Mon. Not. R. Astron. Soc. **000**, 1–5 (2014) Printed 9 September 2014 (MN LaTeX style file v2.2)

Space Warps Extended! Snappy Titles!

Phil Marshall,[1,2⋆] Aprajita Verma,[1] Anupreeta More,[3] Amit Kapadia,[4] Michael Parrish,[4] Chris Snyder,[4] Julianne Wilcox, Elisabeth Baeten, Christine Macmillan, Claude Cornen, Chris Davis,[2] Surhud More,[3] Michael Baumer,[2] Chris Lintott,[1] Robert Simpson,[1] David Miller,[4] Arfon Smith,[4] Edward Paget,[4] Prasenjit Saha,[5] Rafael Kueng,[5] Edwin Simpson,[1] Kelly Borden,[4] Tom Collett, Thomas Jennings, Matthias Tecza,[1] Layne Wright and possibly others

[1] *Dept. of Physics, University of Oxford, Keble Road, Oxford, OX1 3RH, UK*
[2] *Kavli Institute for Particle Astrophysics and Cosmology, Stanford University, 452 Lomita Mall, Stanford, CA 94035, USA*
[3] *Kavli IPMU (WPI), University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa 277-8583, Japan*
[4] *Adler Planetarium, Chicago, IL, USA*
[5] *Department of Physics, University of Zurich, Switzerland*

to be submitted to ?!?!

ABSTRACT

To Do (Chris): Do abstract!

Key words: gravitational lensing – methods: statistical – methods: citizen science

1 INTRODUCTION

2 FORMALISM

Space Warps keeps track of the following parameters:

- x_{ij}, the classification the i-th volunteer made of the j-th image. x_{ij} may take on three values: 0, 1, or empty. Since volunteers do not see most images, the vast majority of x_{ij} are blank.
- PD_i, the probability, given that the image is a dud, that i-th the volunteer will classify it as a dud. The probability, given that the image is a dud, that the volunteer will classify it as a lens follows as $1 - PD_i$.
- PL_i, the probability, given that the image is actually a lens, that the i-th volunteer will classify it as being a lens. The probability, given that the image is a lens, that the volunteer will classify it as a dud follows as $1 - PL_i$.
- p_j, the probability that the j-th image z_j is a lens given the current observations and skills of the volunteers who classified it.
- p^0, the prior probability that an object is a lens.

2.1 The Online System

In Space Warps, this is fixed at 2×10^{-4}, or the expectation that around 100 lenses will be found in 430,000 images. Because Space Warps is an online system that constantly reevaluates most of the above parameters (except p_0 and any non-blank x_{ij}) in order to promote likely lenses[1] or to retire likely duds,[2] we augment p, PL, and PD as p_j^k, the evaluation of p_j at time k. Space Warps uses Bayes' Theorem to update $p_j^{(k+1)}$ for some new evaluation x_{ij}[3]:

$$p_j^{(k+1)} = \left(\frac{x_{ij} PL_i^k}{PL_i^k p_j^k + (1 - PD_i^k)(1 - p_j^k)} + \frac{(1 - x_{ij})(1 - PL_i^k)}{(1 - PL_i^k) p_j^k + PD_i^k (1 - p_j^k)} \right) p_j^k \ , \qquad (1)$$

The first term on the right hand side is the probability update for evaluating the object to be a lens, while the second

[1] Note that this does not change the probability that a volunteer will actually draw said image.
[2] Images whose probability of being a lens drops below a certain threshold are removed from the active dataset.
[3] There is no superscript for x_{ij} because each user only sees an image once.

⋆ pjm@slac.stanford.edu

term is the probability that the image is a lens if the volunteer evaluates it to be a dud. (For example, an obtuse volunteer who always perfectly incorrectly classifies an image will actually change the probability exactly the same as one who always perfectly correctly classifies an image, given that the estimate of the obtuse volunteer's skill ($PL_i = 0$) is accurate.)

Space Warps only updates the volunteer's PL_i and PD_i after volunteer x_i classifies a training image:

$$PL_i^{(k+1)} = \frac{PL_i^k(NL_i^k + M) + \mathbb{I}[x_{ij} = z_j]}{NL_i^k + M + 1} \quad (2)$$

$$PD_i^{(k+1)} = \frac{PD_i^k(ND_i^k + M) + \mathbb{I}[x_{ij} = z_j]}{ND_i^k + M + 1} \quad (3)$$

where ND_i^k and NL_i^k refer to the number of training lenses and training duds observed by the i-th volunteer at time k, z_j refers to the true state of the j-th image, and $M = 4$ is a smoothing factor empirically derived to smooth the skill classification of new volunteers.

With these update rules plus an initialization of $PD_i = PL_i = 0.5$ and $p^0 = 2 \times 10^{-4}$, the online update system is fully specified.

2.2 An Offline Expectation Maximization Approach

Using the above notation but expanding p^0 to p^0_{ij} (allowing, e.g. for the distribution of training images to differ for each volunteer, perhaps based on the number of images they have observed, or to allow a particular image to be more likely to be drawn), the complete log-likelihood for this model may be specified:

$$\begin{aligned}
&\mathrm{CLL}(x_{ij}, z_j, PL_i, PD_i, p^0_{ij}) \\
&= \sum_i \sum_{j\in\Omega_i} x_{ij} z_j \log PL_i + (1 - x_{ij}) z_j \log(1 - PL_i) \\
&+ (1 - x_{ij})(1 - z_j) \log PD_i + x_{ij}(1 - z_j) \log(1 - PD_i) \\
&\qquad + z_j \log p^0_{ij} + (1 - z_j) \log(1 - p^0_{ij}) \quad (4)
\end{aligned}$$

where Ω_i is the set of all images volunteer i has observed in Ω, the set of all images in the program.[4] We can use this complete log-likelihood to derive an offline expectation maximization algorithm for determining the lens probabilities, user skills, and lens priors.

2.2.1 E-Step

The E-Step is just taking the expected complete log-likelihood, or the expectation value over $P(\cdot \mid x, \phi)$. This is equivalent to replacing z_j with p_j:

$$p_j = \frac{1}{N_j} \sum_{i\in\Omega_j} P(z_j = 1 \mid x_{ij}; \Phi) = \frac{1}{N_j} \sum_{i\in\Omega_j} \frac{P(x_{ij} \mid z_j = 1; \Phi) P(z_j = 1; \Phi)}{P(x_{ij}; \Phi)} \quad (5)$$

$$= \frac{1}{N_j} \sum_{i\in\Omega_j} \frac{PL_i^{x_{ij}} (1 - PL_i)^{(1-x_{ij})} p^0_{ij}}{PL_i^{x_{ij}} (1 - PL_i)^{(1-x_{ij})} p^0_{ij} + PD_i^{(1-x_{ij})} (1 - PD_i)^{x_{ij}} (1 - p^0_{ij})} \quad (6)$$

where $i \in \Omega_j$ is now the set of classifications done on the j-th image and N_j is the number of classifications done on the j-th image. This makes sense: p^0_{ij} is just the prior likelihood of an image being a lens, while PL_i is how well we would have identified a lens as such.

2.2.2 M-Step

The M-Step is done by maximizing the expected complete log-likelihood with regard to the input parameters PD_i, PL_i, p^0_{ij}. Doing the maximization process, we find:

$$PL_i = \frac{\sum_{j\in\Omega_i} x_{ij} p_j}{\sum_{j\in\Omega_i} p_j} \quad (7)$$

$$PD_i = \frac{\sum_{j\in\Omega_i} (1 - x_{ij})(1 - p_j)}{\sum_{j\in\Omega_i} (1 - p_j)} \quad (8)$$

$$\begin{aligned}
p^0_{ij} &= p_j\ , & p^0_i &= \frac{\sum_{j\in\Omega_i} p_j}{\sum_{j\in\Omega_i} 1} \\
p^0_j &= p_j\ , & p^0 &= \frac{\sum_i \sum_{j\in\Omega_i} p_j}{\sum_i \sum_{j\in\Omega_i} 1}\ , \quad (9)
\end{aligned}$$

where the possible specializations of p_0 are also given. These mirror quite closely the online systems, except that skill is now assessed against the majority expectation of the probability of an image being a lens, instead of its true value.

In practice we have training images where p_j is known. In those cases we can use the true value when doing the M-Step.

Finally, we also include Laplace smoothing into the M-Step in order to handle pathologic cases, such as when users never identify any lenses ($\sum_{j\in\Omega_i} p_j = 0$). The estimators for PL_i and PD_i now become:

$$PL_i = \frac{M + \sum_{j\in\Omega_i} x_{ij} p_j}{2M + \sum_{j\in\Omega_i} p_j} \quad (10)$$

$$PD_i = \frac{M + \sum_{j\in\Omega_i} (1 - x_{ij})(1 - p_j)}{2M + \sum_{j\in\Omega_i} (1 - p_j)} \quad (11)$$

where for Laplace smoothing, $M = 1$.

We choose to specialize p^0 to vary with image, p^0_j, because images are taken out of the Space Warps system if they reach too low a probability, clearly changing the prior when we evaluate at the end of the run; training images also have a different prior on being a lens as well. Finally, if the image is a training image with known $p_j \in (0, 1)$, then the known p_j is used instead of the current estimate.

ACKNOWLEDGEMENTS

We thank all XXXmembers of the Space Warps community for their contributions to the project so far. A complete list

[4] Because the probability that a viewer views a given image (given it is a training or test image) is random, I choose to simply ignore the unobserved images.

of collaborators is given at... In particular we would like to recognise the efforts of XXX, YYY etc in moderating the discussion.

We are also grateful to Brooke Simmons, David Hogg, XXX and YYY for many useful conversations about citizen science and gravitational lens detection. PJM was given support by the Royal Society, in the form of a research fellowship, and by the U.S. Department of Energy under contract number DE-AC02-76SF00515. AV acknowledges support from the Leverhulme Trust in the form of a research fellowship. The work of AM and SM was supported by World Premier International Research Center Initiative (WPI Initiative), MEXT, Japan. The SPACE WARPS project is open source. The web app was developed at https://github.com/Zooniverse/Lens-Zoo while the SWAP analysis software was developed at https://github.com/drphilmarshall/SpaceWarps.

APPENDIX A: PROBABILISTIC CLASSIFICATION ANALYSIS

Our aim is to enable the construction of a sample of good lens candidates. Since we aspire to making logical decisions, we define a "good candidate" as one which has a high posterior probability of being a lens, given the data: $\Pr(\mathrm{LENS}|\mathbf{d})$. Our problem is to approximate this probability. The data $\mathbf{d}$ in our case are the pixel values of a colour image. However, we can greatly compress these complex, noisy sets of data by asking each volunteer what they think about them. A complete classification in SPACE WARPS consists of a set of Marker positions, or none at all. The null set encodes the statement from the volunteer that the image in question is "NOT" a lens, while the placement of any Markers indicates that the volunteer considers this image to contain a "LENS". We simplify the problem by only using the Marker positions to assess whether the volunteer correctly assigned the classification "LENS" or "NOT" after viewing (blindly) a member of the training set of subjects.

How should we model these compressed data? The circumstances of each classification are quite complex, as are the human classifiers in general: the volunteers learn more about the problem as they go, but also inevitably make occasional mistakes (perhaps because a lens is difficult to see, or they became distracted during the task). To cope with this uncertainty, we assign a simple software *agent* to partner each volunteer. The agent's task is to interpret their volunteer's classification data as best it can, using a model that makes a number of necessary approximations. These interpretations will then include uncertainty arising as a result of the volunteer's efforts and also the agent's approximations, but they will have two important redeeming features. First, the interpretations will be quantitative (where before they were qualititative), and thus will be useful in decision-making. Second, the agent will be able to predict, using its model, the probability of a test subject being a LENS, given both its volunteer's classification, and its volunteer's experience. In this appendix we describe how these agents work, and other aspects of the SPACE WARPS analysis pipeline (SWAP).

A1 Agents and their Confusion Matrices

Each agent assumes that the probability of a volunteer recognising any given simulated lens as a lens is some number, $\Pr(\text{“LENS”}|\mathrm{LENS},\mathbf{d}^{\mathrm{t}})$, that depends only on what the volunteer is currently looking at, and all the previous training subjects they have seen (and not on what type of lens it is, how faint it is, what time it is, *etc.*). Likewise, it also assumes that the probability of a volunteer recognising any given dud image as a dud is some other number, $\Pr(\text{“NOT”}|\mathrm{NOT},\mathbf{d}^{\mathrm{t}})$, that also depends only on what the volunteer is currently looking at, and all the previous training subjects they have seen. These two probabilities define a 2 by 2 "confusion matrix," which the agent updates, every time a volunteer classifies a training subject, using the following very simple estimate:

$$\Pr(\text{“}X\text{”}|X,\mathbf{d}^{\mathrm{t}}) \approx \frac{N_{\text{“}X\text{”}}}{N_X}. \quad \text{(A1)}$$

Here, X stands for the true classification of the subject, *i.e.* either LENS or NOT, while "X'' is the corresponding classification made by the volunteer on viewing the subject. N_X is the number of lenses the volunteer has been shown, while $N_{\text{“}X\text{”}}$ is the number of times the volunteer got their classifications of this type of training subject right. $\mathbf{d}^{\mathrm{t}}$ stands for all $N_{\mathrm{LENS}}+N_{\mathrm{NOT}}$ training data that the agent has heard about to date.

The full confusion matrix of the k^{th} volunteer's agent is therefore:

$$\begin{aligned}\mathcal{M}^k &= \begin{bmatrix}\Pr(\text{“LENS”}|\mathrm{NOT},\mathbf{d}^{\mathrm{t}}_k) & \Pr(\text{“LENS”}|\mathrm{LENS},\mathbf{d}^{\mathrm{t}}_k)\\ \Pr(\text{“NOT”}|\mathrm{NOT},\mathbf{d}^{\mathrm{t}}_k) & \Pr(\text{“NOT”}|\mathrm{LENS},\mathbf{d}^{\mathrm{t}}_k)\end{bmatrix},\\ &= \begin{bmatrix}\mathcal{M}_{LN} & \mathcal{M}_{LL}\\ \mathcal{M}_{NN} & \mathcal{M}_{NL}\end{bmatrix}^k. \end{aligned} \quad \text{(A2)}$$

Note that these probabilities are normalized, such that $\Pr(\text{“NOT”}|\mathrm{NOT}) = 1 - \Pr(\text{“LENS”}|\mathrm{NOT})$.

Now, when this volunteer views a test subject, it is this confusion matrix that will allow their agent to update the probability of that test subject being a LENS. Let us suppose that this subject has never been seen before: the agent assigns a prior probability that it is (or contains) a lens is

$$\Pr(\mathrm{LENS}) = p_0 \quad \text{(A3)}$$

where we have to assign a value for p_0. In the CFHTLS, we might expect something like 100 lenses in 430,000 images, so $p_0 = 2 \times 10^{-4}$ is a reasonable estimate. The volunteer then makes a classification C_k (= "LENS" or "NOT"). We can apply Bayes' Theorem to derive how the agent should update this prior probability into a posterior one using this new information:

$$\Pr(\mathrm{LENS}|C_k,\mathbf{d}^{\mathrm{t}}_k) = \quad \text{(A4)}$$
$$\frac{\Pr(C_k|\mathrm{LENS},\mathbf{d}^{\mathrm{t}}_k)\cdot\Pr(\mathrm{LENS})}{[\Pr(C_k|\mathrm{LENS},\mathbf{d}^{\mathrm{t}}_k)\cdot\Pr(\mathrm{LENS})+\Pr(C_k|\mathrm{NOT},\mathbf{d}^{\mathrm{t}}_k)\cdot\Pr(\mathrm{NOT})]},$$

which can be evaluated numerically using the elements of the confusion matrix.

A2 Examples

Suppose we have a volunteer who is always right about the true nature of a training subject. Their agent's confusion matrix would be

$$\mathcal{M}^{\text{perfect}} = \begin{bmatrix} 0.0 & 1.0 \\ 1.0 & 0.0 \end{bmatrix}. \tag{A5}$$

On being given a fresh subject that actually is a LENS, this hypothetical volunteer would submit C = "LENS". Their agent would then calculate the posterior probability for the subject being a *LENS* to be

$$\Pr(\text{LENS}|\text{“LENS”}, \mathbf{d}^{\text{t}}_k) = \frac{1.0 \cdot p_0}{[1.0 \cdot p_0 + 0.0 \cdot (1 - p_0)]} \quad = 1.0, \tag{A6}$$

as we might expect for such a *perfect* classifier. Meanwhile, a hypothetical volunteer who (for some reason) wilfully always submits the wrong classification would have an agent with the column-swapped confusion matrix

$$\mathcal{M}^{\text{obtuse}} = \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}, \tag{A7}$$

and would submit C = "NOT" for this subject. However, such a volunteer would nevertheless be submitting useful information, since given the above confusion matrix, their agent would calculate

$$\Pr(\text{LENS}|\text{“NOT”}, T_k) = \frac{1.0 \cdot p_0}{[1.0 \cdot p_0 + 0.0 \cdot (1 - p_0)]} \quad = 1.0. \tag{A8}$$

Obtuse classifiers turn out to be as helpful as *perfect* ones.

A3 Online SWAP: Updating the Subject Probabilities

Suppose the $k{+}1^{\text{th}}$ volunteer now submits a classification, on the same subject just classified by the k^{th} volunteer. We can generalise Equation A4 by replacing the prior probability with the current posterior probability:

$$\Pr(\text{LENS}|C_{k+1}, \mathbf{d}^{\text{t}}{}_{k+1}, \mathbf{d}) = \tag{A9}$$

$$\frac{1}{Z}\Pr(C_{k+1}|\text{LENS}, \mathbf{d}^{\text{t}}{}_{k+1}) \cdot \Pr(\text{LENS}|\mathbf{d}) \tag{A10}$$

$$\text{where } Z = \Pr(C_{k+1}|\text{LENS}, \mathbf{d}^{\text{t}}{}_{k+1}) \cdot \Pr(\text{LENS}|\mathbf{d}) + \Pr(C_{k+1}|\text{NOT}, \mathbf{d}^{\text{t}}{}_{k+1}) \cdot \Pr(\text{NOT}|\mathbf{d}),$$

and $\mathbf{d} = \{C_k, \mathbf{d}^{\text{t}}_k\}$ is the set of all previous classifications, and the set of training subjects seen by each of those volunteers. $\Pr(\text{LENS}|\mathbf{d})$ is the fundamental property of each test subject that we are trying to infer. We track $\Pr(\text{LENS}|\mathbf{d})$ as a function of time, and by comparing it to a lower or upper thresholds, make decisions about whether to retire the subject from the classification interface or promote it in Talk, respectively.

A4 Information Gain per Classification, Agent "Skill" and "Contribution"

With an agent's confusion matrix in hand we can compute the *information* generated in any given classification. This will depend on the confusion matrix elements (Equation A2) but also on the probability of the subject being classified containing a lens. The quantity of interest is the relative entropy, or Kullback-Leiber divergence, between the prior and posterior probabilities for the possible truths T given the submitted classification C:

$$\begin{aligned}\Delta I =& \sum_T \Pr(T|C) \log_2 \frac{\Pr(T|C)}{\Pr(T)} \\ =& \Pr(\text{LENS}|C) \log_2 \frac{\Pr(C|\text{LENS})}{\Pr(C)} \\ &+ \Pr(\text{NOT}|C) \log_2 \frac{\Pr(C|\text{NOT})}{\Pr(C)},\end{aligned} \tag{A11}$$

where, as above, C can take the values "LENS" or "NOT". Substituting for the posterior probabilities using Equation A4 we get an expression that just depends on the elements of the confusion matrix $\mathcal{M}$ and the pre-classification subject lens probability $\Pr(\text{LENS}) = p$:

$$\begin{aligned}\Delta I =& p\frac{\mathcal{M}_{CL}}{p_c} \log_2 \frac{\mathcal{M}_{CL}}{p_c} \\ &+ (1-p)\frac{\mathcal{M}_{CN}}{p_c} \log_2 \frac{\mathcal{M}_{CN}}{p_c},\end{aligned} \tag{A12}$$

where the common denominator $p_c = p\mathcal{M}_{CL} + (1-p)\mathcal{M}_{CN}$. This expression has many interesting features. If p is either zero or one, $\Delta I(C) = 0$ regardless of the value of C or the values of the confusion matrix elements: if we know the subject's status with certainty, additional classifications supply no new information. If we set p to be the prior probability, Equation A12 tells us how much information is generated by classifying it all the way to $p = 1$ (which a perfect classifier, with $\mathcal{M}_{LL} = \mathcal{M}_{NN} = 1$, can do in a single classification). For a prior probability of 2×10^{-4}, 12.3 bits are generated in such a "detection." Conversely, only 0.0003 bits are generated during the rejection of a subject with the same prior: we are already fairly sure that each subject does not contain a lens! Imperfect classifiers (with $\mathcal{M}_{LL}$ and $\mathcal{M}_{NN}$ both less than 1) generate less than these maximum amounts of information each classification; the only classifiers that generate zero information are those that have $\mathcal{M}_{LL} = 1 - \mathcal{M}_{NN}$ (or equivalently, $\mathcal{M}_{CL} = \mathcal{M}_{CN}$ for all values of C). We might label such classifiers as "random", since they are as likely to classify a subject as a "LENS" no matter the true content of that subject.

Equation A12 suggests a useful information theoretical definition of the classifier skill perceived by the agent. At a fixed value of p, we can take the expectation value of the information gain ΔI over the possible classifications that could be made:

$$\begin{aligned}\langle \Delta I \rangle =& \sum_C \sum_T \Pr(T|C)\Pr(C) \log_2 \frac{\Pr(T|C)}{\Pr(T)} \\ =& -\sum_T \Pr(T) \log_2 \Pr(T) \\ &+ \sum_C \Pr(C) \sum_T \Pr(T|C) \log_2 \Pr(T|C) \\ =& p\left[\mathcal{S}(\mathcal{M}_{LL}) + \mathcal{S}(1-\mathcal{M}_{LL})\right] \\ &+ (1-p)\left[\mathcal{S}(\mathcal{M}_{NN}) + \mathcal{S}(1-\mathcal{M}_{NN})\right] \\ &- \mathcal{S}\left[p\mathcal{M}_{LL} + (1-p)(1-\mathcal{M}_{NN})\right] \\ &- \mathcal{S}\left[p(1-\mathcal{M}_{LL}) + (1-p)\mathcal{M}_{NN}\right]\end{aligned} \tag{A13}$$

where $\mathcal{S}(x) = x \log_2 x$. If we choose to evaluate $\langle \Delta I \rangle$ at $p = 0.5$, the result has some useful properties. While random classifiers presented with $p = 0.5$ subjects have $\langle \Delta I \rangle_{0.5} = 0.0$ as expected, perfect classifiers appear to the agents to have $\langle \Delta I \rangle_{0.5} = 1.0$. This suggests that $\langle \Delta I \rangle_{0.5}$, the amount of information we expect to gain when a classifier is presented with a 50-50 subject, is a reasonable quantification of *normalised skill*. A consequence of this choice is that the integrated skill (over all agents' histories) should come out to be approximately equal to the number of subjects in the survey, when the search is "complete" (and all subjects are fully classified). Therefore, a particular agent's integrated skill is a reasonable measure of that classifier's *contribution* to the lens search.

We conservatively initialize both elements of each agent's confusion matrix to be $\mathcal{M}^0_{LL} = \mathcal{M}^0_{NN} = 0.5$, that of a maximally ambivalent random classifier, so that all agents start with zero skill. While this makes no allowance for volunteers that actually do have previous experience of what gravitational lenses look like, we might expect it to help mitigate against false positives. Anyone who classifies more than one image (by progressing beyond the tutorial) makes a non-zero information contribution to the project.

The total information generated during the CFHTLS project is shown in Table **??**. Interpreting these numbers is not easy, but we might do the following. Dividing this by the amount of information it takes to classify a SPACE WARPS subject all the way to the detection threshold (lens probability 0.95), and then multiplying by the survey inefficiency gives us a very rough estimate for the effective number of detections corresponding to the crowd's contribution: these are 2830 and 25 bits for stages 1 and 2 respectively. These figures are close to the numbers of detections given in column 7 of the table.

A5 Uncertainty in the Agent Confusion Matrices

Finally, the confusion matrix obtained from the application of Equation A1 has some inherent noise which reduces as the number of training subjects classified by the agent's volunteer increases. For simplicity, the discussion has thus far assumed the case when the confusion matrix is known perfectly; in practice, we allow for uncertainty in the agent confusion matrices by averaging over a small number of samples drawn from Binomial distributions characterised by the matrix elements $\Pr(C_k|\mathrm{LENS}, \mathbf{d}^{\mathrm{t}}_k)$ and $\Pr(C_k|\mathrm{NOT}, \mathbf{d}^{\mathrm{t}}_k)$. The associated standard deviation in the estimated subject probability provides an error bar for this quantity.

A6 Offline SWAP

The probabilistic model described above does not need to be implemented as an online inference. Indeed, it might be more appropriate to perform the inference of all Agent confusion matrix elements and Subject probabilities simultaneously, so that the early classifications are not effectively downweighted as a result of the Agent's ignorance. It might also be that this ignorance builds in some conservatism to the system, reducing the noise due the early classifications if they are unreliable. In the joint analysis, the basic assumption that is built into the Agents, that their volunteers have innate and unchanging talent for lens spotting parameterised by two constant confusion matrix elements which simply need to be inferred given the data, is implemented in full. The effect is that of applying the time-averaged confusion matrices, rather than one that evolves as the Agents (and in the real world, the volunteers) learn.

The mathematics of the offline inference are presented elsewhere (in preparation). Here we briefly note that we maximize the joint posterior probability distribution for all the model parameters (some 66,000 confusion matrix elements and 430,000 subject probabilities) with a simple expectation-maximisation algorithm. This procedure takes approximately the same CPU time as the stage 2 online analysis, because no matrix inversions are required in the algorithm. The algorithm scales well and is actually faster than the online analysis with the larger stage 1 dataset. The expectation-maximisation algorithm is robust to initial starting parameters in, e.g., initial Agent confusion matrix elements and Subject probabilities. The plots in Section **??** show the difference in performance between the online and offline analyses.

This paper has been typeset from a TEX/ LATEX file prepared by the author.