

Implementing a Three-step Expert-elicitation Approach to Inform Ecological Decision-making

Karl A. Lamothe

Ontario and Prairie Region
Fisheries and Oceans Canada
867 Lakeshore Road
Burlington, ON
L7S 1A1

2026

**Canadian Manuscript Report of
Fisheries and Aquatic Sciences 3333**



Fisheries and Oceans
Canada

Pêches et Océans
Canada

Canada

Canadian Manuscript Report of Fisheries and Aquatic Sciences

Manuscript reports contain scientific and technical information that contributes to existing knowledge, but which deals with national or regional problems. Distribution is restricted to institutions or individuals located in particular regions of Canada. However, no restriction is placed on subject matter, and the series reflects the broad interests and policies of Fisheries and Oceans Canada, namely, fisheries and aquatic sciences.

Manuscript reports may be cited as full publications. The correct citation appears above the abstract of each report. Each report is abstracted in the data base *Aquatic Sciences and Fisheries Abstracts*.

Manuscript reports are produced regionally but are numbered nationally. Requests for individual reports will be filled by the issuing establishment listed on the front cover and title page.

Numbers 1-900 in this series were issued as Manuscript Reports (Biological Series) of the Biological Board of Canada, and subsequent to 1937 when the name of the Board was changed by Act of Parliament, as Manuscript Reports (Biological Series) of the Fisheries Research Board of Canada. Numbers 1426 - 1550 were issued as Department of Fisheries and Environment, Fisheries and Marine Service Manuscript Reports. The current series name was changed with report number 1551.

Rapport manuscrit canadien des sciences halieutiques et aquatiques

Les rapports manuscrits contiennent des renseignements scientifiques et techniques qui constituent une contribution aux connaissances actuelles, mais qui traitent de problèmes nationaux ou régionaux. La distribution en est limitée aux organismes et aux personnes de régions particulières du Canada. Il n'y a aucune restriction quant au sujet; de fait, la série reflète la vaste gamme des intérêts et des politiques de Pêches et Océans Canada, c'est-à-dire les sciences halieutiques et aquatiques.

Les rapports manuscrits peuvent être cités comme des publications à part entière. Le titre exact figure au-dessus du résumé de chaque rapport. Les rapports manuscrits sont résumés dans la base de données *Résumés des sciences aquatiques et halieutiques*.

Les rapports manuscrits sont produits à l'échelon régional, mais numérotés à l'échelon national. Les demandes de rapports seront satisfaites par l'établissement auteur dont le nom figure sur la couverture et la page du titre.

Les numéros 1 à 900 de cette série ont été publiés à titre de Manuscrits (série biologique) de l'Office de biologie du Canada, et après le changement de la désignation de cet organisme par décret du Parlement, en 1937, ont été classés comme Manuscrits (série biologique) de l'Office des recherches sur les pêcheries du Canada. Les numéros 901 à 1425 ont été publiés à titre de Rapports manuscrits de l'Office des recherches sur les pêcheries du Canada. Les numéros 1426 à 1550 sont parus à titre de Rapports manuscrits du Service des pêches et de la mer, ministère des Pêches et de l'Environnement. Le nom actuel de la série a été établi lors de la parution du numéro 1551.

Canadian Manuscript Report of
Fisheries and Aquatic Sciences 3333

2026

Implementing a Three-step Expert-elicitation Approach to Inform Ecological Decision-
making

Karl A. Lamothe

Ontario and Prairie Region
Fisheries and Oceans Canada
867 Lakeshore Road
Burlington, ON L7S 1A1

© His Majesty the King in Right of Canada, as represented by the Minister of the Department of Fisheries and Oceans, 2026.

This work is licensed under the [Open Government Licence](#)

Cat. No. Fs97-4/3333E-PDF ISBN 978-0-662-35315-7 ISSN 1488-5387

<https://doi.org/10.60825/b7sr-pv60>

Correct citation for this publication:

Lamothe, K.A. 2026. Implementing a Three-step Expert-elicitation Approach to Inform Ecological Decision-making. Can. Manuscr. Rep. Fish. Aquat. Sci. 3333: vii + 33 p.
<https://doi.org/10.60825/b7sr-pv60>

TABLE OF CONTENTS

LIST OF TABLES.....	iv
LIST OF FIGURES	v
ABSTRACT.....	vi
RÉSUMÉ	vii
INTRODUCTION	1
METHODS.....	2
Expert-elicitation Quantification.....	2
Expert-elicitation Tools	5
RESULTS	6
DISCUSSION.....	12
ACKNOWLEDGEMENTS	13
REFERENCES	14
SUPPLEMENTAL MATERIAL.....	16
APPENDIX 1	18
Functions.....	19
Themes (themes.R)	19
Helper function (helper.R)	20
Plotting functions (plotting_functions.R)	21
Summarize PERT distributions (summarize_pert.R)	23
Data	25
Data (demo_data.R)	25
Application.....	26
User interface (UI.R)	26
Server (server.R)	27
Application (app.R)	33

LIST OF TABLES

Table 1. Demo example summary table from the Shiny application for questions 1–8. Hard Union LPP = lowest plausible probability across all participants per question; Hard Union HPP = highest plausible probability across all participants per question. Nsim = number of mixture samples. Mixture 5th and 95th indicate the 90% confidence interval of mixture samples. N participants = number of participants that answered the question.11

LIST OF FIGURES

Figure 1. The IDEA protocol (Investigate, Discuss, Estimate, Aggregate), modified from Hemming et al. (2018) and Burgman (2015).	1
Figure 2. Responses from three participants (colours) to a question asked during a three-step expert-elicitation process. (Top) Error bars represent the lowest to highest plausible probabilities, and the points represent the best-guess probability for each participant. (Bottom) PERT distributions for each participant generated using three different λ values (line types). Values beyond the lowest and highest plausible probabilities have 0 probability of occurrence. 3	3
Figure 3. A) Participant responses to a question, including their lowest plausible, best-guess, and highest plausible probabilities in square brackets. Lowest and highest plausible probabilities form the confidence intervals and each point represents the best-guess probability. B) Participant responses converted to PERT distributions using $\lambda = 5$	4
Figure 4. A) Mixture distribution (black solid line) and moment-matched Beta distribution (black dashed line) overlaid on the participant responses (coloured lines). B) Cumulative distribution functions demonstrating participant responses (coloured lines), the empirical pooled distribution (black solid line), and the moment-matched Beta distribution (black dashed line).	5
Figure 5. Primary screen of the Shiny application prior to using the demo data or importing results.	7
Figure 6. Demo example figure from the Shiny application illustrating responses from six expert participants (y-axis) to questions 1–8. Each line segment represents the range from the lowest to highest plausible probabilities for each participant for each question, and points represent the best-guess probability. Raw scores are presented in Table S1.	8
Figure 7. Demo example figure from the Shiny application illustrating mixture distributions (black solid lines) and moment-matched Beta distributions (blue dashed lines) overlaid on the participant responses (coloured lines) for questions 1–8.	9
Figure 8. Demo example figure from the Shiny application illustrating histograms (grey bars) of the Monte Carlo sampling of participant distributions along with the pooled distribution function (black solid lines) and moment-matched Beta distribution functions (blue dashed lines) for questions 1–8.	9
Figure 9. Demo example figure from the Shiny application illustrating cumulative distribution functions of participant responses (coloured lines), empirical pooled distributions (black solid lines), and moment-matched Beta distributions (blue dashed lines) for questions 1–8. Note that these functions have significant overlap.	10

ABSTRACT

Lamothe, K.A. 2026. Implementing a Three-step Expert-elicitation Approach to Inform Ecological Decision-making. Can. Manuscr. Rep. Fish. Aquat. Sci. 3333: vii + 33 p.
<https://doi.org/10.60825/b7sr-pv60>

Fisheries and Oceans Canada routinely makes ecological management decisions under uncertainty. In many cases, incomplete monitoring and limited quantitative data constrain their ability to fully assess potential risks and outcomes of management decisions. While quantitative models and empirical data remain central to scientific advice, qualitative evidence, particularly expert judgement, can be critical for supporting defensible decisions. Structured expert-elicitation methods provide transparent and reproducible frameworks to formally incorporate expert judgement into decision-making while minimizing bias and explicitly representing uncertainty. This report describes a modified Delphi-based expert-elicitation approach developed for decision-support contexts. The approach combines a three-step probabilistic elicitation format, in which experts provide lowest plausible, best-guess, and highest plausible probability estimates, with structured aggregation of individual judgements using PERT distributions and an equal-weight linear opinion pool. This pooling approach retains disagreement among experts rather than forcing consensus and yields group-level probability distributions suitable for risk assessment and scenario evaluation. To support implementation and interpretation, a custom R Shiny application was developed to aggregate expert responses, generate visualizations, and summarize pooled uncertainty. Although initially applied to conservation translocation feasibility assessments, the Shiny application is broadly applicable across environmental management scenarios requiring timely decisions under uncertainty and limited data.

RÉSUMÉ

Lamothe, K.A. 2026. Implementing a Three-step Expert-elicitation Approach to Inform Ecological Decision-making. Can. Manuscr. Rep. Fish. Aquat. Sci. 3333: vii + 33 p.
<https://doi.org/10.60825/b7sr-pv60>

Pêches et Océans Canada prend régulièrement des décisions en matière de gestion écologique dans l'incertitude. Dans de nombreux cas, une surveillance incomplète et des données quantitatives limitées limitent leur capacité à évaluer pleinement les risques potentiels et les résultats des décisions de gestion. Bien que les modèles quantitatifs et les données empiriques demeurent au cœur des avis scientifiques, les preuves qualitatives, en particulier le jugement d'experts, peuvent être essentielles pour appuyer des décisions défendables. Les méthodes structurées de sollicitation d'experts fournissent des cadres transparents et reproductibles pour intégrer officiellement le jugement d'experts dans la prise de décision tout en minimisant les biais et en représentant explicitement l'incertitude. Ce rapport décrit une approche modifiée de sollicitation d'experts basée sur Delphi, développée pour des contextes d'aide à la décision. L'approche combine un format d'élucidation probabiliste en trois étapes, dans lequel les experts fournissent les estimations de probabilité les plus plausibles, les plus probables et les plus plausibles, avec une agrégation structurée des jugements individuels à l'aide de distributions PERT et d'un pool d'opinions linéaires à pondération égale. Cette approche de mise en commun permet de maintenir les désaccords entre les experts plutôt que de forcer le consensus et donne des distributions de probabilité au niveau du groupe qui conviennent à l'évaluation des risques et à l'évaluation des scénarios. Pour soutenir la mise en œuvre et l'interprétation, une application R Shiny personnalisée a été développée pour regrouper les réponses des experts, générer des visualisations et résumer l'incertitude regroupée. Bien qu'initialement appliquée aux évaluations de faisabilité de la translocation aux fins de la conservation, l'application Shiny est largement applicable à tous les scénarios de gestion environnementale nécessitant des décisions opportunes dans des conditions d'incertitude et de données limitées.

INTRODUCTION

Fisheries and Oceans Canada (DFO) routinely makes ecological management decisions under uncertainty. These decisions span a wide range of scenarios, from fisheries stock assessments and habitat alteration proposals to the implementation of recovery actions for species listed under the *Species at Risk Act* (Charles 1998; Minns and Moore 2003; Lamothe and Drake 2025). In many cases, decisions must be made despite incomplete data and limited monitoring coverage. In these contexts, environmental decision-making increasingly requires integrating multiple forms of evidence. Quantitative data such as population models and habitat monitoring remain central to scientific assessments, but for many species and ecosystems, particularly those that are rare or data-limited, quantitative information alone may be insufficient to fully inform decision-making. In these cases, qualitative evidence, including expert judgement informed by experience, field observations, and contextual knowledge, can be used to inform decisions.

Structured expert-elicitation approaches provide formal frameworks for incorporating expert judgement into decision-making (O’Hagan et al. 2006; Knol et al. 2010; Martin et al. 2011; Hemming et al. 2018; O’Hagan 2019). Unlike informal consultation, structured elicitation methods are designed to minimize cognitive bias and explicitly represent uncertainty using transparent, reproducible, and defensible protocols (O’Hagan 2019). When applied appropriately, expert elicitation can help translate dispersed knowledge into probabilistic statements that can be used for scenario evaluation and management planning. Among the available frameworks, the Delphi process and modified variants such as the IDEA protocol (“Investigate”, “Discuss”, “Estimate”, “Aggregate”; Figure 1) have become increasingly prominent in environmental and conservation applications (Mukherjee et al. 2015; Hemming et al. 2018; DFO 2025). In short, the Delphi method is a consensus-based approach that uses questionnaires and controlled feedback sequentially to reduce uncertainty and answer complex questions (DFO 2025). The process requires complete anonymity of the expert participants and is incomplete unless consensus is reached or a decision has been made to stop the process. Modifications of the Delphi method, such as the IDEA protocol and protocols used for a DFO Canadian Science Advisory Secretariat meeting in 2023 (DFO 2025), rely on iterative questioning, controlled feedback, and structured discussions to refine expert judgement estimates, quantify uncertainty, and document both convergence and persistent disagreement (Figure 1; Hemming et al. 2018; O’Hagan 2019). These modifications of the Delphi process emphasize describing the probability and uncertainty associated with specific outcomes rather than requiring consensus among participants, and relax the requirement for complete anonymity (Figure 1).

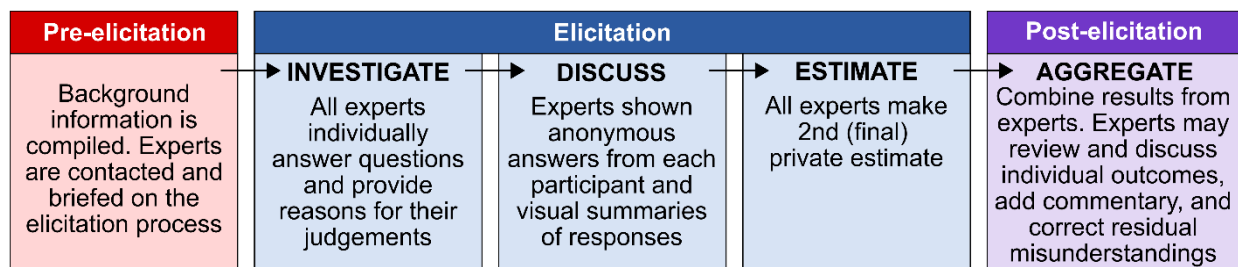


Figure 1. The IDEA protocol (Investigate, Discuss, Estimate, Aggregate), modified from Hemming et al. (2018) and Burgman (2015).

In this report, methods are described for implementing a modified Delphi process (IDEA protocol) within a DFO decision-support context, focusing on quantifying expert judgement for

discussions (Figure 1 - “DISCUSS”) and aggregating results for reporting (Figure 1 - “AGGREGATE”). The approach consists of a three-step elicitation format, the structured aggregation of individual judgements into group-level probability distributions, and the synthesis and visualization of results to support interpretation and decision-making using a custom R Shiny application. Emphasis was placed on representing uncertainty and preserving disagreement among experts rather than forcing consensus. Although originally developed to support conservation translocation feasibility assessments (e.g., Lamothe et al. 2023), this approach and tool can be used for other management scenarios that require timely decision-making on environmental decisions associated with incomplete data and uncertainty.

METHODS

EXPERT-ELICITATION QUANTIFICATION

An important step when implementing a modified Delphi process is developing clear, well-structured questions. In this context, I focus on questions that are framed probabilistically and can therefore be answered using the three-step expert-elicitation format (Hemming et al. 2018), a framework designed for situations where single event probabilities are of interest. For instance, experts might be asked to estimate the probability of a specific environmental event occurring within a specified time frame. One practical example comes from an expert elicitation conducted by DFO related to reintroduction objectives for the Eastern Sand Darter (*Ammocrypta pellucida*) in Ontario (Lamothe et al. 2023). In this meeting, experts were asked a series of questions, including: “*What is the probability that suitable abiotic conditions are available in sufficient quantity [in Big Otter Creek] to support 5,000 Eastern Sand Darter individuals?*” While not all questions lend themselves to probabilistic framing, many can be revised to allow for probability-based responses, which is the focus of this work. Framing questions in this way has notable advantages, including aligning with participant intuition and facilitating quantitative assessments in later stages, as further explored in the discussion.

In a three-step elicitation framework, each expert i provides three probability estimates for each question q : 1) a lowest plausible probability (L_{iq}), 2) a best-guess probability (M_{iq}), and 3) a highest plausible probability (U_{iq}). The lowest and highest plausible probabilities are treated as hard bounds, indicating that the expert judges the true probability to be highly unlikely to fall outside this interval. Individual expert probabilities are informed by the scope of information available to them, which can include quantitative data, model results, field observations, or any other information that the participant deems relevant to inform their judgement.

The belief of each participant about the event probability for question q , represented by the elicited triplet (L_{iq}, M_{iq}, U_{iq}), is modeled as a PERT (Program Evaluation and Review Technique) distribution defined on the interval $[L_{iq}, U_{iq}]$ (Clark 1962). Unlike a triangular distribution, the PERT distribution provides an approach to translate elicited minimum, most likely (i.e., best guess), and maximum values into a smooth probability distribution. Mathematically, it is equivalent to a Beta distribution rescaled to the interval $[L_{iq}, U_{iq}]$, with the mode equal to the best guess (M_{iq}) of the participant. Let X_{iq} follow a Beta distribution on $[0,1]$ with shape parameters α_{iq} and β_{iq} . The participant-specific probability variable is defined as:

$$P_{iq} = L_{iq} + (U_{iq} - L_{iq})X_{iq}$$

which ensures that all probability mass lies between the participant’s stated bounds. The Beta shape parameters are defined from the elicited triplet (L_{iq}, M_{iq}, U_{iq}) using a standard PERT parameterization:

$$\alpha_{iq} = 1 + \lambda \frac{M_{iq} - L_{iq}}{U_{iq} - L_{iq}} ; \beta_{iq} = 1 + \lambda \frac{U_{iq} - M_{iq}}{U_{iq} - L_{iq}}$$

where λ controls how strongly the distribution is concentrated around M_{iq} . A $\lambda = 5$ provides a moderate concentration without being overly narrow, but λ can be adjusted depending on the desired output (Figure 2). Figure 2 illustrates how the choice of λ influences the shape of the PERT distributions for responses that are right skewed (red), left skewed (blue), and symmetrical (grey). The point (M_{iq}) and error bars (L_{iq}, U_{iq}) on the top of Figure 2 represent the elicited triplet from three participants (A, B, C). As shown in the figure, each response is modeled as a unimodal distribution that places zero probability outside of $[L_{iq}, U_{iq}]$, peaks at M_{iq} , and reflects greater uncertainty when the elicited range $[L_{iq}, U_{iq}]$ is wide.

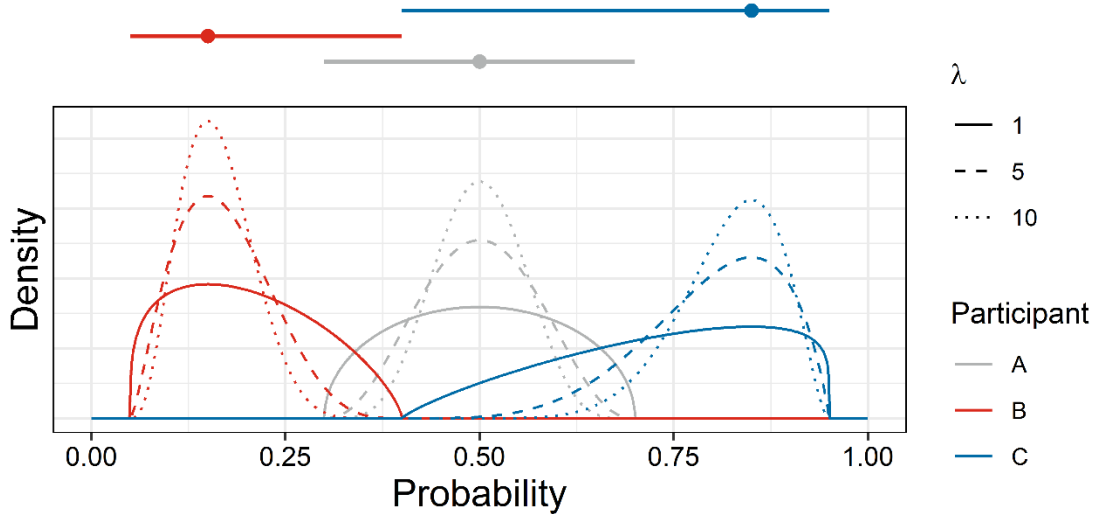


Figure 2. Responses from three participants (colours) to a question asked during a three-step expert-elicitation process. (Top) Error bars represent the lowest to highest plausible probabilities, and the points represent the best-guess probability for each participant. (Bottom) PERT distributions for each participant generated using three different λ values (line types). Values beyond the lowest and highest plausible probabilities have 0 probability of occurrence.

To illustrate the approach further, consider the example in which five participants were asked to estimate the probability of a specified ecological event occurring within a defined time frame. Each participant provided values for L_{iq} , M_{iq} , and U_{iq} which are summarized in Figure 3a. Individual PERT distributions constructed from these inputs using $\lambda = 5$ represent each participant's uncertainty about the event probability (Figure 3b). Participant A (pink), D (yellow), and E (orange) stated that M_{iq} was halfway between L_{iq} and U_{iq} (Figure 3a), which forms a bell-shaped curve (Figure 3b). However, M_{iq} was more proximate to L_{iq} than U_{iq} for Participant C (green; Figure 3a), forcing a slight right skew (Figure 3b). In all cases, the greatest probability mass lies nearest the M_{iq} and tapers off toward L_{iq} and U_{iq} (Figure 3b).

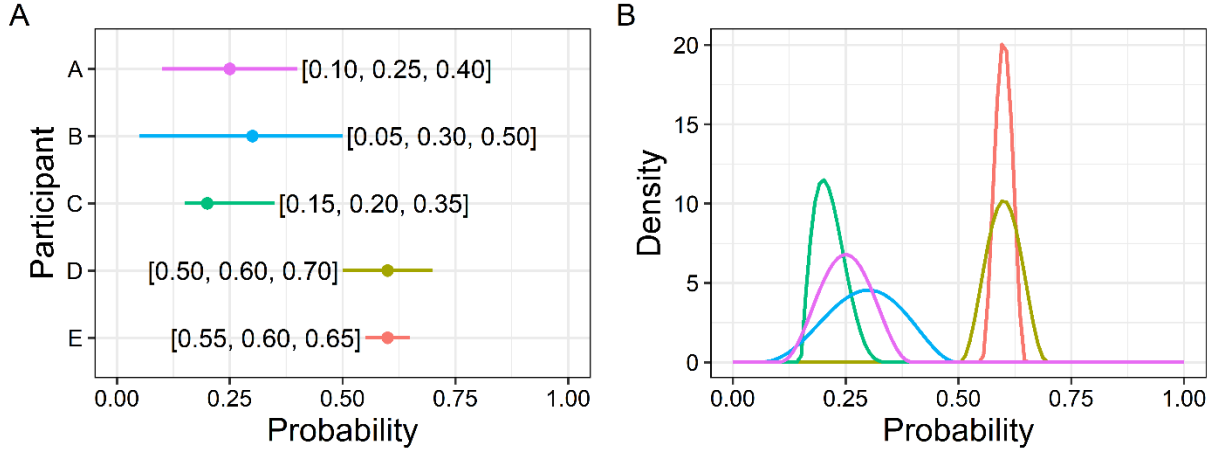


Figure 3. A) Participant responses to a question, including their lowest plausible, best-guess, and highest plausible probabilities in square brackets. Lowest and highest plausible probabilities form the confidence intervals and each point represents the best-guess probability. B) Participant responses converted to PERT distributions using $\lambda = 5$.

An equal-weight linear opinion pool can be used to combine expert opinions into a group-level estimate for each question (Genest and Zidek 1986). This approach combines multiple probability distributions (i.e., Figure 3b) into a single group-level distribution by averaging them directly, treating each distribution as an equally valid expression of uncertainty. Formally, if $f_{iq}(p)$ denotes the probability density function describing the opinion of expert i for question q , and K_q is the number of experts that answered the question, the pooled density is defined as:

$$f_q^{group}(p) = \frac{1}{K_q} \sum_{i=1}^{K_q} f_{iq}(p).$$

This produces a mixture distribution that preserves disagreement among participants such that if expert opinions cluster around different values, the pooled distribution may be multi-modal. The pooled distribution also respects the hard bounds provided by participants (i.e., $[L_{iq}, U_{iq}]$), as probability mass is only assigned within the union of individual plausible ranges. The linear opinion pool can be modified to weight the contribution of individual participants based on some level of expertise or other qualification (Hanea et al. 2018), which is discussed further in the discussion.

Group-level summaries can also be obtained via Monte Carlo sampling from the pooled distribution. For each draw, a participant is selected at random with equal probability and a probability value is sampled from that participant's PERT distribution. This procedure is then repeated n times (e.g., $n = 10,000$) for each question. An analytic version of the pooled cumulative distribution function (CDF) can be obtained by averaging individual CDFs and be used to verify that Monte Carlo results accurately reflect the underlying mixture distribution.

For visualization and compact reporting, a single moment-matched Beta distribution can be fitted to the pooled samples by matching the pooled sample mean ($\hat{\mu}_q$) and variance (\hat{v}_q). The fitted parameters (α_{iq}, β_{iq}) satisfy:

$$\alpha_{iq} = \hat{\mu}_q \left(\frac{\hat{\mu}_q(1-\hat{\mu}_q)}{\hat{v}_q} - 1 \right); \beta_{iq} = (1 - \hat{\mu}_q) \left(\frac{\hat{\mu}_q(1-\hat{\mu}_q)}{\hat{v}_q} - 1 \right).$$

The fitted Beta distribution provides a smooth, unimodal approximation to the pooled opinions. It should be noted that because it is a summary approximation, it does not necessarily follow the individual hard bounds. Figure 4a illustrates individual PERT distributions (coloured solid lines), the equal-weight pooled density function (black solid line), and the moment-matched Beta distribution (black dashed line) for the example responses in Figure 3. The corresponding CDFs compare individual expert judgement (coloured lines), the empirical pooled CDF derived from Monte Carlo sampling (black solid line), and the Beta approximation (black dashed line; Figure 4b).

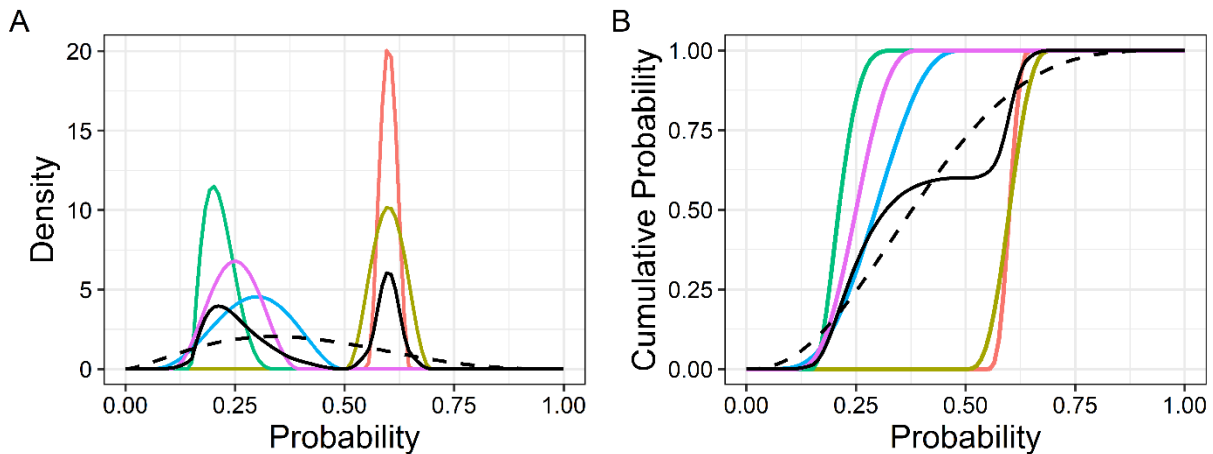


Figure 4. A) Mixture distribution (black solid line) and moment-matched Beta distribution (black dashed line) overlaid on the participant responses (coloured lines). B) Cumulative distribution functions demonstrating participant responses (coloured lines), the empirical pooled distribution (black solid line), and the moment-matched Beta distribution (black dashed line).

EXPERT-ELICITATION TOOLS

The expert-elicitation process can be implemented using different tools, but it generally requires software to distribute questionnaires and aggregate responses. In this case, I consider the use of a shared spreadsheet (e.g., SharePoint, Google Docs) to distribute questionnaires and a custom R Shiny application for aggregation and visualization.

A shared spreadsheet is first created in which individual worksheets (i.e., tabs) are prepared for each participant containing the following columns: “Question”, “Participant”, “Question Text”, “Lowest Plausible Probability”, “Best-Guess Probability”, “Highest Plausible Probability”, and “Notes”. Individual worksheet tabs can be labeled with the surname of each participant to ensure that the experts access the appropriate questionnaire. The first three columns of the questionnaire are pre-populated prior to the elicitation meeting with question numbers (“Question”), participant names (“Participant”), and questions to be answered by experts during the meeting (“Question Text”). The meeting facilitator has a unique master worksheet (i.e., tab) that is dynamically linked to all participant worksheets, allowing responses entered by participants to be automatically captured and consolidated (i.e., updated) in real time.

An R Shiny application was built to be used during elicitation meetings to aggregate expert judgement and to generate visualizations and summaries for each question. Included in these visualizations are participant response line plots (Figure 3a), PERT distribution functions (Figure 3b), pooled mixture distribution functions (Figure 4a), histograms of pooled samples with moment-matched Beta distributions, and CDFs comparing expert judgments, the empirical pooled CDF, and the Beta approximation (Figure 4b). In addition, a summary table can be

exported for each question, reporting the pooled mean and median probability, the 5th and 95th percentiles of the pooled distribution, and the hard union bounds, defined as $\min_i L_{iq}$ and $\max_i U_{iq}$. The application was developed in R Version 4.5.1 using packages from the tidyverse, including readr, dplyr, tidyr, purrr, and ggplot2 (Wickham 2016; Wickham et al. 2023, 2024, 2025a,b; Chang et al. 2025; Wickham and Henry 2025; R Core Team 2025).

During an expert-elicitation meeting, participants complete the questions in their individual questionnaire tab. Once all of the experts complete a particular question, the meeting facilitator can download their personal master worksheet tab and upload that document to the Shiny application. In the Results section, I describe the Shiny application user interface.

RESULTS

An R Shiny application was developed to implement the structured expert-elicitation process. The code for this application is provided in Appendix 1. The main page of the application is divided into two panels (Figure 5). On the left side is a vertically arranged sidebar that contains all user inputs and controls. At the top of the sidebar is a .CSV upload field, which corresponds to where the meeting facilitator uploads the raw participant responses. Below this is a checkbox option to use demo data instead of uploading a file, and several drop-down menus used to map columns in the dataset, including the question column, participant identifier, and the columns corresponding to the lowest plausible probability, best-guess probability, and highest plausible probability (Figure 5). Additional numeric inputs allow the meeting facilitator to specify the PERT shape parameter (λ ; default = 5) and the number of mixture draws (default = 10,000). A selector labeled “Questions to display” is below the input for mixture draws that allows the meeting facilitator to select the particular question, or all questions, to be displayed and assessed (Figure 5). Near the bottom of the sidebar is a blue “Run / Refresh” button, which runs the underlying code to generate the results. Below this are a series of buttons that facilitate downloads of a summary .CSV file and .PNG files, including plots of expert scores and PERT distributions, histograms of mixture samples, and CDF plots. The “All summaries (.ZIP)” file button allows the facilitator to download all summary products (i.e., figures and summary table; Figure 5).

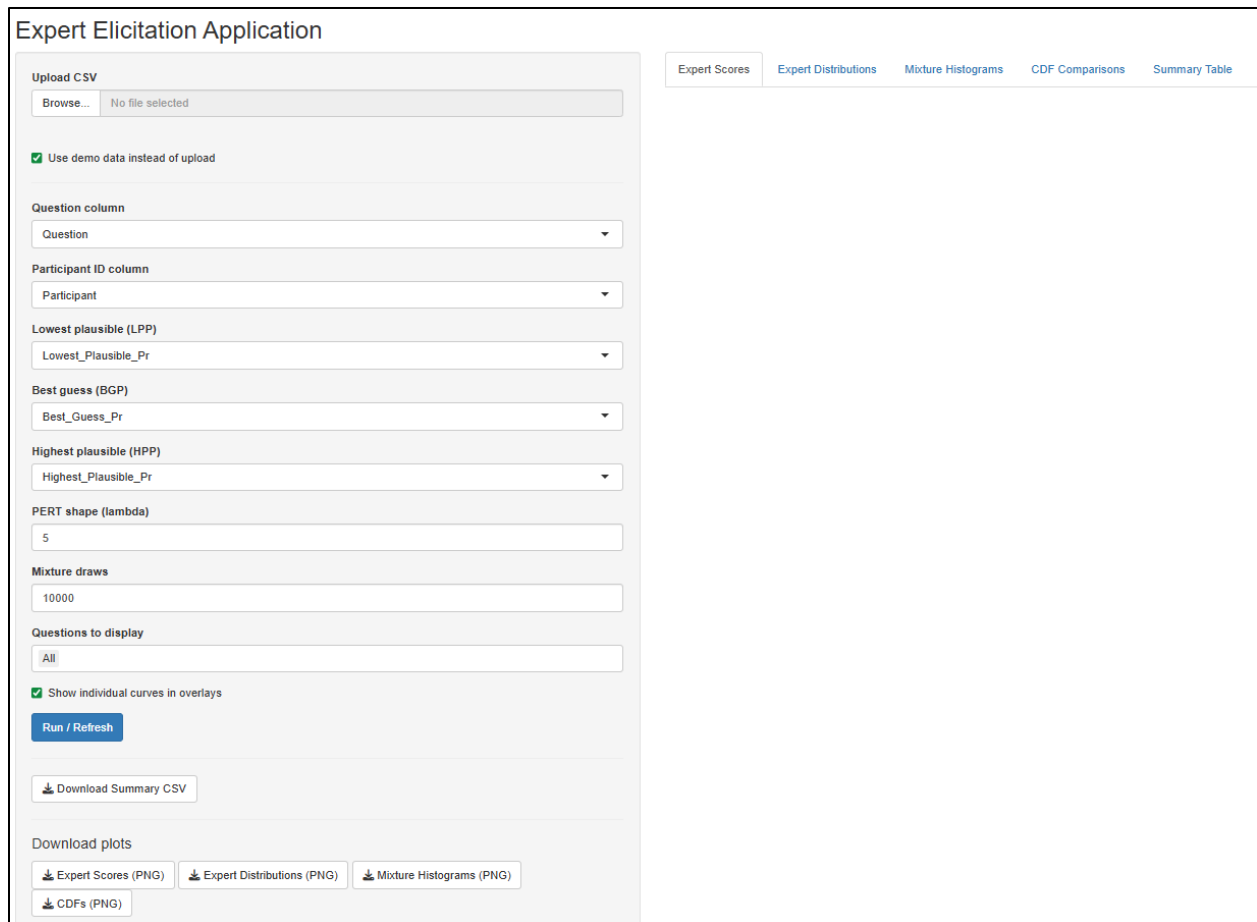


Figure 5. Primary screen of the Shiny application prior to using the demo data or importing results.

The right side of the application screen consists of the main output panel with a horizontal row of tabs across the top (Figure 5). These tabs are labeled “Expert Scores”, “Expert Distributions”, “Mixture Histograms”, “CDF Comparisons”, and “Summary Table,” indicating where the different visualizations and outputs are presented once the analysis is run. To use the application, the meeting facilitator either uploads a .CSV file by clicking the Browse button in the upper section of the left panel and locating the saved file, or selects the “Use demo data instead of upload” choice to explore the application (Figure 5).

For the remainder of the results, I present output using questions from the demo data, which can be replicated by selecting the “Use demo data instead of upload” choice and then running the application. Table S1 presents the demo data used in the application. The five tabs on the right side of the screen are populated after selecting “Use demo data instead of upload” and clicking the “Run / Refresh” button.

The first plots displayed after clicking the “Run / Refresh” button are line plots illustrating the lowest plausible probability, best-guess probability, and highest plausible probability for each participant for each selected question (Figure 6). There are no y-axis labels so as to retain anonymity among the participants. The second tab, “Expert Distributions”, contains density plots illustrating the question-specific pooled mixture distributions and moment-matched Beta distributions overlaid on the participant PERT distributions (Figure 7). Notice the bimodal response for Question 2, illustrating the difference between the mixture distribution and the

moment-matched Beta distribution overlay, and how the moment-matched Beta distribution does not respect the original bounds of participant PERT distributions (Figure 7). The third tab, “Mixture Histograms”, illustrates the Monte Carlo probability samples from the pooled distribution for each question, including the distribution function and moment-matched Beta distribution overlay (Figure 8). Figure 9 illustrates the output of the fourth tab, “CDF Comparisons”, which presents the CDFs of expert responses, empirical pooled distributions, and moment-matched Beta distributions for each selected question. The final tab presents a summary table for the questions being considered (Table 1).

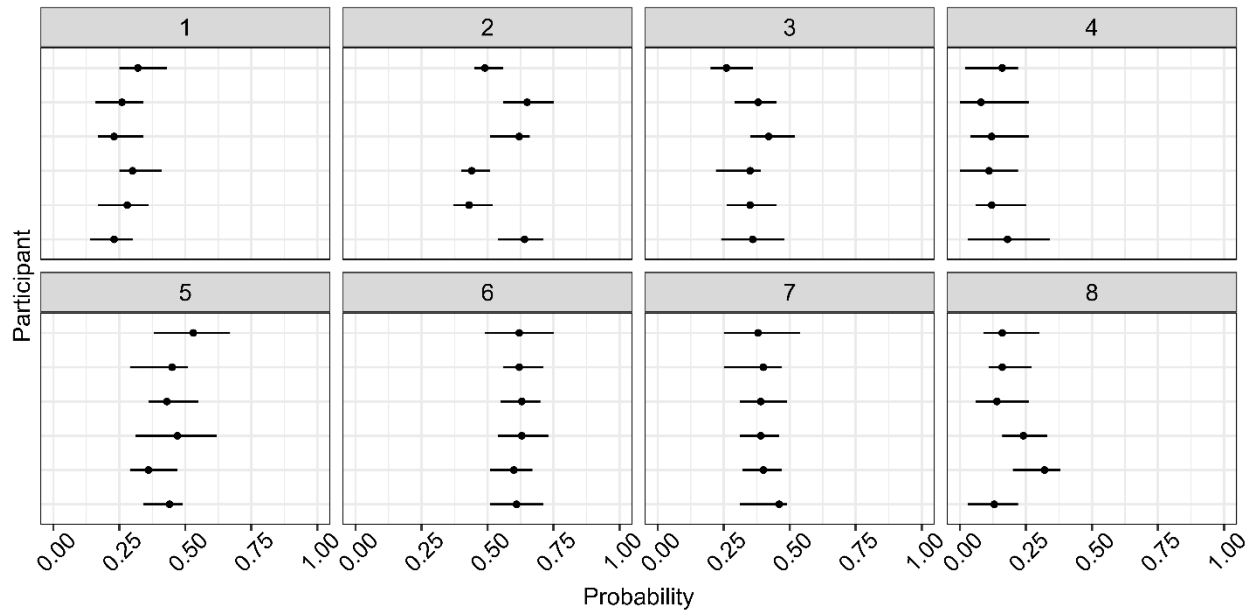


Figure 6. Demo example figure from the Shiny application illustrating responses from six expert participants (y-axis) to questions 1–8. Each line segment represents the range from the lowest to highest plausible probabilities for each participant for each question, and points represent the best-guess probability. Raw scores are presented in Table S1.

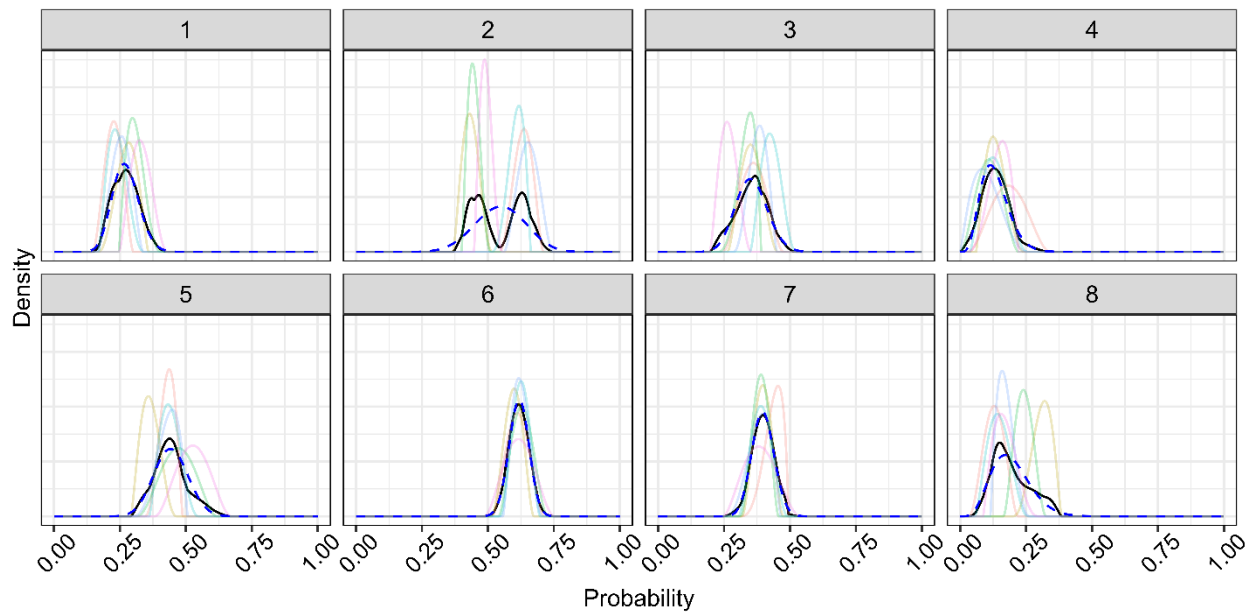


Figure 7. Demo example figure from the Shiny application illustrating mixture distributions (black solid lines) and moment-matched Beta distributions (blue dashed lines) overlaid on the participant responses (coloured lines) for questions 1–8.

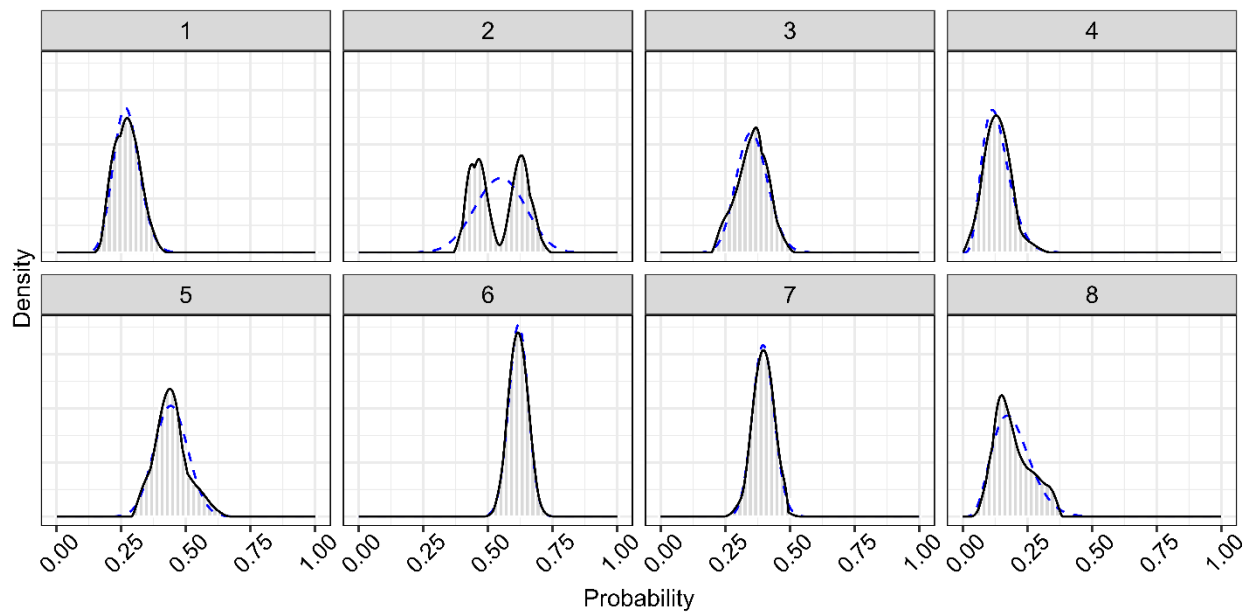


Figure 8. Demo example figure from the Shiny application illustrating histograms (grey bars) of the Monte Carlo sampling of participant distributions along with the pooled distribution function (black solid lines) and moment-matched Beta distribution functions (blue dashed lines) for questions 1–8.

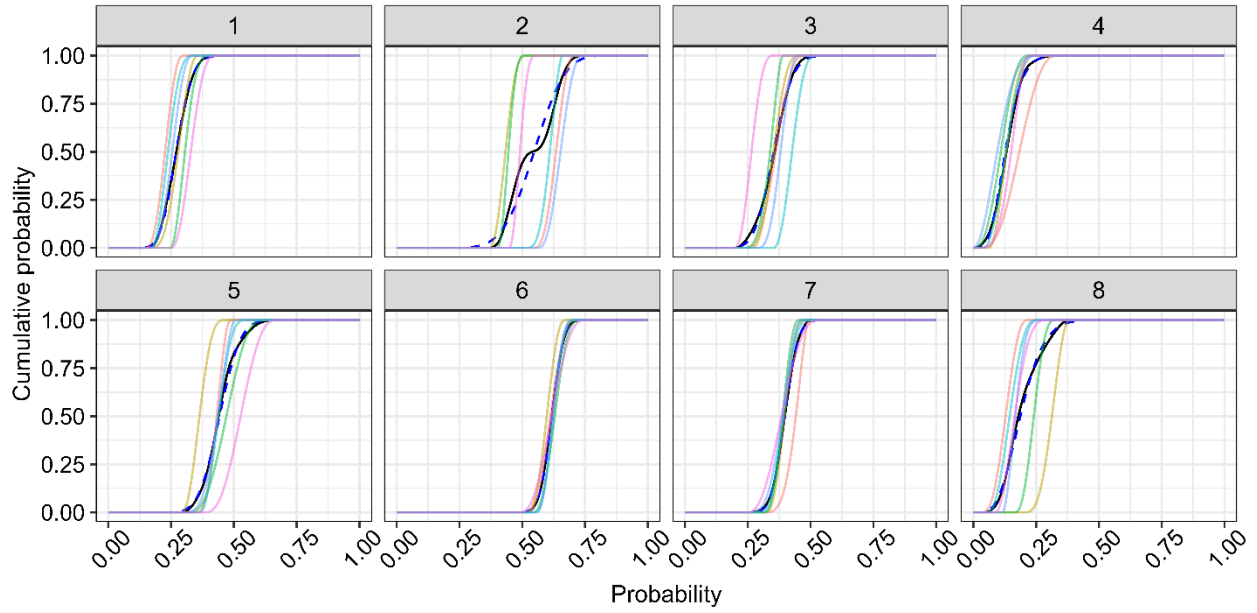


Figure 9. Demo example figure from the Shiny application illustrating cumulative distribution functions of participant responses (coloured lines), empirical pooled distributions (black solid lines), and moment-matched Beta distributions (blue dashed lines) for questions 1–8. Note that these functions have significant overlap.

Table 1. Demo example summary table from the Shiny application for questions 1–8. Hard Union LPP = lowest plausible probability across all participants per question; Hard Union HPP = highest plausible probability across all participants per question. Nsim = number of mixture samples. Mixture 5th and 95th indicate the 90% confidence interval of mixture samples. N participants = number of participants that answered the question.

Question	Pooled Mean	Pooled Median	Hard Union LPP	Hard Union HPP	Mixture 5th	Mixture 95th	N Participants	Lambda	Nsim
1	0.27	0.27	0.14	0.43	0.19	0.36	6	4	10000
2	0.55	0.55	0.37	0.75	0.41	0.68	6	4	10000
3	0.35	0.36	0.20	0.52	0.25	0.45	6	4	10000
4	0.13	0.13	0.00	0.34	0.05	0.22	6	4	10000
5	0.44	0.44	0.29	0.67	0.34	0.57	6	4	10000
6	0.62	0.62	0.49	0.75	0.56	0.68	6	4	10000
7	0.40	0.40	0.25	0.54	0.33	0.47	6	4	10000
8	0.19	0.18	0.03	0.38	0.09	0.33	6	4	10000

DISCUSSION

Resource management agencies regularly make decisions with imperfect information. Structured expert-elicitation approaches can be used to help inform these decisions while explicitly documenting associated uncertainty (O'Hagan et al. 2006; Knol et al. 2010; Martin et al. 2011; Hemming et al. 2018; O'Hagan 2019). Here, I presented a tool that can be used, as is or with minor modifications, to implement a three-part expert-elicitation process. This application delivers a pipeline for translating expert judgements into probability distributions, aggregates these distributions with an equal-weight linear opinion pool, and presents the results using visualizations and table summaries. In its current form, the tool is well-suited for decisions where evidence is heterogeneous or sparse, and when decision-makers value a reproducible workflow that considers uncertainty. Using PERT distributions with a hard-bounds interpretation ensures that expert judgements remain within clearly defined plausible ranges. The linear opinion pool then combines these judgements while preserving disagreement rather than forcing consensus. The interface also supports transparency by allowing users to trace results from raw participant inputs, through pooled densities and histograms, to CDF comparisons. Ultimately, this tool and its underlying code (Appendix 1), which depend on freely available, open-source software, can be used immediately or improved upon to include new features, which are considered in more detail below.

In this work, I chose to convert participant responses into probability distributions to support decision-making processes as this approach provides several benefits. First, while the three elicited values capture the range and central tendency of expert judgement, they do not describe how that judgement is distributed within the hard bounds. Representing these values as a probability distribution makes uncertainty explicit by indicating which outcomes are considered more or less plausible. Next, expressing responses as a distribution allows expert opinions to be combined coherently. Simple summary statistics such as the arithmetic mean, median, or range of the elicited probabilities can mask important patterns. Alternatively, aggregation methods such as linear opinion pools, Monte Carlo sampling, and CDFs operate naturally on probability distributions providing greater insight than individual summary values. Moreover, distributions support clearer visualization and communication of uncertainty and disagreement, while remaining true to the information provided by experts.

PERT distributions were chosen over the standard triangular distribution to reflect participant responses for several reasons. First, PERT distributions are smoother and less angular than triangular distributions, with gradually tapering tails. This smoothness avoids sharp changes in density at the mode and boundaries, which may overemphasize the “best-guess” value. Second, the PERT distribution allows control over how tightly probability mass is concentrated around the best-guess value through the shape parameter λ . Third, PERT distributions are mathematically equivalent to rescaled Beta distributions, making them convenient for aggregation and subsequent analysis (e.g., densities, cumulative distribution functions). Future work would benefit from a greater understanding of the effect of shape parameter choice on final summary statistics. Ultimately, PERT distributions behave more smoothly than triangular distributions.

Future modifications to the Shiny application could increase its utility. One potential modification relates to weighting. Equal weights are transparent and easy to interpret, but they assume that all experts are equally informed for each question. In situations where calibration data are available, such as responses to seed questions, performance-weighted pooling could improve both accuracy and credibility (Hanea et al. 2018). For example, Cooke-style weighting uses measures of statistical accuracy and informativeness to assign greater influence to better-performing experts (Cooke 1991). A second consideration is elicitation semantics. In the

current approach, lowest and highest plausible probabilities are interpreted as hard bounds. However, in some elicitation processes experts may intend these values to represent quantiles, such as the 10th and 90th percentiles (Hemming et al. 2018). Providing a project-level option to specify whether bounds or quantiles are being elicited, and fitting expert-specific distributions accordingly, would broaden the range of applications.

From a pooling perspective, some projects benefit from hierarchical or grouped pooling approaches. For example, expert judgements can first be combined within teams or shared information sources and then pooled across those groups (Wilson and Farrow 2018). This can help reduce the influence of dependence among experts who rely on the same data or experiences, which can skew the final results. Further modifications to the app could include robustness checks, such as showing trimmed mixtures that place less weight on extremely wide or extremely narrow intervals (i.e., overconfident or underconfident participants), or from sensitivity analyses that compare results under equal weighting, performance weighting, and different values of the shape parameter λ . In cases where decisions hinge on specific thresholds, decision-oriented summaries can be especially useful such as reporting the probability that a threshold is exceeded or not exceeded. These extensions could be introduced incrementally without disrupting the core workflow.

The tool and approach presented have limitations, many of which were recognized and accepted by design. First, equal weighting among participants is straightforward and transparent, but it can give disproportionate influence to experts whose judgements are correlated because they draw on similar information sources (Hanea et al. 2018). The use of PERT distributions also imposes a unimodal shape, which may not reflect beliefs that include heavy tails or more complex forms of uncertainty. In addition, interpreting the lowest and highest plausible probabilities as hard bounds can understate tail risk if participants instead intended these values to represent percentiles; hence clear communication with experts on how to answer questions is critically important. While mixture distributions represent disagreement among experts, they can be difficult to summarize with a single metric. The moment-matched Beta distribution helps address this challenge, but it does not enforce the original bounds provided by experts (e.g., Figures 7, 8). Finally, as with any elicitation exercise, cognitive biases may persist without procedural safeguards such as independent estimation, anonymity between rounds, and controlled feedback.

In summary, the application presented in this report represents the first version of a distribution-based approach to aggregating expert judgements in expert-elicitation processes. Future enhancements will add value by providing a clear choice between interpreting inputs as bounds or quantiles, allowing flexibility in distributional assumptions (i.e., participant and question specific λ), incorporating sensitivity analyses, and enabling grouped pooling when experts share information sources. As well, the ability to consider alternative styles of questions (e.g., discrete, ranks) could benefit future processes. Ultimately, the results of this work provide an open-source tool that can be used to better incorporate expert knowledge into ecological decision-making. This approach is intended to complement, not replace, empirical evidence, quantitative analyses, and established science advisory processes, particularly in situations where data are limited, uncertain, or incomplete.

ACKNOWLEDGEMENTS

I would like to thank Andrew Drake and Todd Morris for engaging conversations on the use of expert elicitation for decision-making that inspired this work and for reviewing an initial draft of this report. Funding was provided by Fisheries and Oceans Canada's Species at Risk Program.

REFERENCES

- Burgman, M.A. 2015. *Trusting Judgements: How to Get the Best Out of Experts*. Cambridge University Press, Cambridge, UK.
- Chang, W., Cheng, J., Allaire, J., Sievert, C., Schloerke, B., Xie, Y., Allen, J., McPherson, J., Dipert, A., and Borges, B. 2025. shiny: Web Application Framework for R. R package version 1.11.1. doi:10.32614/CRAN.package.shiny.
- Charles, A.T. 1998. Living with uncertainty in fisheries: analytical methods, management priorities and the Canadian groundfishery experience. *Fish. Res.* **37**(1–3): 37–50. doi:10.1016/S0165-7836(98)00125-8.
- Clark, C.E. 1962. The PERT model for the distribution of an activity. *Operat. Res.* **10**: 405–406.
- Cooke, R.M. 1991. *Experts in Uncertainty: Opinion and Subjective Probability in Science*. Environmental Ethics and Science Policy Series. Oxford University Press, Oxford, England.
- DFO. 2025. Evaluating the Ecological Benefits, Risks, and Feasibility of Reintroduction for Eastern Sand Darter (*Ammocrypta pellucida*) in Ontario. DFO Can. Sci. Advis. Sec. Sci. Advis. Rep. 2025/023.
- Genest, C., and Zidek, J.V. 1986. Combining probability distributions: a critique and an annotated bibliography. *Stat. Sci.* **1**(1): 114–148. doi:10.1214/ss/1177013825.
- Hanea, A.M., McBride, M.F., Burgman, M.A., and Wintle, B.C. 2018. The value of performance weights and discussion in aggregated expert judgements. *Risk Anal.* **38**(9): 1781–1794. doi:10.1111/risa.12992.
- Hemming, V., Burgman, M.A., Hanea, A.M., McBride, M.F., and Wintle, B.C. 2018. A practical guide to structure expert elicitation using the IDEA protocol. *Meth. Ecol. Evol.* **9**: 169–180. doi:10.1111/2041-210X.12857.
- Knol, A.B., Slottje, P., van der Sluijs, J.P., and Lebret, E. 2010. The use of expert elicitation in environmental health impact assessment: a seven step procedure. *Environ. Health* **9**: 19. doi:10.1186/1476-069X-9-19.
- Lamothe, K.A., and Drake, D.A.R. 2025. Considerations for the Design of a SARA-listed Freshwater Fish Monitoring Program in Point Pelee National Park. *Can. Manuscr. Rep. Fish. Aquat. Sci.* 3308: vi + 19 p. doi:10.60825/3d07-0q37.
- Lamothe, K.A., Morris, T.J., and Drake, D.A.R. 2023. Decision Support Framework for the Conservation Translocation of SARA-listed Freshwater Fishes and Mussels. DFO Can. Sci. Advis. Sec. Res. Doc. 2022/064. vii + 83 p.
- Lamothe, K.A., van der Lee, A.S., and Drake, D.A.R. 2026. Evaluating the Ecological Benefits, Risks, and Feasibility of Reintroduction for Eastern Sand Darter (*Ammocrypta pellucida*) in Ontario. DFO Can. Sci. Advis. Sec. Res. Doc. 2026/018. xi + 108 p.
- Martin, T.G., Burgman, M.A., Fidler, F., Kuhnert, P.M., Low-Choy, S., McBride, M., and Mengersen, K. 2011. Eliciting expert knowledge in conservation science. *Conserv. Biol.* **26**(1): 29–38. doi:10.1111/j.1523-1739.2011.01806.x.
- Minns, C.K., and Moore, J.E. 2003. Assessment of net change of productive capacity of fish habitats: the role of uncertainty and complexity in decision-making. *Can. J. Fish. Aquat. Sci.* **60**: 100–116. doi:10.1139/F02-168.

- Mukherjee, N., Hugé, J., Sutherland, W.J., McNeill, J., Van Opstal, M., Dahdouh-Guebas, F., and Koedam, N. 2015. The Delphi technique in ecology and biological conservation: applications and guidelines. *Meth. Ecol. Evol.* **6**(9): 1097–1109. doi:10.1111/2041-210X.12387.
- O'Hagan, A. 2019. Expert knowledge elicitation: subjective but scientific. *Am. Stat.* **73**(S1): 69–81. doi:10.1080/00031305.2018.1518265.
- O'Hagan, A., Buck, C.E., Daneshkhah, A., Eiser, J.R., Garthwaite, P.H., Jenkinson, D.J., Oakley, J.E., and Rakow, T. 2006. *Uncertain Judgements: Eliciting Expert Probabilities*. Wiley, Chichester, UK.
- R Core Team. 2025. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. <https://www.R-project.org/>.
- Wickham, H. 2016. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag, New York.
- Wickham, H., and Henry, L. 2025. *purrr: Functional Programming Tools*. R package version 1.1.0. doi:10.32614/CRAN.package.purrr.
- Wickham, H., François, R., Henry, L., Müller, K., and Vaughan, D. 2023. *dplyr: A Grammar of Data Manipulation*. R package version 1.1.4. doi:10.32614/CRAN.package.dplyr.
- Wickham, H., Vaughan, D., and Girlich, M. 2024. *tidyr: Tidy Messy Data*. R package version 1.3.1. doi:10.32614/CRAN.package.tidyr.
- Wickham, H., Hester, J., and Bryan, J. 2025a. *readr: Read Rectangular Text Data*. R package version 2.1.6. doi:10.32614/CRAN.package.readr.
- Wickham, H., Pedersen, T., and Seidel, D. 2025b. *scales: Scale Functions for Visualization*. R package version 1.4.0. doi:10.32614/CRAN.package.scales.
- Wilson, K.J., and Farrow, M. 2018. Combining judgements from correlated experts. *In Elicitation: The Science and Art of Structuring Judgement*. Edited by L.C. Dias, A. Morton, and J. Quigley. Springer Nature, Cham, Switzerland. pp. 211–240.

SUPPLEMENTAL MATERIAL

Table S1. Demo data used within the application.

Question	Participant	Lowest plausible probability	Best-guess probability	Highest plausible probability
1	P1	0.14	0.23	0.30
1	P2	0.17	0.28	0.36
1	P3	0.25	0.30	0.41
1	P4	0.17	0.23	0.34
1	P5	0.16	0.26	0.34
1	P6	0.25	0.32	0.43
2	P1	0.54	0.64	0.71
2	P2	0.37	0.43	0.52
2	P3	0.40	0.44	0.51
2	P4	0.51	0.62	0.66
2	P5	0.56	0.65	0.75
2	P6	0.45	0.49	0.56
3	P1	0.24	0.36	0.48
3	P2	0.26	0.35	0.45
3	P3	0.22	0.35	0.39
3	P4	0.35	0.42	0.52
3	P5	0.29	0.38	0.45
3	P6	0.20	0.26	0.36
4	P1	0.03	0.18	0.34
4	P2	0.06	0.12	0.25
4	P3	0.00	0.11	0.22
4	P4	0.04	0.12	0.26
4	P5	0.00	0.08	0.26
4	P6	0.02	0.16	0.22
5	P1	0.34	0.44	0.49
5	P2	0.29	0.36	0.47
5	P3	0.31	0.47	0.62
5	P4	0.36	0.43	0.55
5	P5	0.29	0.45	0.51
5	P6	0.38	0.53	0.67
6	P1	0.51	0.61	0.71
6	P2	0.51	0.60	0.67
6	P3	0.54	0.63	0.73
6	P4	0.55	0.63	0.70
6	P5	0.56	0.62	0.71
6	P6	0.49	0.62	0.75
7	P1	0.31	0.46	0.49
7	P2	0.32	0.40	0.47
7	P3	0.31	0.39	0.46
7	P4	0.31	0.39	0.49

Question	Participant	Lowest plausible probability	Best-guess probability	Highest plausible probability
7	P5	0.25	0.40	0.47
7	P6	0.25	0.38	0.54
8	P1	0.03	0.13	0.22
8	P2	0.20	0.32	0.38
8	P3	0.16	0.24	0.33
8	P4	0.06	0.14	0.26
8	P5	0.11	0.16	0.27
8	P6	0.09	0.16	0.30

APPENDIX 1

The code in this appendix is organized into eight R scripts, divided into three main groups:

Functions:

1. Themes (themes.R)
2. Helper function (helper.R)
3. Plotting functions (plotting_functions.R)
4. Summarize PERT distributions (summarize_pert.R)

Data:

5. Data (demo_data.R)

Application:

6. User interface (UI.R)
7. Server (server.R)
8. Application (app.R)

Each of these R scripts is accessible below.

FUNCTIONS

Themes (themes.R)

```
# List of packages being used
list.of.packages <- c('shiny', 'readr', 'dplyr', 'tidyr', 'purrr', 'ggplot2', 'scales')

# Identify packages in the list that are not on the computer
new.packages <- list.of.packages[!(list.of.packages %in% installed.packages()[,"Package"])]

# Install packages in "new.packages"
if(length(new.packages)) install.packages(new.packages); rm(list.of.packages);
  rm(new.packages)

# packages
suppressPackageStartupMessages({
  library(shiny)
  library(readr)
  library(dplyr)
  library(tidyr)
  library(purrr)
  library(ggplot2)
  library(scales)
})

options(scipen=999) # Remove scientific notation

#APP THEME (on-screen)
theme_app <- theme_bw() +
  theme(axis.title = element_text(size=24, family="sans", colour="black"),
        axis.text.x = element_text(size=20, family="sans", colour="black"),
        axis.text.y = element_text(size=20, family="sans", colour="black"),
        strip.text = element_text(size=20, family="sans", colour="black"),
        plot.title = element_text(size=24, family="sans", colour="black"),
        panel.border = element_rect(colour="black"),
        legend.position = "none")

# Apply it app-wide
theme_set(theme_app)

#EXPORT THEME (smaller text)
theme_export <- theme_bw() +
  theme(axis.title = element_text(size=11, family="sans", colour="black"),
        axis.text.x = element_text(size=10.5, angle=45, vjust=0.7, family="sans",
        colour="black"),
        axis.text.y = element_text(size=10.5, family="sans", colour="black"),
        strip.text = element_text(size=11, family="sans", colour="black"),
        plot.title = element_text(size=11, family="sans", colour="black"),
        panel.border = element_rect(colour="black"),
        legend.position = "none")
```

Helper function (helper.R)

```
# Choose a reasonable default for "Question" column if multiple exist
best_question_col <- function(df) {
  cols <- names(df)
  cand <- grep("^question(\\.{3}\\d+)?$", tolower(cols), value = TRUE)
  if (length(cand) == 0) {
    cand <- grep("^q(uestion)?", tolower(cols), value = TRUE)
  }
  if (length(cand) <= 1) {
    return(ifelse(length(cand) == 1, cand, "<none>"))
  }
  # Prefer the one with most non-missing + unique values
  score <- sapply(cand, function(cc) {
    v <- df[[cc]]
    sum(!is.na(v)) + 0.001 * length(unique(na.omit(v)))
  })
  cand[order(score, decreasing = TRUE)][1]
}
```

Plotting functions (plotting_functions.R)

```
# LPP / BGP / HPP per participant faceted by Question
build_individuals_plot <- function(df_raw,
                                  id_col, lpp_col, bgp_col, hpp_col,
                                  question_col = NULL,
                                  selected_questions = NULL,
                                  use_export_theme = FALSE,
                                  theme_export = NULL,
                                  facet_cols = 4) {
  # Ensure we have a Question column (or synthesize one)
  has_q <- !is.null(question_col) && !identical(question_col, "<none>")
  if (!has_q) {
    q_col <- "__Question__"
    df <- df_raw %>% mutate(`__Question__` = "Q1")
  } else {
    q_col <- question_col
    df <- df_raw
  }

  # Filter to the selected subset of questions (if provided)
  if (!is.null(selected_questions) && length(selected_questions) > 0) {
    # When app has an "All" option, remove it before filtering
    sel <- setdiff(selected_questions, "All")
    if (length(sel) > 0) {
      df <- df %>% filter(.data[[q_col]] %in% sel)
    }
  }

  # Build plotting frame using raw values
  dfp <- df %>%
    transmute(
      Question          = .data[[q_col]],
      Participant       = as.character(.data[[id_col]]),
      Lowest_Plausible_Pr = suppressWarnings(as.numeric(.data[[lpp_col]])),
      Best_Guess_Pr     = suppressWarnings(as.numeric(.data[[bgp_col]])),
      Highest_Plausible_Pr = suppressWarnings(as.numeric(.data[[hpp_col]]))
    )

  g <- ggplot() +
    geom_linerange(data = dfp, aes(x = Participant,
                                   ymin = Lowest_Plausible_Pr,
                                   ymax = Highest_Plausible_Pr), lwd = 0.5) +
    geom_point(data = dfp, aes(x = Participant, y = Best_Guess_Pr), size = 1) +
    coord_flip() + ylim(0,1)+
    labs(x = "Participant", y = "Probability") +
    facet_wrap(~ Question, ncol = facet_cols, scales = "fixed")+
    theme(axis.text.y = element_blank(),
          axis.ticks.y = element_blank())

  # Apply export theme
  if (isTRUE(use_export_theme) && !is.null(theme_export)) {
    g <- g + theme_export + theme(axis.text.y = element_blank(),
                                  axis.ticks.y = element_blank())
  }
  g
}

build_density_plot <- function(r, show_individual = TRUE, facet_cols = NULL) {
  p <- ggplot() +
    geom_line(data = r$dens_mix_all, aes(x = p, y = density), lwd = 0.5) +
```

```

  labs(x = "Probability", y = "Density")
  if (isTRUE(show_individual)) {
    p <- p +
      geom_line(data = r$dens_ind_all, aes(x = p, y = density, color = id),
                alpha = 0.5, lwd = 0.5) +
      guides(color = "none")
  }
  p <- p +
    geom_line(data = r$beta_all, aes(x = p, y = density),
              color = "blue", lwd = 0.5, lty = "dashed")
  if (is.null(facet_cols)) {
    p <- p + facet_wrap(~ Question, scales = "free_y")
  } else {
    p <- p + facet_wrap(~ Question, ncol = facet_cols, scales = "free_y")
  }
  p
}

build_hist_plot <- function(r, facet_cols = 4) {
  ggplot() +
    geom_histogram(data = r$samples_all,
                  aes(x = samples, y = after_stat(density)),
                  bins = 50, fill = "grey85", color = "white") +
    geom_line(data = r$beta_all, aes(x = p, y = density),
              color = "blue", lwd = 0.5, lty = "dashed") +
    geom_line(data = r$dens_mix_all, aes(x = p, y = density), lwd = 0.5) +
    labs(x = "Probability", y = "Density") +
    facet_wrap(~ Question, ncol = facet_cols) + theme_export +
    theme(axis.text.y = element_blank(), axis.ticks.y = element_blank())
}

build_cdf_plot <- function(r, show_individual = TRUE, facet_cols = 4) {
  p <- ggplot() +
    geom_line(data = r$cdf_mix_all, aes(p, cdf), lwd = 0.5) +
    geom_line(data = r$cdf_beta_all, aes(p, cdf),
              color = "blue", lwd = 0.5, lty = "dashed") +
    #geom_line(data = r$cdf_emp_all, aes(p, cdf),
    #          color = "darkgrey", lwd = 0.5, lty = "dotdash") +
    labs(x = "Probability", y = "Cumulative probability")

  if (isTRUE(show_individual)) {
    p <- p +
      geom_line(data = r$cdf_ind_all, aes(p, cdf, color = id),
                alpha = 0.5, lwd = 0.5) + guides(color = "none")
  }
  p + facet_wrap(~ Question, ncol = facet_cols) + theme_export
}

```

Summarize PERT distributions (summarize_pert.R)

```
summarize_question_pert <- function(df, id_col = "Participant",
                                   lpp_col = "Lowest_Plausible_Pr",
                                   bgp_col = "Best_Guess_Pr",
                                   hpp_col = "Highest_Plausible_Pr",
                                   lambda = 5, Nsim = 40000,
                                   grid = seq(0, 1, length.out = 1000),
                                   seed = NULL,
                                   question_label = NULL) {
  # Use a random seed if none provided
  if (is.null(seed)) seed <- sample.int(1e9, 1)
  set.seed(seed)

  # Standardize columns (string-safe)
  df2 <- df %>%
    transmute(
      id = .data[[id_col]],
      a = .data[[lpp_col]],
      m = .data[[bgp_col]],
      b = .data[[hpp_col]]
    ) %>%
    mutate(
      a = pmin(a, b),
      b = pmax(a, b),
      m = pmin(pmax(m, a + 1e-8), b - 1e-8),
      alpha = 1 + lambda * (m - a) / (b - a),
      beta = 1 + lambda * (b - m) / (b - a)
    )

  # Mixture simulation (equal weights)
  idx <- sample.int(nrow(df2), Nsim, replace = TRUE)
  samples <- df2$a[idx] + (df2$b[idx] - df2$a[idx]) * rbeta(Nsim, df2$alpha[idx],
df2$beta[idx])

  # Vectorized individual densities
  dens_individual <- df2 %>%
    select(id, a, b, alpha, beta) %>%
    crossing(p = grid) %>%
    mutate(
      density = if_else(
        p >= a & p <= b,
        dbeta((p - a) / (b - a), alpha, beta) / (b - a),
        0
      )
    )

  # Linear opinion pool
  dens_mixture <- dens_individual %>%
    group_by(p) %>%
    summarise(density = mean(density), .groups = "drop")

  # Moment-matched Beta on [0,1] (guard against tiny variance)
  m_hat <- mean(samples)
  v_hat <- var(samples)
  if (!is.finite(v_hat) || v_hat <= 1e-12) v_hat <- 1e-6
  ab_term <- max(m_hat * (1 - m_hat) / v_hat - 1, 2)
  alpha_star <- m_hat * ab_term
  beta_star <- (1 - m_hat) * ab_term
  grid_beta <- seq(0, 1, length.out = 1000)
  df_beta_fit <- data.frame(p = grid_beta, density = dbeta(grid_beta, alpha_star, beta_star))
}
```

```

# Summaries
summary_tbl <- tibble(
  Question      = if (is.null(question_label)) NA_character_ else question_label,
  Pooled_Mean   = m_hat,
  Pooled_Median = median(samples),
  Hard_Union_LPP = min(df2$a),
  Hard_Union_HPP = max(df2$b),
  Mixture_5th   = as.numeric(quantile(samples, 0.05)),
  Mixture_95th  = as.numeric(quantile(samples, 0.95)),
  N_Participants = nrow(df2),
  Lambda        = lambda,
  Nsim          = Nsim
)

# CDFs
cdf_individual <- df2 %>%
  select(id, a, b, alpha, beta) %>%
  crossing(p = grid) %>%
  mutate(
    cdf = case_when(
      p <= a ~ 0,
      p >= b ~ 1,
      TRUE ~ pbeta((p - a) / (b - a), alpha, beta)
    )
  )
cdf_mixture <- cdf_individual %>%
  group_by(p) %>%
  summarise(cdf = mean(cdf), .groups = "drop")
ec <- ecdf(samples)
cdf_emp <- data.frame(p = grid, cdf = ec(grid))
cdf_beta_fit <- data.frame(p = grid_beta, cdf = pbeta(grid_beta, alpha_star, beta_star))

# Facet-ready helper
add_q <- function(df_fac) {
  df_fac %>%
    mutate(Question = if (is.null(question_label)) NA_character_ else question_label) %>%
    relocate(Question, .before = 1)
}

list(
  summary      = summary_tbl,
  densities_individual = dens_individual,
  density_mixture = dens_mixture,
  samples      = samples,
  cdfs         = list(individual = cdf_individual,
                      mixture = cdf_mixture,
                      empirical = cdf_emp,
                      beta_fit = cdf_beta_fit),
  facet        = list(
    mixture      = add_q(dens_mixture),
    individual   = add_q(dens_individual),
    beta         = add_q(df_beta_fit),
    cdf_mixture  = add_q(cdf_mixture),
    cdf_emp      = add_q(cdf_emp),
    cdf_beta     = add_q(cdf_beta_fit),
    cdf_individual = add_q(cdf_individual)
  ),
  beta_fit_params = list(alpha = alpha_star, beta = beta_star)
)
}

```

DATA

Data (demo_data.R)

```
demo_df <- cbind.data.frame(
  Question = rep(seq(1:8), each=6),
  Participant = rep(c("P1", "P2", 'P3', 'P4', 'P5', 'P6'), 8),
  Lowest_Plausible_Pr = c(0.14,0.17,0.25,0.17,0.16,0.25,0.54,0.37,0.40,0.51,
    0.56,0.45,0.24,0.26,0.22,0.35,0.29,0.20,0.03,0.06,
    0.00,0.04,0.00,0.02,0.34,0.29,0.31,0.36,0.29,0.38,
    0.51,0.51,0.54,0.55,0.56,0.49,0.31,0.32,0.31,0.31,
    0.25,0.25,0.03,0.20,0.16,0.06,0.11,0.09),
  Best_Guess_Pr = c(0.23,0.28,0.30,0.23,0.26,0.32,0.64,0.43,0.44,0.62,0.65,
    0.49,0.36,0.35,0.35,0.42,0.38,0.26,0.18,0.12,0.11,0.12,
    0.08,0.16,0.44,0.36,0.47,0.43,0.45,0.53,0.61,0.60,0.63,
    0.63,0.62,0.62,0.46,0.40,0.39,0.39,0.40,0.38,0.13,0.32,
    0.24,0.14,0.16,0.16),
  Highest_Plausible_Pr = c(0.30,0.36,0.41,0.34,0.34,0.43,0.71,0.52,0.51,0.66,
    0.75,0.56,0.48,0.45,0.39,0.52,0.45,0.36,0.34,0.25,
    0.22,0.26,0.26,0.22,0.49,0.47,0.62,0.55,0.51,0.67,
    0.71,0.67,0.73,0.70,0.71,0.75,0.49,0.47,0.46,0.49,
    0.47,0.54,0.22,0.38,0.33,0.26,0.27,0.30))
```

APPLICATION

User interface (UI.R)

```
ui <- fluidPage(
  titlePanel("Expert Elicitation Application"),
  sidebarLayout(
    sidebarPanel(
      fileInput("file", "Upload CSV", accept = c(".csv")),
      checkboxInput("use_demo", "Use demo data instead of upload", value = TRUE),
      tags$hr(),
      uiOutput("colmap_ui"),
      numericInput("lambda", "PERT shape (lambda)", value = 5, min = 1, step = 1),
      numericInput("Nsim", "Mixture draws", value = 10000, min = 1000, step = 1000),
      uiOutput("question_filter_ui"), # multi-select
      checkboxInput("show_individual", "Show individual curves in overlays", TRUE),
      actionButton("run", "Run / Refresh", class = "btn-primary"),
      tags$hr(),
      downloadButton("download_summary", "Download Summary CSV"),
      tags$hr(),
      h4("Download plots"),
      downloadButton("download_participants_png", "Expert Scores (PNG)",
        downloadButton("download_density_png", "Expert Distributions (PNG)",
          downloadButton("download_hist_png", "Mixture Histograms (PNG)",
            downloadButton("download_cdf_png", "CDFs (PNG)",
              br(), br(),
              downloadButton("download_all_plots_zip", "All Summaries (ZIP)")
            ),
          ),
        ),
      mainPanel(
        tabsetPanel(
          tabPanel("Expert Scores", plotOutput("plot_participants", height = "600px")),
          tabPanel("Expert Distributions", plotOutput("plot_density", height = "600px")),
          tabPanel("Mixture Histograms", plotOutput("plot_hist", height = "600px")),
          tabPanel("CDF Comparisons", plotOutput("plot_cdf", height = "600px")),
          tabPanel("Summary Table", div(style = "font-size: 20px;",
            tableOutput("summary_table")),
            helpText("Summaries are for the equal-weight mixture of participant PERT
distributions (hard bounds with mode at BGP)."))
          )
        )
      )
    )
  )
)
```

Server (server.R)

```
# Server
server <- function(input, output, session) {
  # Load data (reactive) and clean names without emitting "New names" message
  raw_df <- reactive({
    if (isTRUE(input$use_demo) || is.null(input$file)) {
      demo_df
    } else {
      df <- suppressMessages(
        read_csv(input$file$datapath,
                 show_col_types = FALSE,
                 name_repair = "minimal")
      )
    }
  })

  # Column mapping UI
  output$colmap_ui <- renderUI({
    req(raw_df())
    df <- raw_df()
    cols <- names(df)
    default_question <- best_question_col(df)

    tagList(
      selectInput("col_question", "Question column", choices = c("Question", cols),
                  selected = if (default_question %in% cols) default_question else "<none>"),
      selectInput("col_id", "Participant ID column", choices = cols,
                  selected = if ("Participant" %in% cols) "Participant" else cols[1]),
      selectInput("col_lpp", "Lowest plausible (LPP)", choices = cols,
                  selected = grep("Lowest|LPP", cols, ignore.case = TRUE, value = TRUE)[1]),
      selectInput("col_bgp", "Best guess (BGP)", choices = cols,
                  selected = grep("Best|BGP", cols, ignore.case = TRUE, value = TRUE)[1]),
      selectInput("col_hpp", "Highest plausible (HPP)", choices = cols,
                  selected = grep("Highest|HPP", cols, ignore.case = TRUE, value = TRUE)[1])
    )
  })

  # Multi-question filter UI (select many or "All")
  output$question_filter_ui <- renderUI({
    req(raw_df(), input$col_question)
    if (identical(input$col_question, "<none>")) return(NULL)
    qs <- sort(unique(raw_df()[[input$col_question]]))
    selectInput("question_multi", "Questions to display",
                choices = c("All", qs), selected = "All", multiple = TRUE)
  })

  # Run/Refresh
  results <- eventReactive(input$run, {
    df <- raw_df()
    req(input$col_id, input$col_lpp, input$col_bgp, input$col_hpp)

    # Attach a question column if none provided
    has_q <- !identical(input$col_question, "<none>")
    if (!has_q) {
      df <- df %>% mutate(`__Question__` = "Q1")
    }
    q_col <- if (has_q) input$col_question else "__Question__"

    # If one or more questions chosen (not "All"), filter to those
```

```

if (has_q && !is.null(input$question_multi)) {
  sel <- setdiff(input$question_multi, "All")
  if (length(sel) > 0) {
    df <- df %>% filter(.data[[q_col]] %in% sel)
  }
}

qs <- unique(df[[q_col]])

# Summarize per selected questions (random seed each run)
res_list <- lapply(qs, function(q) {
  df_q <- df %>% filter(.data[[q_col]] == q)
  summarize_question_pert(
    df_q,
    id_col = input$col_id,
    lpp_col = input$col_lpp,
    bgp_col = input$col_bgp,
    hpp_col = input$col_hpp,
    lambda = input$lambda,
    Nsim = input$Nsim,
    seed = NULL, # random seed inside
    question_label = as.character(q)
  )
})
names(res_list) <- as.character(qs)

# Bind everything
summary_all <- bind_rows(lapply(res_list, `[`, "summary"))
dens_mix_all <- bind_rows(lapply(res_list, function(r) r$facet$mixture))
dens_ind_all <- bind_rows(lapply(res_list, function(r) r$facet$individual))
beta_all <- bind_rows(lapply(res_list, function(r) r$facet$beta))

cdf_mix_all <- bind_rows(lapply(res_list, function(r) r$facet$cdf_mixture))
cdf_emp_all <- bind_rows(lapply(res_list, function(r) r$facet$cdf_emp))
cdf_beta_all <- bind_rows(lapply(res_list, function(r) r$facet$cdf_beta))
cdf_ind_all <- bind_rows(lapply(res_list, function(r) r$facet$cdf_individual))

samples_all <- bind_rows(lapply(names(res_list), function(nm) {
  tibble(Question = nm, samples = res_list[[nm]]$samples)
}))

list(
  summaries = summary_all,
  dens_mix_all = dens_mix_all,
  dens_ind_all = dens_ind_all,
  beta_all = beta_all,
  cdf_mix_all = cdf_mix_all,
  cdf_emp_all = cdf_emp_all,
  cdf_beta_all = cdf_beta_all,
  cdf_ind_all = cdf_ind_all,
  samples_all = samples_all
)
}, ignoreInit = TRUE)

# Single PNG downloads (use current selections in results())
output$download_participants_png <- downloadHandler(
  filename = function() paste0("participants_raw_", Sys.Date(), ".png"),
  content = function(file) {
    req(results())
    df0 <- raw_df()
    req(input$col_id, input$col_lpp, input$col_bgp, input$col_hpp)
  }
)

```

```

has_q <- !identical(input$col_question, "<none>")
if (!has_q) {
  q_col <- "__Question__"
  df0 <- df0 %>% mutate(`__Question__` = "Q1")
} else {
  q_col <- input$col_question
}
if (has_q && !is.null(input$question_multi)) {
  sel <- setdiff(input$question_multi, "All")
  if (length(sel) > 0) {
    df0 <- df0 %>% filter(.data[[q_col]] %in% sel)
  }
}

dfp <- df0 %>%
  transmute(
    Question = .data[[q_col]],
    Participant = as.character(.data[[input$col_id]]),
    Lowest_Plausible_Pr = suppressWarnings(as.numeric(.data[[input$col_lpp]])),
    Best_Guess_Pr = suppressWarnings(as.numeric(.data[[input$col_bgp]])),
    Highest_Plausible_Pr = suppressWarnings(as.numeric(.data[[input$col_hpp]]))
  )

facet_cols <- if (!is.null(input$facet_cols)) input$facet_cols else 4

g <- ggplot() +
  coord_flip()+
  geom_linerange(data = dfp, aes(x = Participant, ymin = Lowest_Plausible_Pr,
                                ymax = Highest_Plausible_Pr), lwd = 0.5) +
  geom_point(data = dfp, aes(x = Participant, y = Best_Guess_Pr), size = 1) +
  labs(x = "Participant", y = "Probability") + ylim(0,1) +
  facet_wrap(~Question, ncol = facet_cols, scales = "fixed") + theme_export +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank())

ggsave(file, g, width = 8, height = 4, dpi = 1200, units='in', limitsize = FALSE)
}
)

output$download_density_png <- downloadHandler(
  filename = function() paste0("density_", Sys.Date(), ".png"),
  content = function(file) {
    req(results())
    r <- results()
    facet_cols <- if (!is.null(input$facet_cols)) input$facet_cols else 4
    g <- build_density_plot(r,
                          show_individual = isTRUE(input$show_individual),
                          facet_cols = facet_cols)
    g_small <- g + theme_export # <-- apply smaller export theme
    ggsave(file, g_small, width = 8, height = 4, dpi = 1200, units='in', limitsize = FALSE)
  }
)

output$download_hist_png <- downloadHandler(
  filename = function() paste0("histogram_", Sys.Date(), ".png"),
  content = function(file) {
    req(results())
    r <- results()
    facet_cols <- if (!is.null(input$facet_cols)) input$facet_cols else 4
    g <- build_hist_plot(r, facet_cols = facet_cols)
    g_small <- g + theme_export # smaller export theme
  }
)

```

```

    ggsave(file, g_small, width = 8, height = 4, dpi = 1200, units='in', limitsize = FALSE)
  }
)

output$download_cdf_png <- downloadHandler(
  filename = function() paste0("cdf_", Sys.Date(), ".png"),
  content = function(file) {
    req(results())
    r <- results()
    facet_cols <- if (!is.null(input$facet_cols)) input$facet_cols else 4
    g <- build_cdf_plot(r, show_individual = isTRUE(input$show_individual),
                       facet_cols = facet_cols)
    g_small <- g + theme_export # smaller export theme
    ggsave(file, g_small, width = 8, height = 4, dpi = 1200, units='in', limitsize = FALSE)
  }
)

# ZIP files
output$download_all_plots_zip <- downloadHandler(
  filename = function() paste0("elicitation_plots_", Sys.Date(), ".zip"),
  content = function(file) {
    req(results())
    r <- results()

    # facet columns
    facet_cols <- if (!is.null(input$facet_cols)) input$facet_cols else 4
    show_ind <- isTRUE(input$show_individual)

    # Build plots
    g1 <- build_density_plot(r, show_individual = show_ind, facet_cols = facet_cols)
    g2 <- build_hist_plot(r, facet_cols = facet_cols)
    g3 <- build_cdf_plot(r, show_individual = show_ind, facet_cols = facet_cols)
    g4 <- build_individuals_plot(
      df_raw = raw_df(),
      id_col = input$col_id,
      lpp_col = input$col_lpp,
      bgp_col = input$col_bgp,
      hpp_col = input$col_hpp,
      question_col = input$col_question,
      selected_questions = input$question_multi,
      facet_cols = facet_cols,
      use_export_theme = TRUE, # set FALSE to use app theme
      theme_export = if (exists("theme_export")) theme_export else NULL
    )

    # Save all figures to a temp directory
    outdir <- tempfile("plots_")
    dir.create(outdir, showWarnings = FALSE)

    f1 <- file.path(outdir, paste0("density_", Sys.Date(), ".png"))
    f2 <- file.path(outdir, paste0("histogram_", Sys.Date(), ".png"))
    f3 <- file.path(outdir, paste0("cdf_", Sys.Date(), ".png"))
    f4 <- file.path(outdir, paste0("individuals_", Sys.Date(), ".png"))

    ggsave(f1, g1, width = 8, height = 4, dpi = 1200, units = 'in', limitsize = FALSE)
    ggsave(f2, g2, width = 8, height = 4, dpi = 1200, units = 'in', limitsize = FALSE)
    ggsave(f3, g3, width = 8, height = 4, dpi = 1200, units = 'in', limitsize = FALSE)
    ggsave(f4, g4, width = 8, height = 4, dpi = 1200, units = 'in', limitsize = FALSE)

    # summary CSV
    f_summary_csv <- file.path(outdir, paste0("summary_", Sys.Date(), ".csv"))

```

```

write_csv(r$summaries, f_summary_csv)

# Zip
oldwd <- setwd(outdir); on.exit(setwd(oldwd), add = TRUE)
zip(zipfile = file, files = basename(c(f1, f2, f3, f4, f_summary_csv)))
}
)

output$plot_participants <- renderPlot({
  req(results())

  build_individuals_plot(
    df_raw = raw_df(),
    id_col = input$col_id,
    lpp_col = input$col_lpp,
    bgp_col = input$col_bgp,
    hpp_col = input$col_hpp,
    question_col = input$col_question,
    selected_questions = input$question_multi,
    facet_cols = NULL
  )
})

# Mixture density
output$plot_density <- renderPlot({
  req(results())

  build_density_plot(
    r = results(),
    show_individual = isTRUE(input$show_individual),
    facet_cols = NULL
  )
})

# Histogram + Beta fit
output$plot_hist <- renderPlot({
  req(results())

  build_hist_plot(
    r = results(),
    facet_cols = NULL
  )
})

# CDF comparison
output$plot_cdf <- renderPlot({
  req(results())

  build_cdf_plot(
    r = results(),
    show_individual = isTRUE(input$show_individual),
    facet_cols = NULL
  )
})

# Summary table & download
output$summary_table <- renderTable({
  req(results())
  results()$summaries
})

```

```
output$download_summary <- downloadHandler(  
  filename = function() paste0("elicitation_summaries_", Sys.time(), ".csv"),  
  content = function(file) {  
    req(results())  
    readr::write_csv(results()$summaries, file)  
  }  
)  
}  
  
# Run app  
shinyApp(ui, server)
```

Application (app.R)

```
source("Functions/themes.R")
source("Functions/helper.R")
source("Functions/summarize_pert.R")
source("Functions/plotting_functions.R")
source("Data/demo_data.R")
source("App/UI.R")
source("App/server.R")

shinyApp(ui, server)
```