsequent predictions after observing each trial.

2.2 Prediction with expert advice

Prediction with expert advice is a another common method in the forecasting literature. Prediction with expert advice can incorporate statistical models too, but this is not relevant to the approaches considered in this paper. The idea behind prediction with expert advice is as follows. If there are *K* experts making predictions after each stage, we can combine the information provided by each expert along with any other information we might possess to do almost as well as the best expert when seen in retrospect. The goodness of an expert is measured using a pre-selected *proper scoring rule*. Before taking a look at such approaches in detail, I would briefly describe proper scoring rules. Gneiting and Raftery (2007) provide a more detailed account.

Probabilistic forecasts are given in the form of probability distributions. *Scoring rules* are functions that evaluate to a numerical measure based on the predictive probability distribution and the actual outcome. A scoring rule can be used to evaluate the forecasting ability of a forecaster and comparatively rate different forecasters. The scoring rule is in this sense, a reward function. For example, if γ is the forecaster's predictive probability distribution, and ω is the actual event that materializes,

$$\lambda(\omega, \gamma)$$

is a scoring rule. A *proper scoring rule* is one that incentivizes the forecaster to give their true beliefs as the predictive probability distribution. When a proper scoring rule is used, the reward for a forecaster is maximized when they give the truth as their predictive probability distribution. A *strictly proper scoring rule* ensures that the forecaster's reward is uniquely maximized when they give the truth as the predictive probability distribution. Gneiting and Katzfuss (2014) describe the necessity for *calibrated* and *sharp* forecasts. A well-calibrated forecast is such that the predictive probability distribution and the observations are statistically indistinguishable. They behave as if they are from the same stochastic distribution. A sharp forecast is well-concentrated, rather than dispersed, in the predictive space.

Brier (1950) provided one of the earliest known formulations of a scoring rule, a generalized version of which is known as the *Brier score*. We will be discussing the generalized version of Brier's formulation below. While Brier score is a *quadratic score*, other proper scoring rules include *logarithmic score* and *spherical score*. These are described in Gneiting and Raftery (2007). For a generalized version of Brier score, we refer to Vovk and Zhdanov (2009). Continuing

with the notations ω, λ and γ introduced earlier, let the observation space be Ω and the decision space be Γ. We restrict Ω to the finite set. The Brier score is defined as

$$\lambda(\omega, \gamma) = \sum\nolimits_{o \in \Omega} (\gamma\{o\} - \delta_\omega\{o\})^2$$

Here, $\delta_\omega \in P(\Omega)$ is the probability measure concentrated on ω. Vovk and Zhdanov (2009) provide an example where, if the observation space Ω is {1, 2, 3}, and the predictive probability distribution γ assigns probabilities of 1/2, 1/4 and 1/4 to events 1, 2 and 3 respectively, we can calculate

$$\lambda(\omega, \gamma) = (1/2 - 1)^2 + (1/4 - 0)^2 + (1/4 - 0)^2 = 3/8$$

.

Vovk and Zhdanov (2009) define a protocol ((1)) for predictions obtained by the combination of expert opinions. As is common in forecasting literature, the idea of prediction is seen as a game between reality and the forecaster.

Protocol 1 Prediction with expert advice

$L_0 \leftarrow 0$
$L^{\mathrm{k}}{}_0 \leftarrow 0$, for k = 1, 2, 3,.. K
for N = 1, 2, ... **do**
  Expert k announces $\gamma_{\mathrm{N}}{}^{\mathrm{k}} \in \Gamma$, k = 1, 2,.. K.
  Forecaster announces $\gamma_{\mathrm{N}} \in \Gamma$.
  Reality announces $\omega_N \in \Omega$.
  $L_{\mathrm{N}} \leftarrow L_{\mathrm{N-1}} + \lambda(\omega_N, \gamma_N)$.
  $L^{\mathrm{k}}{}_{\mathrm{N}} \leftarrow L^{\mathrm{k}}{}_{\mathrm{N-1}} + \lambda(\omega_N, \gamma_{\mathrm{N}}{}^{\mathrm{k}})$. k = 1, 2, .. K.
end for

The protocol is fairly simple. We initiate the total loss function of the forecaster and the individual loss functions of each expert with 0. We then observe each trial one-by-one. For every trial, each expert announces their predictive probability distribution. The forecaster combines the predictions made by all experts in some unspecified way and makes their own prediction. Finally, reality announces the truth. The total loss function of the forecaster and each individual expert is then augmented with the score from the current trial.

For such a protocol, Vovk and Zhdanov (2009) provide the *strong aggregating algorithm* ((2)) as an optimal strategy.

Algorithm 2 Strong aggregating algorithm

$w^{\mathrm{k}}{}_0 \leftarrow 1$, for k = 1, 2, 3,.. K
for N = 1, 2, .. **do**
  Read the expert predictions $\gamma_{\mathrm{N}}{}^{\mathrm{k}}$, k = 1, 2,.. K.
  Set $G_N(\omega) \leftarrow -ln \sum_{k=1}^{K} w_{\mathrm{N-1}}{}^{\mathrm{k}} e^{-\lambda(\omega, \gamma^{\mathrm{k}}{}_{\mathrm{N}})}, \omega \in \Omega$.
  Solve $\sum_{\omega \in \Omega} (s - G_N(\omega))^+ = 2 \; such \; that \; s \in \mathbb{R}$.
  Set $\gamma_N\{\omega\} \leftarrow (s - G_N(\omega))^+/2, \; \omega \in \Omega$.
  Output prediction $\gamma_N \in P(\Omega)$.
  Read observation ω_N.
  $w^{\mathrm{k}}{}_{\mathrm{N}} \leftarrow w^{\mathrm{k}}{}_{\mathrm{N-1}} + \lambda(\omega_N, \gamma_{\mathrm{N}}{}^{\mathrm{k}})$. k = 1, 2,.. K.
end for

The algorithm starts by assigning a weight of 1 to each of the K experts. For every observation, the prediction given by each expert is averaged based on their weights and then normalized to a value between 0 and 1 (to arrive at a probability). After seeing the truth, the weights of each expert are reconsidered based on their score for that particular trial and their previous weight. The scoring rule used here is the Brier score. Vovk and Zhdanov (2009) prove that this is asymptotically the optimal and the best possible strategy. Their theorem is stated below :

> Using Algorithm 1 (the strong aggregating algorithm) as Learner's strategy in Protocol 1 for the Brier game guarantees that
>
> $$L_N \leq \min_{k=1,2,\ldots K} L_{\mathrm{N}}{}^{\mathrm{k}} + \ln K$$
>
> for all N = 1, 2, . . . If A < $\ln$ K, Learner does not have a strategy guaranteeing
>
> $$L_N \leq \min_{k=1,2,\ldots K} L_{\mathrm{N}}{}^{\mathrm{k}} + A$$

Here, the term *Learner* refers to the forecaster.

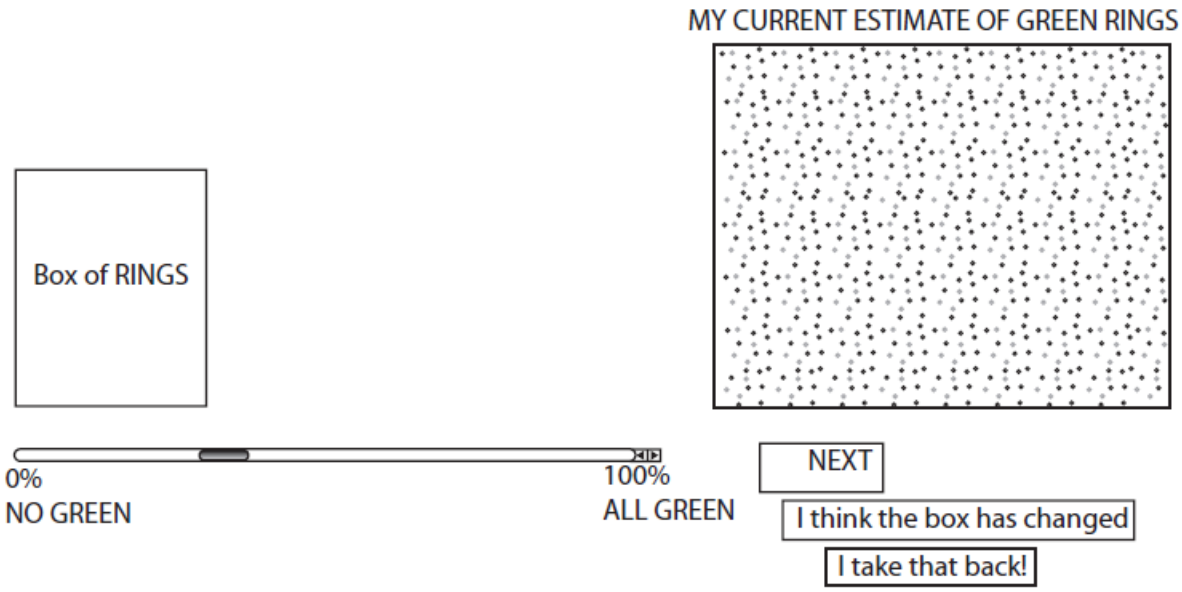

Figure 1: The experimental setup used by Gallistel et. al. (Fig. 3 from "The perception of probability")

3 The Gallistel arguement for human statistical models

3.1 Description of the experiment

The experimental set-up used by Gallistel et al. (2014) is shown in figure 1 . 10 subjects were involved in this experiment, and each subject participated in 10 sessions. In every session, the subjects were shown 1000 rings from an imaginary urn. Each ring was either red or green. The subjects were shown one ring at a time, and were instructed to use the slider at the bottom-left to indicate their estimate of the proportion of the green balls in the urn. The box on the top-right is a visualization of the slider setting selected by the user. On the bottom-right, there is a button for the subject to request the next ring. The subject can also use the *"I think the box has changed"* button to indicate that they feel there is an underlying change in the true proportion of green and red balls in the urn. The *"I take that back"* button can be used to indicate that the subject changed their mind about there being a change in the true probability. 1 note that this button was unavailable for the first 5 subjects. For the remaining subjects, this button indicates a *"second thought"* about their previous slider movement.

The probability that the true proportion of green rings in the urn would change in any given trial was fixed at 0.005, and a geometric distribution was used to decide if the true proportion would change in a given trial. Hence the expectation of step width was 200. The magnitude of such a change was approximately uniformly distributed between the two intervals -0.85 to -0.2 and 0.2 to 0.85.

3.2 Human subjects' distribution of step heights and widths not matched by simple trial-by-trial updating models

Gallistel et al. (2014) attempt to model the subjects' behavior with a few variations of trial-by-trial updating methods proposed by Behrens et al. (Behrens et al. (8 05)). This method is identical to the *exponential smoothing* method we discussed in section 2.1. The exponential smoothing method is essentially a running average method, and it re-estimates the predicted value after every trial. The re-estimated prediction is very unlikely to be the same as the previous prediction. An interpretation of making such predictions which change in every trial is that there is a underlying change in the hidden proportion of green rings in the urn at every trial. This, as we know, is not true. The subjects do not re-estimate the proportion of green balls at each turn, and the subjects' estimate across all trials follows a step function (fig. 2). Subjects in earlier experiment conducted by Robinson (1964) also exhibited a similar step pattern.

To model this step-behavior, Gallistel et al. (2014) start by introducing a threshold T. In their first model, it is assumed that subjects do not move the slider unless the difference between their previously estimated probability and the actual

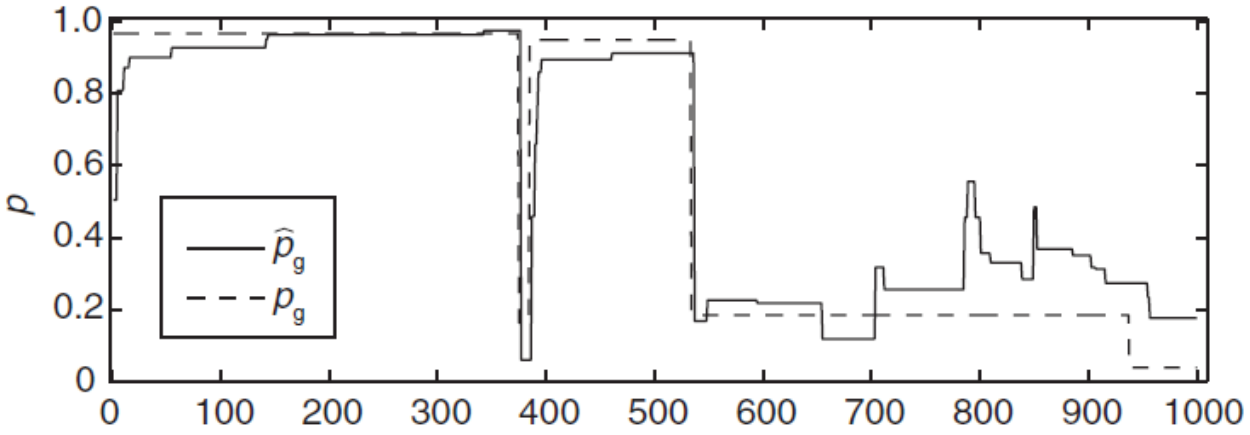

Figure 2: The trial-by-trial slider setting of the subject (solid line) vs the hidden true probability (dotted line). The subject display a step-holding pattern. Fig. 5 from "The perception of probability".

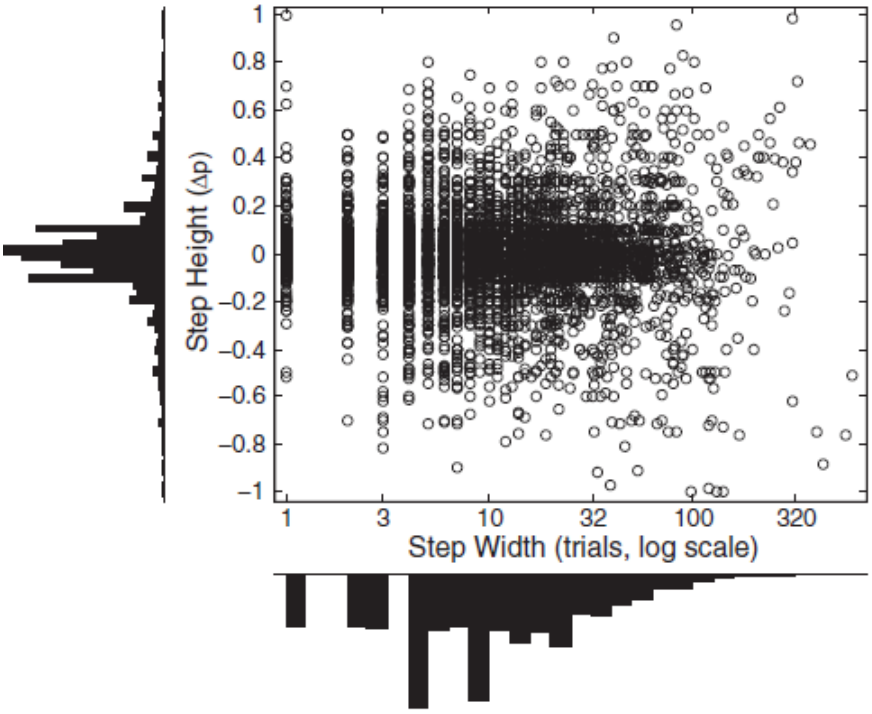

Figure 3: The joint distribution of step heights and step widths of subjects' slider moments for a single session. The step-widths are in the x-axis and the step heights are in y-axis with linear scale. The marginal distribution of step heights and step widths are shown in the overlaid diagrams along the y-axis and x-axis respectively. (See Fig. 11 from "The perception of probability" for a similar figure across all subjects and sessions)

observed probability is larger than the parameter *T*. However, this model too does not match the actual subjects' slider movements. Fig. 3 shows the joint distribution of step heights and step widths of subjects' slider movements. The marginal distributions are shown along each of the axis. As can be seen from the marginal distribution of step heights, subjects are more prone to making slider movements that are smaller in magnitude. Introducing a threshold will suppress these smaller slider movements and result in a distribution that is bi-modal.

Gallistel et al. (2014) then attempt two other models. The first is similar to the model we just described. However, instead of using a fixed threshold, they use a Gaussian distribution to randomly generate a threshold value. The second, referred to as the *"two kernel model"*, is a little more complicated and combines ideas from both the previous models. We will start by briefly explaining the term volatility to understand this model. Volatility is roughly how frequently the hidden true proportion of green rings in the urn changes. In a single session of experiment consisting of 1000 trials, there may be small sub-sequences where the volatility is high, and other sub-sequences where volatility is relatively low. This is because, as described in the previous section, the probability that the true proportion of rings in the urn would change is not uniformly distributed (it is a geometric distribution). This model generates two different probability estimates using two different memory parameters - one with a longer memory indicating low volatility, and another with a shorter memory indicating high volatility. The absolute difference between these two estimates are compared to yet another threshold to decide the current running average probability. Finally, this running average probability is again compared with the threshold T to decide if the slider should be moved. Despite the seeming complexity, it is obvious that this model will not work as it still suppress the smaller slider movements.

Gallistel et al. (2014) conclude that both these models do not fit the behavior of subjects. In section 4, we will propose a model that extends on the exponential smoothing method and still fits the distribution of subjects' slider movements reasonably.

3.3 The GKLML model

Having attempted simpler models that do not incorporate Bayesian priors, Gallistel et al. (2014) then propose a semi-Bayesian model to explain the behavior of the subjects. The flowchart in fig. 4 gives an overview of this model. We will refer to this two-step changepoint algorithm as the GKLML model. Our detailed implementation of this algorithm is described in Appendix B. Appendix A gives some of the notations we use as a part of this algorithm. We will briefly describe the algorithm here for the sake of continuity.

The core idea behind this algorithm is to reduce the amount of information that needs to be stored at any point of time. Instead of storing the complete sequence of observations in a given trial, we store just two sub-sequences of variable lengths. These sub-sequences are determined by our estimate of points at which we believe there was a change in the underlying true proportion of green rings in the urn. We call these points *changepoints*. Changepoints are essentially trial numbers identifying the trials at which we presume the changes occurred. After every trial, the current estimate of the probability of seeing a green ball is compared with the relative number of green balls seen since the latest changepoint. Until the product of this deviation and the number of observations since the latest changepoint is significant, no action is taken. If there is a significant deviation, a Bayesian model comparison is used to determine if there was an additional change in the true probability of seeing a green ring. If no changepoint is detected, the latest changepoint is temperorily removed, and a second round of Bayesian model comparison is performed. Based on this comparison, the latest changepoint is either dropped or relocated. Throughout the calculation, prior probability distributions for the true probability of seeing a green ring, as well as the volatility in this true probability is maintained and reevaluated.

4 Alternative simple models that minimize a proper scoring rule

4.1 Degraded Brier smoothing algorithm

We use Algorithm (2) as a part of what I call the *degraded Brier smoothing algorithm* (algorithm (3)). The degraded Brier smoothing algorithm uses the simple exponential smoothing method described in section 2.1. Different experts have a different memory parameter, and their forecasts are combined using the strong aggregating algorithm described in section 2.2. The final prediction is continuous, and can potentially be different for each trial. To model the behavior of the human subjects, we then degrade this algorithm by introducing an element of randomness to decide if we will move the slider.

Since we are trying to model a human subject's behavior, we visualize an imaginary slider that keeps track of our current estimate of the number of green balls in the urn. This allows us to continue having the notion of step heights and step widths.

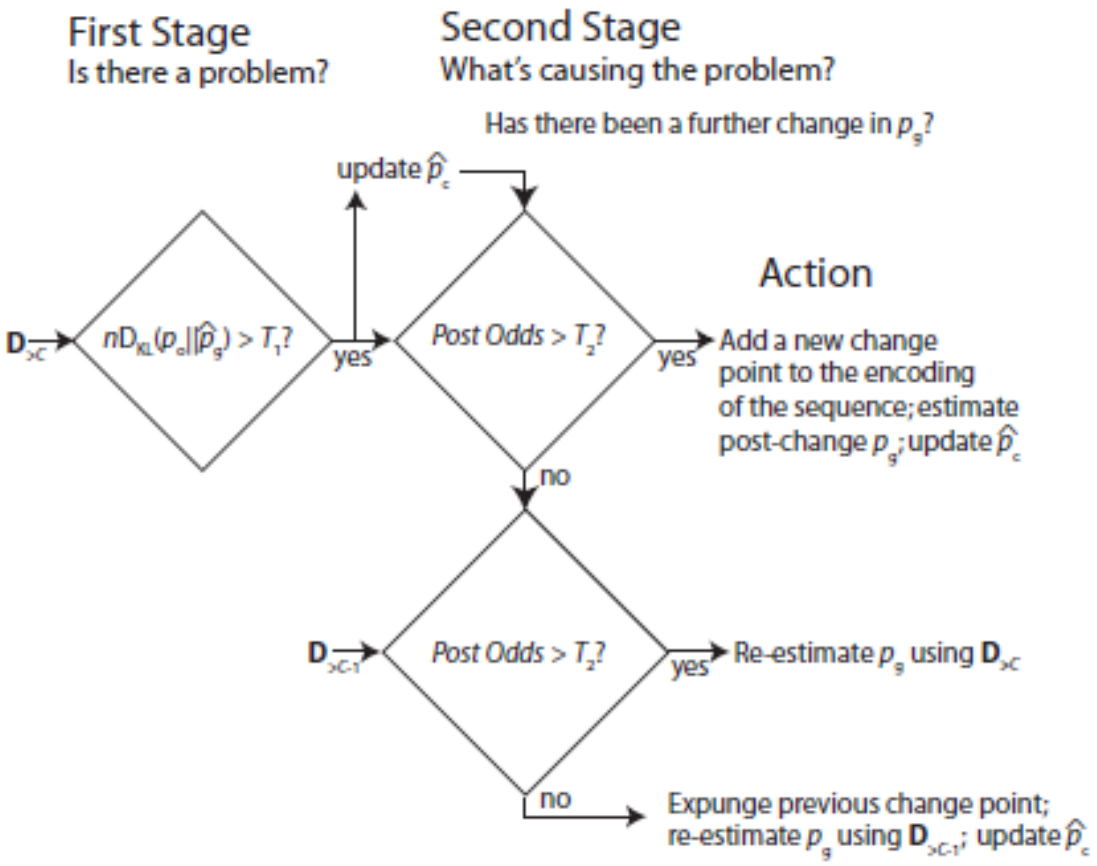

Figure 4: The flowchart for the two-step changepoint algorithm (Fig. 13 from "The perception of probability")

Algorithm 3 Degraded Brier smoothing algorithm

Set width $\leftarrow 0$
for N = 1, 2, .. **do**
  Read the expert predictions $\hat{P_N}^k$, k = 1, 2,.. K.
  Combine the predictions of K experts using the strong aggregating algorithm into $\hat{P}_N$.
  Generate a random number from an uniform distribution of 0 to 1.
  if the random number falls within region Q **then**
    Move the slider to $\hat{P}_N$.
    Set width $\leftarrow 0$.
    Skip the current iteration and continue with the next iteration.
  end if
  Calculate height = $\hat{P}_N - \hat{P_{N-1}}$.
  Set Deviation $\leftarrow \sqrt{(width/A)^2 + (height/B)^2}$.
  if Deviation > C **then**
    Move the slider to $\hat{P}_N$.
    Set width $\leftarrow 0$.
    Skip the current iteration and continue with the next iteration.
  end if
  Keep the slider at $\hat{P_{N-1}}$.
  Set width $\leftarrow$ width + 1.
end for

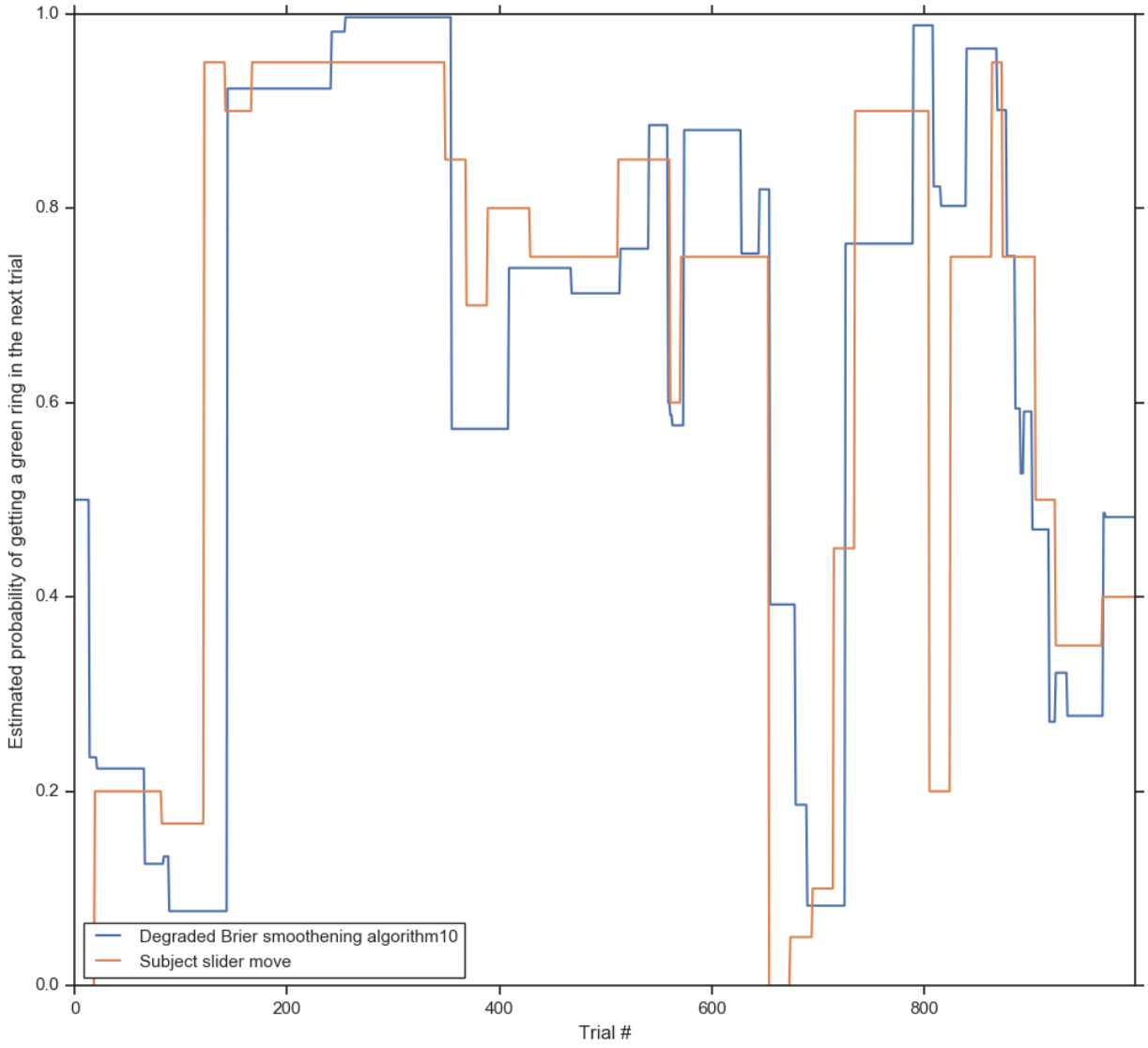

Figure 5: Slider setting as a function of the trial number. The orange line indicates the subject's slider movements, whereas the blue line indicates the slider movements by the degraded Brier smoothening algorithm. The degraded Brier smoothening algorithm closely resembles the subject's own behaviour.

In algorithm 3, Q is a sub-interval within 0 to 1 where the slider is randomly moved regardless of the slider height or width. One possible value is the interval (0.70, 0.74) excluding both the bounds. Here, the region itself is not significant as much as the size of the region, since we are using an uniform distribution to generate the random numbers. A, B and C are free-parameters that depends on the subject we are trying to model. One possible combination is A = 250, B = 100 and C = 0.7.

5 Results

We used 3 experts for the degraded Brier smoothing algorithm with memory parameters of 1/7, 1/10 and 1/13 respectively. The trial-by-trial slider setting of the degraded Brier smoothing algorithm for a single subject in a single trial is shown in fig. 5 (blue line). The subject's own slider movements (orange line) for the same trial is also superimposed in the figure. The degraded Brier smoothing model is able to model the subject's behavior reasonably well. The four free-parameters of the degraded Brier smoothing algorithm control the number of slider movements made. The joint-distribution of step

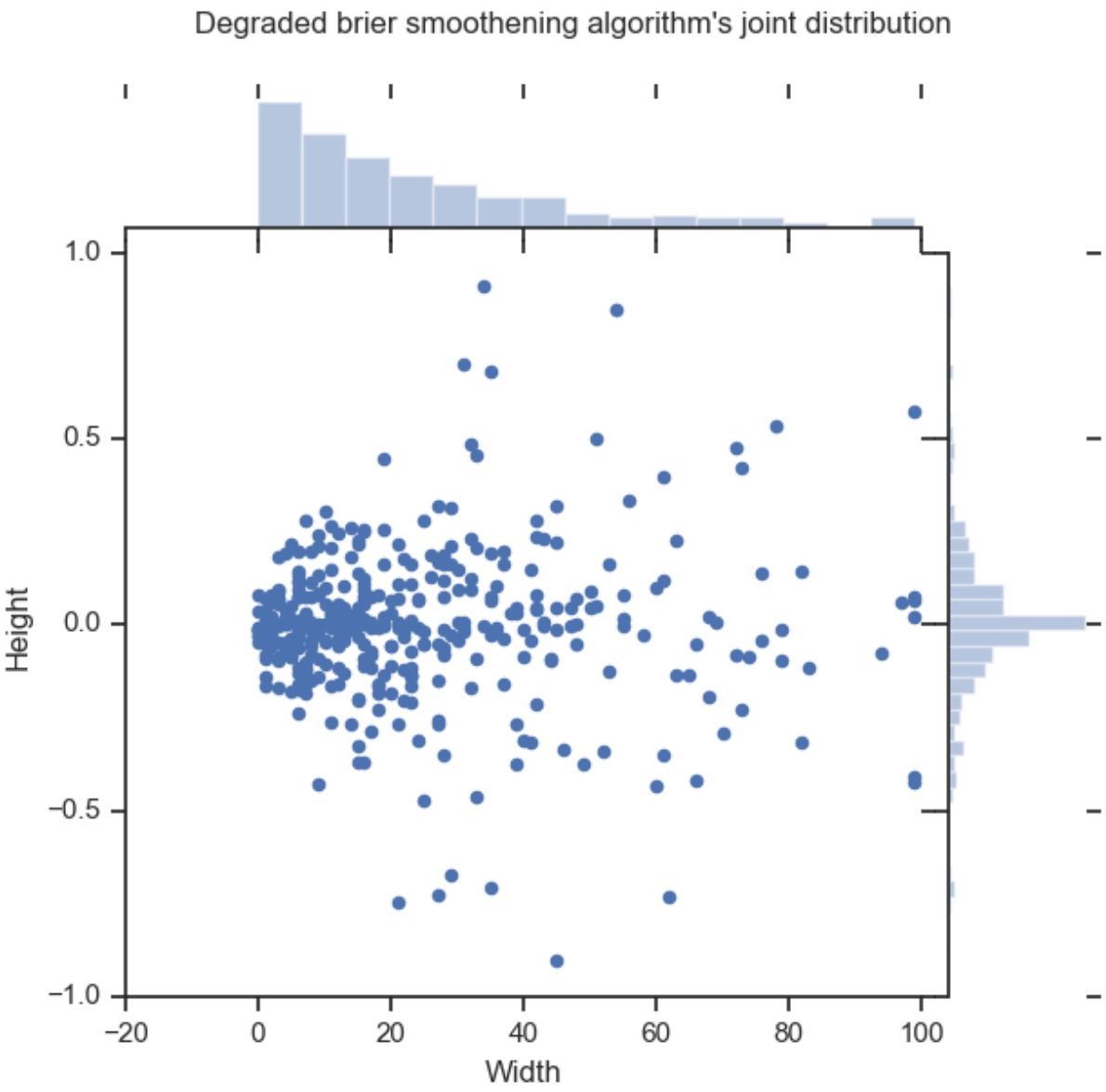

Figure 6: The joint distribution of the step heights and step widths of the slider movements generated by the degraded Brier smoothing algorithm for a single subject in one trial. The uni-modal nature of the step heights shown is very similar to the subject's behavior from 3 above. The step widths are a reasonable imitation of the subject's behavior, but there is an asymptotic decrease of the number of slider movements with the step height, unlike the subject's behavior. However, this behavior is similar to what was achieved by the GKLML model.

heights and step widths of the slider movements generated by the degraded Brier smoothing algorithm is shown in fig. 6. Comparing this with fig. 3, we see a fair resemblance. The behavior of step heights is very similar to that of the subjects' own behavior. The behavior of step widths is reasonably similar. However, the subjects do not display the asymptotic decrease in number of slider movements with increasing step widths that is seen in fig. 6. The behavior of the degraded Brier score smoothing algorithm in this regards is similar to that displayed by the GKLML model shown in fig. 15 of Gallistel et al. (2014).

Table 1 compares the Brier scores for each of the methodology for a single trial of a single subject. The truth has a Brier score of 0.1392, which is the best we can hope to achieve. The Brier smoothing algorithm without the degradation does well, but this method has a varying probability estimate in each trial and does not model the subject's behavior. The GKLML method does very well considering that it is able to model the subject's behavior too. The degraded Brier smoothing algorithm matches the subject's behavior, but it's Brier score needs to be improved upon. Surprisingly, the subject does better than the degraded Brier smoothing algorithm.

Methodology	**Brier Score**
Truth	0.1392
Brier smoothing algorithm without degradation	0.1517
GKLML model	0.1560
Subject	0.1587
Degraded Brier smoothing algorithm	0.1794

Table 1: A comparison of Brier scores for various methodologies.

6 Discussion

Robinson (1964) and Gallistel et al. (2014) have shown us that the human mind is surprisingly optimal in specific task of estimating the parameter of a non-stationary Bernoulli process. But does Gallistel et al. (2014)'s computational Bayesian model imply an identical algorithmic model?

Bowers and Davis (2012) used Bayes theorem to make a case against studies that conclude that humans are optimal Bayesian estimators. They formulated Bayes theorem as follows :

$$P(H_{\text{Optimal}}|E) = \frac{P(H_{\text{Optimal}}) * P(E|H_{\text{Optimal}})}{P(E)}$$

Here, H_{Optimal} is the hypothesis that that human mind is an optimal Bayesian estimator. E is the evidence we have in the form of data. The left hand term is the posterior distribution that the mind is an optimal Bayesian estimator given the evidence. The numerator on the right hand term is a product of the prior probability that the human mind is an optimal Bayesian estimator and the likelihood of seeing the data we do given that the human mind is an optimal Bayesian estimator. The denominator is the total evidence that the human mind is an optimal Bayesian estimator. Bowers and Davis (2012) argue that the two terms in the numerator are overestimated and that the denominator is underestimated. In their view, the prior probability that the human mind is optimal is low because of the conflicting results from studies on the optimality of the mind. The likelihood is overestimated because, Bowers and Davis (2012) claim, Bayesian methodologies are flexible in fitting the data. The total evidence is underestimated because researchers often do not consider enough non-Bayesian methods that can explain human behavior.

Responding to the general criticism from Bowers and Davis (2012), Griffiths et al. (2012) clarified that

> Most Bayesian models of cognition are defined at Marr's (1982) "computational level," characterizing the problem people are solving and its ideal solution. Such models make no direct claim about cognitive processes – what Marr termed the "algorithmic level."

However, we notice that there is no clear delineation between algorithmic and computational claims in literature, and often, researchers who set out to make computational claims draw algorithmic conclusion. Bowers and Davis (2012) remarked on this fact too, commenting that :

> If neuroscience is to provide any evidence for the theoretical Bayesian perspective, the key question is, what nonbehavioral evidence exists that the neurons compute in this way? The answer is none, unless Bayesian computations are characterized so loosely that they are consistent with almost any computational theory.

In principle, we agree with the prominent proponents of Bayesian mind Griffiths et al. (2012) when they state

> .. a computational level analysis plays a role in explaining cognition similar to that played by a mathematical theory of aerodynamics in explaining bird flight. The theory of aerodynamics says nothing about the anatomical mechanisms of bone and muscle that support flight other than that they must provide a solution with particular properties.

In other words, any computational model we can come up with says nothing about the actual algorithmic model implemented by our brain. Referring back to the Bayes formulation given by Bowers and Davis (2012), our study targets the denominator term of total evidence, where we would be attempting to show that there are multiple non-Bayesian methods that explain mind's optimal behaviour. The degraded Brier smoothening algorithm is just a starting point that makes use of a computationally simple algorithm which reasonably resembles subject's behavior. We would like to broaden the debate by questioning the claim that the mind makes use of statistical models in the first place. Such a study will also enable us to compare various methods from the forecasting literature on this dataset, and thus gather additional insights for probabilistic forecasting.

Finally, Griffiths et al. (2012) make an argument about the explanatory power of Bayesian models, arguing that :

> The teleological explanations yielded by Bayesian models of cognition are valuable not just because they satisfy our desire to answer why questions, but because they provide the foundation for universal laws of cognition – principles that we expect to hold true for intelligent organisms of any kind, anywhere in the universe.

Such a quest for universal laws has gained more recent traction with ideas such as *free energy principle* by Friston et al. (2006). On the other hand, Ramachandran (1990) suggested that

> Nature is inherently opportunistic and will often adopt some curious – even bizarre – solutions to its problems, especially when it has to make use of preexisting hardware.

Despite the attraction of Universal models that explain forecasting abilities, it may be that the mind, successfully at the times and unsuccessfully at others, is just trying to beat reality by making forecasts without using statistical models.

7 Further areas of research

Our implementation of the degraded Brier smoothing algorithm is only a starting point in this study. The next step would be to improve the Brier score of this algorithm. A surprising fact is that the subjects do better than the degraded Brier smoothing algorithm. The data from experiments conducted by Gallistel et al. (2014) would prove useful for comparing different forecasting methods in terms of minimizing a proper scoring rule. A few methods that can be considered are from Vovk et al. (2005), Freund (2003), Weissman and Merhav (2001), Vovk and Shafer (2005), Rakhlin and Sridharan (2015) and use of static experts (Cesa-Bianchi et al. (1997)). There could be potentially many computational models that explain human behavior in a non-stationary Bernoulli process.

8 Acknowledgements

I would like to thank Prof. Glenn Shafer for his invaluable guidance and advice for this study. I would also like to thank Prof. C.R. Gallistel for providing the data and programs from his experiment, and clarifying some of the queries we had on the data-set. Finally, I would like to thank Prof. Robin Gong and Prof. Harry Crane for their helpful suggestions.

References

Anderson, J. R. (1991). Is human cognition adaptive? *Behavioral and Brain Sciences*, 14(3):471–485.

Behrens, T. E. J., Woolrich, M. W., Walton, M. E., and Rushworth, M. F. S. (2007-08-05). Learning the value of information in an uncertain world. *Nature Neuroscience*, 10(9):1214–1221.

Bogacz, R. (2007). Optimal decision-making theories: linking neurobiology with behaviour. *Trends in cognitive sciences*, 11(3):118–125.

Bowers, J. S. and Davis, C. J. (2012). Bayesian just-so stories in psychology and neuroscience. *Psychological bulletin*, 138 3:389–414.

Breiman, L. et al. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, 16(3):199–231.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthey Weather Review*, 78(1):1–3.

Brown, S. D. and Steyvers, M. (2009). Detecting and predicting changes. *Cognitive psychology*, 58(1):49–67.

Cesa-Bianchi, N., Freund, Y., Haussler, D., Helmbold, D. P., Schapire, R. E., and Warmuth, M. K. (1997). How to use expert advice. *Journal of the ACM (JACM)*, 44(3):427–485.

Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, learning, and games*. Cambridge university press.

Colombo, M., Elkin, L., and Hartmann, S. (2018). Being realist about bayes, and the predictive processing theory of mind.

Edwards, M. J., Adams, R. A., Brown, H., Parees, I., and Friston, K. J. (2012). A bayesian account of 'hysteria'. *Brain*, 135(11):3495–3512.

Elqayam, S. and Evans, J. S. B. (2011). Subtracting "ought" from "is": Descriptivism versus normativism in the study of human thinking. *Behavioral and Brain Sciences*, 34(5):233–248.

Freund, Y. (2003). Predicting a binary sequence almost as well as the optimal biased coin. *Information and Computation*, 182(2):73 – 94.

Friston, K., Kilner, J., and Harrison, L. (2006). A free energy principle for the brain. *Journal of Physiology-Paris*, 100(1-3):70–87.

Gallistel, C. R., Krishan, M., Liu, Y., Miller, R., and Latham, P. E. (2014). The perception of probability. *Psychological review*, 121(1):96.

Gneiting, T. and Katzfuss, M. (2014). Probabilistic forecasting. *Annual Review of Statistics and Its Application*, 1:125–151.

Gneiting, T. and Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378.

Griffiths, T. L., Chater, N., Norris, D., and Pouget, A. (2012). How the bayesians got their beliefs (and what those beliefs actually are): Comment on bowers and davis (2012).

Griffiths, T. L. and Tenenbaum, J. B. (2006). Optimal predictions in everyday cognition. *Psychological science*, 17(9):767–773.

Holt, C. (1957). Forecasting trends and seasonals by exponentially weighted averages. carnegie institute of technology. Technical report, Pittsburgh ONR memorandum.

Kahneman, D. and Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, 47(2):263–291.

Knill, D. C. and Pouget, A. (2004). The bayesian brain: the role of uncertainty in neural coding and computation. *TRENDS in Neurosciences*, 27(12):712–719.

Kording, K. P. (2014). Bayesian statistics: relevant for the brain? *Current opinion in neurobiology*, 25:130–133.

Marr, D. (1982). Vision: A computational investigation into the human representation and processing of visual information. mit press. *Cambridge, Massachusetts*.

Nassar, M. R., Wilson, R. C., Heasly, B., and Gold, J. I. (2010). An approximately bayesian delta-rule model explains the dynamics of belief updating in a changing environment. *Journal of Neuroscience*, 30(37):12366–12378.

Peterson, C. R. and Beach, L. R. (1967). Man as an intuitive statistician. *Psychological bulletin*, 68(1):29.

Rakhlin, A. and Sridharan, K. (2015). Sequential probability assignment with binary alphabets and large classes of experts. *arXiv preprint arXiv:1501.07340*.

Ramachandran, V. (1990). Interactions between motion, depth, color and form: The utilitarian theory of perception. *Vision: Coding and efficiency*, pages 346–360.

Robinson, G. H. (1964). Continuous estimation of a time-varying probability. *Ergonomics*, 7(1):7–21.

Tauber, S., Navarro, D. J., Perfors, A., and Steyvers, M. (2017). Bayesian models of cognition revisited: Setting optimality aside and letting data drive psychological theory. *Psychological review*, 124 4:410–441.

Vovk, V. and Shafer, G. (2005). Good randomized sequential probability forecasting is always possible. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(5):747–763.

Vovk, V., Takemura, A., and Shafer, G. (2005). Defensive forecasting. *CoRR*, abs/cs/0505083.

Vovk, V. and Zhdanov, F. (2009). Prediction with expert advice for the brier game. *Journal of Machine Learning Research*, 10(Nov):2445–2471.

Weissman, T. and Merhav, N. (2001). Universal prediction of individual binary sequences in the presence of noise. *IEEE Transactions on Information Theory*, 47(6):2151–2173.

A Notations

This section describes the notations used in our algorithm. Some of the notations are the same as in the description of the algorithm in "The perception of probability'. Notations that are different have been explicitly indicated.

- D - the complete sequence of observed data for a particular subject in a single session. This is an array of 0 or 1 values with size 1000. 1 indicates a green ball and 0 indicates a red ball.
- CP - Indicates the latest changepoint. A changepoint is a numerical value that indicates the index in D where the algorithm predicts there was a change in the hidden probability. The penultimate changepoint is referred to as CP - 1 and so on.
- $D_{>CP}$ - Indicates the sequence of observations after the latest changepoint. In "The perception of probability", this is indicated by $D_{>n}$. Stored as a sequence (of variable length) of 0s and 1s.
- $D_{>CP\text{-}1}$ - Indicates the sequence of observations after the penultimate changepoint. In "The perception of probability", this is indicated by $D_{>n-1}$. Stored as a sequence (of variable length) of 0s and 1s.
- p_g - The hidden true probability of observing a green ball in the next trial, which is the proportion of green balls in the urn. The value of p_g remains unknown, and could be different for each trial
- $\hat{p}_g$ - The algorithm's estimate of the true probability of observing a green ball in the next trial. The value of $\hat{p}_g$ can be different for each trial
- p_c - The hidden probability of a change in p_g. In the "The perception of probability", this value is fixed at 0.05. The probability that the true proportion of green balls in the urn changes in a given trial follows a geometric distribution with mean 0.05.
- $\hat{p}_c$ - The algorithm's estimate of the probability of a change in p_g. Unlike the hidden value of p_c which remains fixed, the changepoint algorithm starts with a prior value on p_c and updates this value whenever it detects a change or "takes back" a previously determined change.
- p_o - The observed probability of green balls in a given sequence of trials.
- n_g - The number of green balls observed in a given sequence of trials.
- n - The total number of balls observed in a given sequence of trials. In other words, this is the total number of trials in a given sequence.
- KLCrit - The critical threshold for decision criteria 1. This is referred to in the "The perception of probability" as T_1. KLCrit is a hyper-parameter which is used to decide if the difference between our estimate $\hat{p}_g$ and the observed probability p_o using the current sequence of observations $D_{>CP}$ is high enough to warrant further investigation. One possible value is 0.23. This is the first of 4 inputs to the program.
- BFCrit - The critical threshold for decision criteria 2. This is referred to in the "The perception of probability" as T_2. BFCrit is the threshold to which the Bayes Factor between two models – a model where there has been 1 change in true probability (M_1), and a model in which there has been no changes in the true probability (M_0) – are compared. One possible value for BFCrit is 1. This is the second of 4 inputs to the program.
- N - The total number of trials the given subject has seen at a given point of time, across all sessions. In "The perception of probability", there is no consistent notation for this value.
- N_c - The total number of change points observed across all sessions for the given subject. In "The perception of probability", this is referred to as n_c.
- α_P , β_P - The initial hyper parameters for the beta distribution on the prior probability for p_g. The paper recommends using the Bayes-Laplace prior of 1 and 1. The vector of α_p and β_p is the third of 4 parameters to the program, and the value remains unchanged.
- α_C , β_C - The hyper parameters for the beta distribution on the prior probability for p_c. The paper recommends using the Jeffreys prior of 0.5 and 0.5. The vector of α_c and β_c is the fourth of 4 inputs for the program.
- α_p , β_p - The evolving hyper-parameters for the beta distribution on the prior probability of p_g. These prior values start with the initial user input for the program, but evolves after each changepoint.
- α_c , β_c - The evolving hyper-parameters for the beta distribution on the prior probability of p_c. These prior values start with the initial user input for the program, but evolves after each changepoint.

B
Changepoint Detection Algorithm

1. Set N = 0, $N_g = \alpha_C$
2. Set $D_{>CP\text{-}1}$ and $D_{>CP}$ as empty sequences
3. Set $\alpha_p = \alpha_P$, $\beta_p = \beta_P$, $\alpha_c = \alpha_C$ and $\beta_c = \beta_C$
4. Calculate
$$\hat{p_g} = \frac{\alpha_g}{\alpha_g + \beta_g}$$
and
$$\hat{p_c} = \frac{\alpha_c}{\alpha_c + \beta_c}$$
5. Make a copy of α_p and β_p. When α_p and β_p change, the copies will be updated with the penultimate values. This will be required in case we decide to remove the latest changepoint.
6. Initiate a Boolean variable detectedChange with FALSE. When we detect a change and update $\hat{p_g}$, we use this boolean flag to ignore the next trial for calculation. This is because our estimates are always for the next trial but they are based on the current trial. If we already have a new estimate for the next trial, we need not recalculate it for the next observation
7. Repeat the following for each trial in the given session.
8. Append the current observation (0 if red, 1 if green) to $D_{>CP}$.
9. Increment n_g with the value of observation. Increment n with 1
10. Calculate
$$p_o = \frac{n_g}{n}$$
11. Calculate the Kullback-Leibler divergence using the equation
$$KL = p_o log \frac{p_o}{\hat{p_g}} + (1 - p_o) log \frac{1 - p_o}{1 - \hat{p_g}}$$
12. Calculate the evidence of a change, E, using the equation
$$E = n * KL$$
, where n = $|D_{>CP}|$ is the length of the sequence in $D_{>CP}$.
13. Test the decision criteria for the first stage using the comparison
$$E > KLCrit$$
and detectedChange is FALSE

    At this point, the changepoint algorithm is trying to decide if the difference between our estimate of probability of seeing a green ball in the next trail and the actual proportion of green balls observed in the current trail is big enough to warrant further investigation. If the difference is less than the threshold, the algorithm is satisfied with the current estimate of probability of seeing a green ball, and continues to the next trail. If there was a change in estimate at the end of the previous trial, once again we need not change the estimate yet. In this case, proceed to step 4. If not, proceed to step 33

    If the difference is higher than the threshold, we investigate further. There could be 3 possibilities in this case. Briefly, here is how we proceed in each of the 3 cases. These are described in more detail as part of the algorithm.

    (a) There was a change in the true proportion of the green ball, p_g, since the last identified changepoint CP. In this case, we need to estimate a new changepoint in the sequence $D_{>CP}$.

    (b) Our estimate $\hat{p}_g$ at the previous changepoint was incorrect, but we are convinced there was a change. In other words, a change in true probability did occur, but we mis-estimated the extent of the change, and possibly the trial where the change occurred. This could happen if the previous estimate was made after seeing very few observations. To test this case, we temporarily remove CP and append $D_{>CP}$ with $D_{>CP\text{-}1}$. We then try to see if adding a new changepoint to the merged sequence of observations can explain the mis-estimate. If we find such a point, we make it the new change point, and recalculate $\hat{p}_g$.

(c) Our estimation of the previous changepoint itself was incorrect. In this case, we eliminate the latest changepoint and proceed with the penultimate changepoint. If the step (b) above does not yield a changepoint, we conclude that the changepoint that we removed temporarily can be ignored permanently.

14. Update the priors for $\hat{p}_c$ as
$$\alpha_c = \alpha_C + N_c$$
$$\beta_c = \beta_C + N - N_c$$
15. Recalculate $\hat{p}_c$ using
$$\hat{p_c} = \frac{\alpha_c}{\alpha_c + \beta_c}$$
16. Call the "Change-point detection algorithm" sub-routine with the parameters :
    (a) The priors for the current estimate of $\hat{p}_g$, α_p and β_p
    (b) The sequence of observations $D_{>CP}$
    (c) The current estimated of $\hat{p}_c$
17. Compare the posterior odds returned by the subroutine to BFCrit.
18. If
$$PosteriorOdds > BFCrit$$
, we proceed with the next step. This is case (a) described above. If not, go to step (24)
19. Increment N_c with 1.
20. Update the copies of priors α_p and β_p with the current value of the priors.
21. Update α_p and β_p with the values returned from the subroutine
22. The new estimate of $\hat{p}_g$ becomes
$$\hat{p_g} = \frac{\alpha_g}{\alpha_g + \beta_g}$$
23. Use the index of the changepoint returned by the subroutine to split the observed sequence into two parts. Replace $D_{>CP-1}$ with the first part and $D_{>CP}$ with the second part.
24. Mark detectedChange as TRUE
25. If there are no likely change points in the current sequence of observations, we need to verify cases (2) and (3). However, we need a penultimate sequence for this. Hence, check for the following condition
$$PosteriorOdds \leq BFCrit$$
and the number of changepoints in the current session is greater than 1. If so, proceed with the next step. If not, proceed to step ()
26. Call the "Change-point detection algorithm" sub-routine with the parameters :
    (a) The initial, fixed priors for the current estimate of $\hat{p}_G$, α_G and β_G
    (b) The combined sequence of observations $D_{>CP-1}$ and $D_{>CP}$
    (c) The current estimated of $\hat{p}_c$
27. Compare the posterior odds returned by the subroutine to BFCrit.
28. If
$$PosteriorOdds > BFCrit$$
, we proceed with the next step. This is case (b) described above. If not, go to step (34)
29. Update the copies of priors α_p and β_p with the current value of the priors.
30. Update α_p and β_p with the values returned from the subroutine
31. The new estimate of $\hat{p}_g$ becomes
$$\hat{p_g} = \frac{\alpha_g}{\alpha_g + \beta_g}$$
32. Use the index of the changepoint returned by the subroutine to split the combined sequence into two parts. Replace $D_{>CP-1}$ with the first part and $D_{>CP}$ with the second part.
33. Mark detectedChange as TRUE
34. is (28) is false, proceed to next step. If not, continue from (). This is case (c) described above

35. Decrement N_c with 1.
36. Update the priors for $\hat{p}_c$ as
$$\alpha_c = \alpha_C + N_c$$
37. Recalculate $\hat{p}_c$ using
$$\hat{p_c} = \frac{\alpha_c}{\alpha_c + \beta_c}$$
38. Empty $D_{>CP\text{-}1}$. Set $D_{>CP}$ as the combined sequence
39. Mark detectedChange as TRUE
40. Update α_p and β_p with the copies from the previous trial.
41. The new estimate of $\hat{p}_g$ becomes
$$\hat{p_g} = \frac{\alpha_g}{\alpha_g + \beta_g}$$
42. If 25 is false, there has been no change points detected until this point. Proceed with the following steps
43. Set
$$\alpha_p = \alpha_P + Number_greens_in_{D>CP}$$
and
$$\beta_p = \beta_P + Number_reds_in_{D>CP}$$
44. The new estimate of $\hat{p}_g$ becomes
$$\hat{p_g} = \frac{\alpha_g}{\alpha_g + \beta_g}$$
45. If (13) is false, set detectedChange as FALSE
46. Proceed to next observation.