Interpretable Aggregation of Correlated LLM Annotators*

Adam Bouyamourn
Center for Data Science
New York University
azb236@nyu.edu

Arthur Spirling
Department of Politics
Princeton University
arthur.spirling@princeton.edu

Abstract

We study the problem of using multiple large language models (LLMs) to annotate data, or to generate predictions. We first show that aggregating LLM outputs requires that the researcher make at least one of three identification assumptions: first, that the analyst has access to ground truth labels; second, that LLMs are highly accurate and unbiased for the specific task they are being used for; third, that the researcher can estimate ground truth labels using item features extracted from the documents themselves. We then introduce a dependence-aware aggregation approach, the correlated Dawid-Skene crowdsourcing model, in which the correlation matrix of raters is explicitly estimated, in addition to rater accuracies. This allows us to estimate predicted probabilities, get valid confidence intervals, and flag difficult items, while taking into account the estimated dependence between raters. Finally, we use sparse autoencoders to provide interpretable outputs for the correlated Dawid-Skene model: this layer explains the predicted probabilities, the biases of specific LLMs, and the items that were difficult for the LLMs to classify, in natural language using both LLM summaries and sampled sentences from the documents themselves. The result is a fully-automated pipeline that provides both quantitative outputs that take into account the correlation between LLM raters, and provides plain language, qualitative insights that are useful for applied researchers in a substantive domain. We study the performance of this method via simulation, and apply it to a manifesto classification task.

1 Introduction

Political scientists have been at the forefront of using large language models (LLMs) to code and annotate data for downstream use. These models are fast, accurate and can outperform experts for many tasks [e.g. 7, 21, 10, 22]. The typical setup is to use one particular model, e.g. ChatGPT, and compare its annotation decisions to a pre-existing "gold standard". This allows the researcher to make comments about performance, in terms of accuracy, precision, recall and so on. As the number of cheap, high performing LLMs has grown, authors have worked on ways to draw inferences from multiple models in their analysis. One possibility is to use averaging, which includes the use of LLMs as ensembles [20, 1]. The logic is via appeal to the "wisdom of crowds" insofar as we believe that aggregation essentially cancels out errors, and returns more stable estimates.

An immediate problem is that we cannot distinguish between two cases: one where the LLMs are accurate vs one where the LLMs are dependent. Put starkly: the pessimistic scenario with an ensemble of LLMs is that they all agree, but are all incorrect for the same reasons. In this case, one cannot take even a very lop-sided majority decision as truth: the average of biased models is biased, possibly worse than any single one. This is a concern in that we know from prior work that LLM

*First version July 10, 2026. This version: September 20, 2026.

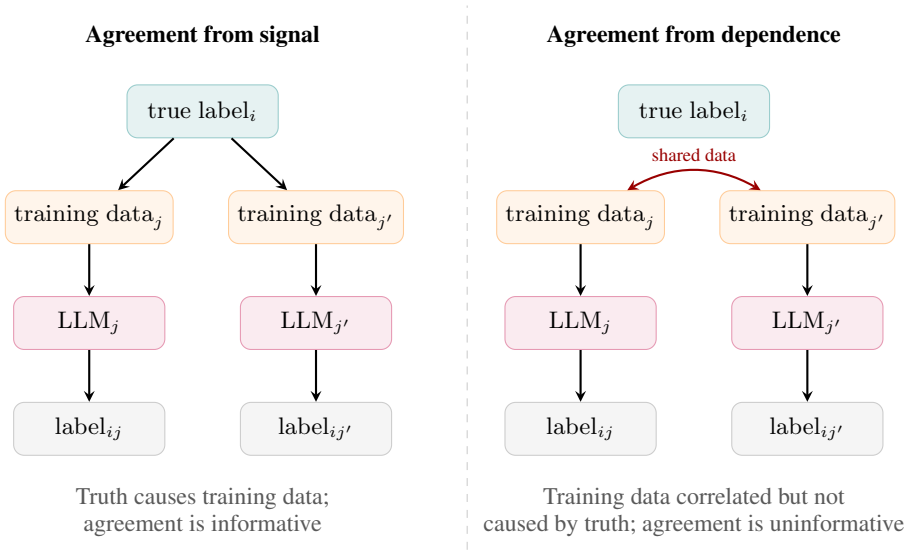


Figure 1: The two situations above are observationally equivalent without further assumptions.

decisions are typically highly correlated for political science tasks [e.g. 12, 23]. An obvious route for such dependence to occur is the fact that different models are trained on the same information. These include sources such as CommonCrawl, an open repository of scraped web data. As an empirical matter, we have strong reason to suspect that LLMs have correlated errors [11].

Issues of dependence are compounded by the fact that LLMs can exhibit bias (potentially 'deliberately wrong' labels) for some tasks [e.g. 18]. There are methods to correct such outputs back to human expert standards for subsequent analysis [6, 9]. But this requires some external 'gold standard' benchmark against which to compare the LLM output. For many problems, this may be unavailable, perhaps because humans are not the gold standard or because the previously assembled expert benchmark does not align well with the use-case in question.

This dependence and bias for LLMs violates the usual assumptions we are willing to make about *human* raters—including crowd workers. There it is conceivable that each rater produces a (conditionally) independent annotation for a given item. That is, errors across raters are uncorrelated. If one then assumes that each worker is unbiased, on average (i.e. raters have mean zero errors relative to the truth), then a majority vote or mean of the worker decisions will be correct at the corpus level.[2]

Using approaches such as Dawid and Skene [5], researchers can rely on these arguably mild assumptions to produce both measures of rater quality (good to bad) and estimates of item difficulty (easy to hard). Note that one does not need an external benchmark or agreed gold standard to do this. Such an approach is of obvious utility to researchers using LLMs: we might want to reweight aggregates according to differential model performance and use disagreement on items when including them in downstream analysis.

To do this, we propose a modification to the Dawid-Skene model that estimates the correlation structure between LLM annotators directly from the observed votes. The method requires no gold standard labels and no external benchmark. It takes as input the same vote matrix used by majority vote, and returns classification probabilities, confidence intervals, and flags for follow-up that account for both annotator accuracy and annotator dependence.

To motivate that modification, we first clarify in some detail the difference between ensembles of humans and ensembles of LLMs, and why the lessons from the former do not generally carry through to the latter. This requires that we explicate the nature of LLM dependence and how it affects aggregations. From here, we explain under what circumstances a Dawid-Skene style approach could work, albeit we do not think they are met in practice for most use cases. Second, we report estimators that do allow us to recover correlation structure and accuracies for LLMs in a Dawid-Skene

[2]The Condorcet result where each worker has a better than chance probability of making a correct decision deals with a special case here.

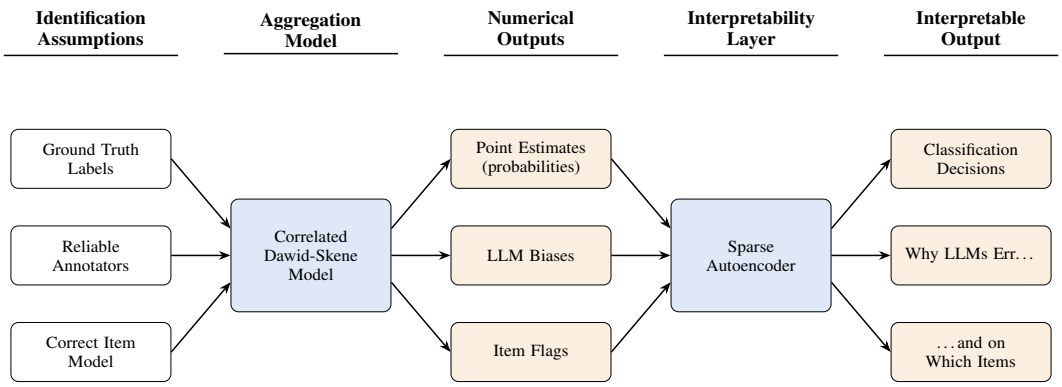


Figure 2: **Structure of the paper.** Any one of three identification assumptions (left) is sufficient to identify the correlated Dawid-Skene model. The model returns three numerical outputs: point estimates of the class probabilities, per-rater biases, and item flags. A sparse autoencoder maps these outputs to features of the document text, producing interpretable classification decisions, explanations of LLM biases, and reasons why items were flagged for analyst follow-up.

arrangement with correlation terms. These are not trivial to identify, specifically in the case without a gold standard.

These allow for helpful quantities to be estimated that are valuable in applied research settings. Specifically, we obtain class label probabilities and uncertainty estimates that account for estimated dependence between LLMs, as well as their relative accuracies. This allows for sensible and systematic inclusion of ensemble-produced annotations in downstream estimates.

This provides two things that majority vote does not. First, classification probabilities that reflect which LLMs voted for which label, how accurate each LLM is estimated to be, and how correlated they are with one another: a 3–2 vote from five correlated LLMs is treated as weaker evidence than a 3–2 vote from five independent annotators. Second, confidence intervals that widen when annotator dependence is high, rather than assuming each vote is an independent signal.

By studying the bias of the estimator formally and via simulation, we show that the approach is an improvement over majority vote when LLMs are only approximately correct. This is because for our method to fail, it is necessary that the average prediction is uncertain, LLMs are inaccurate, and are highly correlated. If any of these conditions does not obtain, the bias is small in practice.

Proofs of all results appear in Appendix A.

2 Problem Setup: LLM aggregation as an identification problem

Suppose we observe the annotations of K LLMs on N items. We observe a vector of LLM annotations y_{ij}, which is the proposed label of the j^{th} LLM for the i^{th} item. Each item has a *true* but *unobserved* label c_i associated with it. We collect the votes in a vector $\boldsymbol{y}$ and the true class labels in the vector $\boldsymbol{c}$.

When aggregating LLMs, we observe a statistic of the votes, $T(\boldsymbol{y})$, and our question is, when is it the case that $T(\boldsymbol{y})$ is a consistent and unbiased estimator of $\boldsymbol{c}$?

In the simplest case, we can take $T(\boldsymbol{y}) = \left(\sum_{j=1}^{K} \frac{1}{K}\mathbb{I}\{y_{ij} = 1\}\right)_{i=1}^{N}$, that is, the simple average of the votes for each item. The question then becomes, under what assumptions is a simple average of the votes a valid estimator of the ground truth labels $\boldsymbol{c}$?

This is an *identification problem*, in the sense that we want to know, when does our observed statistic converge to the correct unknown quantity $\boldsymbol{c}$? Identification is a core problem in causal inference: we want to know when we can use observed data to draw unbiased and consistent estimates of treatment effects, and we do so in an environment in which ground truth labels are only partially observed.

We study three cases in which ground truth labels are identified from observed data. These cases highlight that the analyst must make assumptions when using LLMs to annotate data or make

predictions; and that the analyst must choose and defend a specific assumption in order to justify valid inference using an ensemble of LLMs.

These assumptions are non-overlapping, and the analyst must choose at least one.

2.1 Access to ground truth labels

The first assumption is that the analyst has **access to ground truth labels**. Here, the analyst supposes, simply, that ground truth labels are observed, or partially observed. From this, a proxy $\hat{c}_i$, for $i \leq N$, can be formed, and this can be used as plug-in estimate for downstream analysis. This is the approach used in [6, 8, 3].

It is analogous to running a randomized experiment, in which potential outcomes under treatment and control are partially observed.

Assumption 1 (Ground truth labels)**.** *For a subset $\mathcal{S} \subseteq \{1, \ldots, I\}$, the true labels c_i are observed. The subset is selected independently of the true labels and the votes.*

Under this assumption, the analyst can simply observe $\hat{c} = c$ for some subset of the labels, and plug this in to downstream inference, as we describe in more detail in Section 3.

2.2 Assumptions on the LLMs themselves

The second assumption is that the LLMs are, themselves, collectively good enough at the task that they have been set. We make this precise in two ways.

Assumption 2 (Reliable raters)**.** *One of the following holds.*

(a) Oracle. *The leave-one-out majority vote equals the true label for every item: $\hat{c}_{i,-j} = c_i$ for all i and j.*

(b) Conditional independence.

(i) The votes are conditionally independent given the true label: $y_{ij} \perp\!\!\!\perp y_{ij'} \mid c_i$ for all $j \neq j'$.

(ii) For every rater j, the average accuracy of the other raters exceeds chance:

$$\frac{1}{K-1}\sum_{k\neq j}\alpha_k > \tfrac{1}{2} \qquad \text{and} \qquad \frac{1}{K-1}\sum_{k\neq j}\gamma_k > \tfrac{1}{2}.$$

Conditional Independence First, if the LLMs are, in fact, conditionally independent, and are more than accurate by chance, then we can use Condorcet's jury theorem to show that then aggregating LLM decisions is innocuous. Let α_j be the accuracy of LLM j, that is, the probability that its label is correct for a given item. Then, if $\alpha_j > 1/2$, we have:

$$\begin{aligned} P(\text{Majority}\{y_j\} \neq 1 \mid c_i = 1) &= \sum_{m=0}^{(K-1)/2} \binom{K}{m} \alpha^m (1-\alpha)^{K-m} \\ &= P\left(\frac{1}{K}\sum_{j=1}^{K} y_j \leq \tfrac{1}{2}\right) \\ &\leq \exp\Big(-2K\left(\alpha - \tfrac{1}{2}\right)^2\Big) \qquad \text{[By Hoeffding's inequality]} \\ &\searrow 0 \qquad \text{as } K \to \infty. \end{aligned}$$

This is the motivation for the Dawid-Skene model of annotation [5], in which the errors of conditionally independent annotators are washed out by recruiting larger numbers of annotators.

In general, however, this assumption is difficult to believe for LLMs, because LLMs are trained on overlapping training data. It is also impossible for a third-party analyst to know exactly what data the LLM is trained on, meaning the *dependence* of LLMs is a necessary assumption.

Reliable Raters In place of assuming that the LLMs are independent, it is possible instead that they are dependent because their votes are *caused* by the ground truth labels. Formally, in the binary case, we require that every leave-one-out-majority vote of the LLMs is correct, and in the ordinal case, we require that the leave-one-out-median vote of the LLMs is correct.

This can appear to be a strong assumption. But there are two points to note. First, LLMs may well be reliable raters for tasks that are repeatable and in-distribution. Tasks that are well-specified and do not require specific factual information that is unlikely to be found in the training data are tasks in which the assumptions may hold. In the empirical section below, we study the manifesto classification task of Benoit et al. [1], for which LLMs have approximately 90% agreement with expert labels out of the box – in a situation where expert labels themselves only have 90% agreement with each other. So the fact that LLM performance is indistinguishable from expert label performance on average suggests that LLMs are reasonably good at the task.

Second, in practice, the bias from treating LLMs as though they are reliable may be small. The bias is large when three conditions hold simultaneously: i) when LLMs are correlated even conditional on the ground truth votes ii) when there is high disagreement about the class label for a given item and iii) when LLMs are inaccurate. We describe this result formally in Appendix A.3. This can be understood as analogous to a sensitivity analysis argument in causal inference [4]. Note, however, that the analyst cannot actually observe conditions i) and iii) without further assumptions. In the next section we provide model-based methods for estimating the conditional correlation and LLM accuracies. But it is circular to use these estimates to argue that the bias from treating LLMs as though they are reliable is small: this is something that must be believed ex ante, and the consequences of being wrong about LLM reliability can be *simulated* ex post, but not measured per se.

2.3 Access to a valid model of the items

The third possible identification approach is that the analyst has a **valid model of the items**. The idea here is that the analyst can generate indicators of the ground truth based on the *document contents* themselves. This requires positing a relationship between the semantic content of documents, and the classification decisions. This can be done using either a coarse analyst-driven scoring model, or an unsupervised learning method.

While an LLM presumably makes classification decision on the document contents, because it is a black-box classifier, we cannot be entirely sure *why* it made the classification decisions that it did. The idea here is to cross-check LLM decisions against a coarse but analyst-implemented model of classification decisions. When the analyst model is good, this is a source of identifying information we can leverage.

Feature extraction for the item model First, we need to be able to extract *item-level covariates* (x_i) from the documents to be annotated. These can be thought of as conceptual features or semantic properties of the documents themselves. For instance, if we are classifying manifestos as left-wing or right-wing on the economic dimension, we might think of this as indicators of the type "refers to public ownership" or "refers to worker's rights".

In order to extract features, we suppose that the user specifies a feature-extraction procedure, or 'recipe', that generates features x_i from a document D_i. We write $R(D_i)$ for the output of the feature extraction recipe.

In practice, there are several recipes a researcher could use. The simplest would be a user-specified dictionary, with regex and string-matching. A researcher could also use an LLM to extract features from documents.

This requires an assumption that the LLM-extracted features do not inherit the same dependence issues they have in the classification case. A subtle point is that using LLMs for feature *extraction* may well be more robust than using LLMs for *interpretation decisions*. In general, the more well-specified the task given to the LLM, the smaller the possibility of 'agency loss'.

The required assumption is that the features carry no information about the votes beyond the true label. We can decompose this claim into two claims. First, the LLM must execute the recipe correctly. The concepts the recipe targets must themselves be unrelated to LLM biases.

Assumption 3 (Mechanical feature extraction)**.**

$$x_i \perp\!\!\!\perp (D_i, \mathbf{y}_i) \mid R(D_i),$$

where $\mathbf{y}_i = (y_{i1}, \ldots, y_{iK})$ *is the vector of votes on item* i*.*

The extractor implements the recipe. Its output depends on the document through the recipe alone, and its errors are independent of the errors of the LLMs.

The first component is not very onerous in practice, for the reason that it is easy to force an LLM to implement a deterministic and unambiguous recipe. Suppose the recipe is fully specified, such as "extract all noun phrases from the document". Then, there is much less scope for LLM 'discretion' in selecting items – we are essentially using the LLM as a natural language interface to regex. As a result, we care about LLM dependence inducing erroneous results on the *subtask* of feature extraction with a fully-specified recipe, rather than on the more complex task of document classification. It is also much easier to check whether the LLM completed its feature extraction decisions correctly or not: the correct output of the recipe is checkable by reading the document, so failures of execution can be detected by inspection.

The second assumption is the requirement that the recipe not extract terms that are related to LLM biases, but not to the ground truth labels.

Assumption 4 (Semantic exclusion restriction)**.**

$$R(D_i) \perp\!\!\!\perp \mathbf{y}_i \mid c_i.$$

The second component is more demanding, and it is *distinct* from the method by which features were extracted. In practice, it means that the item model *must be constructively built up by the researcher to include concepts that they know are relevant to the classification task, but not a cause of LLM biases.* For instance, if using Chinese-origin LLMs to classify documents that refer to China, it would not be possible to use features that refer to Taiwan. Hence, it is up to the researcher to provide a working model of ground truth labels. This is essentially a coarse, lower-dimensional approximation of the conceptual material that explains the classification decision, and is the part of the process where substantive knowledge can be used as an input into the decision-making.

The true label is a property of the document, so c_i is a function of D_i. Together, these assumptions entail the conditional independence of the features from the votes given the true label:

Lemma 1 (Validity of extracted features)**.** *Under Assumptions 3 and 4,*

$$x_i \perp\!\!\!\perp \mathbf{y}_i \mid c_i.$$

The proof is stated in Appendix A.1.

Using an item model to estimate ground truth labels The item model is an estimate of the probability of the label given the features. Write

$$e(x) = P(c_i = 1 \mid x_i = x).$$

The item model is an estimate $\hat{e}$ of this function, and the *score* for item i is $\hat{e}(x_i)$.[3]

Under the hypothesis that features are semantically related to the classification decision, features should occur at different rates in documents that belong to different classes.

So, for instance, this analysis requires that a left-wing manifesto will refer to public ownership and worker's rights at a different rate to right-wing manifestos. In general, specifying terms that increase the contrast between items is helpful.

Appendix B describes a two-class mixture model that can be used to fit the score. We state the requirement on the fitted score as an assumption.

Assumption 5 (Item-model consistency)**.** *The fitted score takes values in* $[0, 1]$*, and its average error converges to zero:*

$$\frac{1}{I} \sum_{i=1}^{I} \left| \hat{e}(x_i) - e(x_i) \right| \xrightarrow{p} 0 \qquad \text{as } I \to \infty.$$

[3]Note the direct analogy with propensity score modeling in causal inference [19].

The mean of the score equals the prevalence, $\pi = P(c_i = 1)$. Because $e(x_i) = P(c_i = 1 \mid x_i)$ by definition, the law of iterated expectations gives $\mathbb{E}[e(x_i)] = \pi$.

The score varies across items when the features are informative about the class.

Assumption 6 (Relevance)**.** $\mathrm{Var}(e(x_i)) > 0$.

Assumption 6 holds when the features occur at different rates in the two classes. It fails when the features occur at the same rate in both, in which case $e(x_i)$ equals the prevalence π at every item. This is a fairly minimal requirement on features.

The item model orders items by their probability of belonging to the positive class. Section 3.2 uses this ordering to identify the accuracies of the LLMs, from the rate at which each LLM's votes increase with the score.[4]

In general, the quality of the item model is determined by the quality of the features and the model, both of which are chosen by the analyst.

Fitting the item model Given a consistent score $\hat{e}(x)$, we can use the fact that the votes are linear in the score to derive the accuracies. Hence, the second step is to regress the votes on the score, and derive LLM accuracies from this. This is described in Appendix A.2.

3 The Correlated Dawid–Skene Model

The above arguments describe assumptions under which ground truth labels are recoverable. We are now interested in describing statistical machinery that allows one to get quantities of interest from aggregations of LLMs. These are quantities that are directly of interest, and which are useful for downstream analysis, where they may be plugged into causal inference analyses [14]. Second, these estimates take into account *estimated dependence between raters*. In general, simple averages or majority votes of LLMs may overstate our certainty in a given conclusion, because they do not take into account how correlated LLMs are. These quantities yield *uncertainty estimates*, which is vitally important for inference – we want labels, but we also want confidence intervals that can be used in downstream analysis, based on both which LLM voted for which decision, *and* information we have about how correlated the LLMs are. We also get diagnostic information.

We extend the Dawid-Skene model in a way that is compatible with aggregations of LLMs, as opposed to aggregations of independent human raters. First, we describe the original model in words. The general problem is one of noisy, imperfect raters placing items in bins. For now, we can assume those are binary choices: in the political science context, this might be crowdworkers (or experts) placing manifesto sentences in either the "liberal" or "conservative" category. As with real human workers, the raters are not homogenous: some are better than others, and thus are more likely to recover the truth on average. In principle, for any given item, the researcher could take, say, a vote across the decisions by each rater and treat that as the overall classification. But this ignores the heterogeneity noted above, and treats high quality workers as having the same weight as low quality ones.

The key modification in our proposal is that the Dawid-Skene model assumes that raters are conditionally independent, conditioning on the true label c_i. That is, for some given item which is in fact "liberal" (or "conservative"), knowing one rater's decision does not help one predict another rater's decision. They may well agree (or disagree), but the true label is assumed to be the only common determinant of their actions. This, for example, rules out cases where one rater learns directly from another, or where both learn from some third source of information outside the problem. We do not make this assumption: we instead modify this model to consider cases where raters have a dependence structure that can be estimated from the data.

Accuracy parameters play a key role in the Dawid-Skene framework, and the approach in this paper is to *identify* the accuracy parameters using our estimates of ground truth labels. Given the estimate of ground truth labels, we can then estimate the *accuracies* of each of the LLM raters. Then, once we know how accurate each LLM is, we can estimate what part of the LLMs' voting decisions are explained by how accurate they are, and what part is explained by common dependence. In

[4] A score defined by hand, such as a normalized count of left-coded features, orders items without placing them on the probability scale. Section 3.2 requires the score to be the conditional probability $e(x_i)$.

other words, grounding the decision-making process in estimated ground truth labels allows us to decompose agreement into 'agreement caused by the ground truth label' and 'erroneous agreement caused by common training data'. In the second case, we can understand this as a type of hallucination [2].

3.1 Dawid-Skene with Estimated Correlation Matrix

We observe I items each labelled by K binary annotators, producing a vote matrix $\mathbf{Y} \in \{0,1\}^{I \times K}$. Each item has an unobserved true label drawn independently with a common base rate:

$$c_i \sim \text{Bernoulli}(\pi), \quad i = 1, \ldots, I.$$

Each annotator j has a sensitivity $\alpha_j = P(y_j = 1 \mid c = 1)$, the probability of correctly labelling a positive item, and a specificity $\gamma_j = P(y_j = 0 \mid c = 0)$, the probability of correctly labelling a negative item.

We model the dependence between annotations using a multivariate probit. For each item i, draw a vector of latent utilities:

$$\mathbf{z}_i \mid c_i = d \;\sim\; \mathcal{N}(\boldsymbol{\mu}_d,\; R), \qquad y_{ij} = \mathbf{1}[z_{ij} > 0].$$

Each annotator's binary vote is determined by whether their latent utility exceeds zero. The mean vectors are determined by the accuracy parameters:

$$\mu_{1,j} = \Phi^{-1}(\alpha_j), \qquad \mu_{0,j} = \Phi^{-1}(1 - \gamma_j),$$

where Φ^{-1} is the standard normal quantile function.

Our contribution is to allow annotators to be correlated in their errors. We suppose that annotators have correlation matrix R:

$$R = \begin{pmatrix} 1 & \rho_{12} & \cdots & \rho_{1K} \\ \rho_{12} & 1 & \cdots & \rho_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{1K} & \rho_{2K} & \cdots & 1 \end{pmatrix},$$

where each $\rho_{jj'} \in (-1, 1)$ is a free parameter.

An important point is that each $\rho_{jj'}$ represents the *conditional correlation* of the latent utilities $Corr(\boldsymbol{z}_j, \boldsymbol{z}_{j'} | \boldsymbol{c})$. That is, the correlation between the annotations of raters j and j' *after correlation that is due to agreement on correct annotations* has been taken into account. This is illustrated in Figure 1.

The latent utilities $\mathbf{z}_i$ are not observed. We observe only the votes $\mathbf{y}_i$. To obtain the probability of an observed vote pattern $\mathbf{y}_i$ given the true label, we integrate the multivariate normal density over the region of $\mathbf{z}_i$ consistent with $\mathbf{y}_i$:

$$\begin{aligned} P(\mathbf{y}_i \mid c_i = d, \boldsymbol{\alpha}, \boldsymbol{\gamma}, R) &= P(z_1 \in A_1,\; z_2 \in A_2,\; \ldots,\; z_K \in A_K) \\ &= \int_{A_1} \int_{A_2} \cdots \int_{A_K} \phi_K(\mathbf{z};\; \boldsymbol{\mu}_d, R)\, dz_1\, dz_2 \cdots dz_K \end{aligned}$$

where $P(z_1 \in A_1,\; z_2 \in A_2,\; \ldots,\; z_K \in A_K)$ is the joint probability that each annotator's latent utility has the sign prescribed by their vote, $\phi_K(\mathbf{z};\; \boldsymbol{\mu}_d, R)$ is the K-dimensional normal density with mean $\boldsymbol{\mu}_d$ and correlation matrix R, where $A_j > 0$ if $y_{ij} = 1$, and $A_j < 0$ otherwise.

Note that, unlike in Dawid-Skene, the joint probability does *not* factorize, because we do not have conditional independence of raters. Specifically

$$P(z_1 \in A_1,\; z_2 \in A_2,\; \ldots,\; z_K \in A_K) \neq \prod_{j=1}^{K} P(y_{ij} \mid c_i = d) = \prod_{j=1}^{K} \alpha_j^{y_{ij}} (1 - \alpha_j)^{1 - y_{ij}}$$

The right-hand side is the likelihood under conditional independence, in which the votes factorize given the label. Our model does not impose this factorization.

Summing over the unobserved true labels gives the marginal probability of a vote pattern:

$$P(\mathbf{y}_i \mid \pi, \alpha, \gamma, R) = \pi \cdot P(\mathbf{y}_i \mid c_i = 1, \alpha, R) + (1 - \pi) \cdot P(\mathbf{y}_i \mid c_i = 0, \gamma, R). \tag{1}$$

And, given the model parameters, the classification of each item follows from an application of Bayes' Rule:

$$P(c_i = 1 \mid \mathbf{y}_i, \alpha_j, \gamma_j, R) = \frac{\pi \cdot P(\mathbf{y}_i \mid c = 1,\, \alpha,\, R)}{\pi \cdot P(\mathbf{y}_i \mid c = 1,\, \alpha,\, R) \;+\; (1-\pi) \cdot P(\mathbf{y}_i \mid c = 0,\, \gamma,\, R)}$$

In other words, the probability of a label depends on which LLMs vote for it ($\boldsymbol{y_i}$), how accurate those LLMs are ($\boldsymbol{\alpha}, \boldsymbol{\gamma}$), and how correlated those LLMs are (R). We describe the correlated Dawid-Skene model for the binary case. We generalize it to ordinal data in Appendix C, which is the form used in the application in Section 5.

3.2 Estimating parameters

Accuracy terms In the above setting, we have three parameters we need to consistently estimate: $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$, and R. We only have access to the observed votes $\boldsymbol{Y}$, and do not observe the true labels c_i.

The sensitivity of annotator j is the conditional expectation:

$$\alpha_j = E[y_{ij} \mid c_i = 1].$$

By the definition of conditional expectation, this can be written as a ratio of unconditional expectations:

$$\alpha_j = \frac{E[c_i \cdot y_{ij}]}{E[c_i]}.$$

If we observed c_i, we could estimate this directly by its sample analogue:

$$\hat{\alpha}_j = \frac{\sum_{i=1}^{I} c_i \cdot y_{ij}}{\sum_{i=1}^{I} c_i}.$$

This is a weighted average of annotator j's votes, where the weights are the true labels. One approach is to substitute a proxy $\hat{c}_i$:

$$\hat{\alpha}_j = \frac{\sum_{i=1}^{I} \hat{c}_i \cdot y_{ij}}{\sum_{i=1}^{I} \hat{c}_i}.$$

Likewise, for the specificity:

$$\hat{\gamma}_j = \frac{\sum_{i=1}^{I} (1-\hat{c}_i)(1-y_{ij})}{\sum_{i=1}^{I} (1-\hat{c}_i)}.$$

Under the ground-truth path, we can get c_i directly from a labelled subset, and use this as a plug-in estimate in the above. If we are willing to trust the LLMs, or believe they are conditionally independent, we can construct a proxy via Leave-One-Out Majority Vote.

Estimating accuracies under the reliable raters assumption If we make assumptions on the LLMs themselves, we can use an aggregation of votes as a plug-in estimate of $\hat{c}$. A natural first attempt is the majority vote: set $\hat{c}_i = 1$ when more than half the LLMs vote 1, and $\hat{c}_i = 0$ otherwise. The problem is that this proxy includes LLM j's own vote. When estimating α_j, the numerator $\sum_i \hat{c}_i \cdot y_{ij}$ counts every item where LLM j voted 1 and the majority agreed. Because LLM j is part of the majority, its vote pushes the majority toward agreement with itself. The result is a mechanical upward bias in $\hat{\alpha}_j$: the estimator credits LLM j for agreement it helped create.

The fix is to exclude LLM j when forming the proxy for LLM j. Define the leave-one-out majority vote:

$$\hat{c}_{i,-j} = \mathbb{I}\left[\frac{1}{K-1}\sum_{k \neq j} y_{ik} > 0.5\right].$$

This is the majority vote of the other $K-1$ LLMs. It does not include y_{ij}, so the proxy and the vote being evaluated are functions of different raters. The proxy $\hat{c}_{i,-j}$ depends on j: each LLM gets its own proxy, formed from the votes of the others.

Estimating accuracy terms under the item model The item model of Section 2.3 supplies a score $\hat{e}(x_i)$, the conditional probability of class one given the features. We can estimate the accuracies of each LLM by regressing the votes on the fitted score, so that we have $y \sim \hat{e}(x)$.

By Lemma 1 and the law of total expectation, the conditional mean of a vote given the features is linear in the score:

$$\mathbb{E}[y_{ij} \mid x_i] = (1 - \gamma_j) + (\alpha_j + \gamma_j - 1)\, e(x_i).$$

The intercept is $1 - \gamma_j$ and the slope is $\alpha_j + \gamma_j - 1$. Regressing LLM j's votes on the score gives us:

$$\hat{\gamma}_j = 1 - \widehat{\text{intercept}}, \qquad \hat{\alpha}_j = \widehat{\text{intercept}} + \widehat{\text{slope}}.$$

Lemma 2 gives the derivation.

The prevalence The base rate $\pi = P(c_i = 1)$ is estimated differently depending on the path. Under the ground-truth path, $\hat{\pi}$ is the sample mean of the observed labels. Under the reliable-rater path, $\hat{\pi}$ is the fraction of items for which the full majority vote is positive:

$$\hat{\pi} = \frac{1}{I} \sum_{i=1}^{I} \mathbb{I}\left[\frac{1}{K} \sum_{j=1}^{K} y_{ij} > 0.5 \right].$$

Under the item-model path, $\hat{\pi}$ is the mean of the fitted score: $\hat{\pi} = \frac{1}{I} \sum_{i=1}^{I} \hat{e}(x_i)$.

Correlation Terms Once we have estimates of the accuracies and the prevalence, we are then able to estimate the correlation matrix R.

Define the pairwise agreement rate:

$$S_{jj'} = \frac{1}{I} \sum_{i=1}^{I} y_{ij} \cdot y_{ij'}$$

$S_{jj'}$ is a sample quantity consistent for $P(y_{ij} = 1 \cap y_{ij'} = 1)$.

From Equation (1) above, we know that:

$$P(\mathbf{y}_i \mid \pi, \boldsymbol{\alpha}, \boldsymbol{\gamma}, R) = \pi \cdot P(\mathbf{y}_i \mid c_i = 1, \boldsymbol{\alpha}, R) + (1 - \pi) \cdot P(\mathbf{y}_i \mid c_i = 0, \boldsymbol{\gamma}, R)$$

So that, applying this to the event $\{y_{ij} = 1 \cap y_{ij'} = 1\}$, which involves only annotators j and j':

$$P(y_{ij} = 1 \cap y_{ij'} = 1) = \pi \cdot \Phi_2(\mu_{1,j},\ \mu_{1,j'};\ \rho_{jj'}) + (1 - \pi) \cdot \Phi_2(\mu_{0,j},\ \mu_{0,j'};\ \rho_{jj'}) \qquad (2)$$

Where $\Phi_2(a, b;\ \rho)$ is the bivariate normal CDF. Now, from Section 3.2 above, we have plug-in estimators for α and γ, so we can set:

$$\hat{\mu}_{1,j} = \Phi^{-1}(\hat{\alpha_j}) \qquad \hat{\mu}_{0,j} = \Phi^{-1}(1 - \hat{\gamma}_j)$$

We can plug these, and the sample estimate of $P(y_{ij} = 1 \cap y_{ij'} = 1)$, in to Equation (2) to get:

$$\hat{S}_{jj'} = \hat{\pi} \cdot \Phi_2(\hat{\mu}_{1,j},\ \hat{\mu}_{1,j'};\ \rho_{jj'}) + (1 - \hat{\pi}) \cdot \Phi_2(\hat{\mu}_{0,j},\ \hat{\mu}_{0,j'};\ \rho_{jj'})$$

This is an equation with one unknown: $\rho_{jj'}$. Write the right-hand side as the function $h_{jj'}(\rho_{jj'})$. This has an inverse because it is continuous and strictly increasing in ρ. This allows us to solve for:

$$\hat{\rho}_{jj'} = h_{jj'}^{-1}(\hat{S}_{jj'})$$

This is computed numerically by bisection on $\rho \in (-1, 1)$ (see Appendix D).

This gives us the correlation matrix $\hat{R}$, which converges to the true R under the conditions of Theorem 1. We require $K < I$ so that $\hat{R}$ is nonsingular. In practice, we also want I large enough so that the finite-sample error of each $\hat{\rho}_{jj'}$ is small.

In our code, we implement two additional options for the finite-sample case. First, if $\hat{R}$ is in not positive semi-definite, it is projected to the nearest correlation matrix, and the user is warned. Second, we allow for $\hat{R}$ to be shrunk towards the identity matrix in cases where I is small, and estimation of the entries $\rho_{jj'}$ is noisy. This does involve a bias-variance trade-off, as it reduces the variance of the estimated entries, at the cost of a bias towards independence. It is recommended, however, to apply the method in cases where I is large enough for $\hat{R}$ to be estimated precisely without shrinkage.

3.3 Model outputs: Point estimates, confidence intervals, and flags for follow-up

We now have what we need to be able to calculate useful statistics. Given the estimated parameters $\hat{\pi}$, $\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\gamma}}$, and $\hat{R}$, we can compute the conditional probability of the label, given the parameters; define a statistical procedure to flag items for analyst review and follow-up; generate confidence intervals, and estimate LLM biases on individual items.

Item classifications. The probability that item i has true label 1, given its observed vote pattern and the estimated parameters, is:

$$\hat{P}(c_i = 1 \mid \mathbf{y}_i, \hat{\pi}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\gamma}}, \hat{R}) = \frac{\hat{\pi} \cdot P(\mathbf{y}_i \mid c_i = 1,\, \hat{\boldsymbol{\alpha}},\, \hat{R})}{\hat{\pi} \cdot P(\mathbf{y}_i \mid c_i = 1,\, \hat{\boldsymbol{\alpha}},\, \hat{R}) \;+\; (1 - \hat{\pi}) \cdot P(\mathbf{y}_i \mid c_i = 0,\, \hat{\boldsymbol{\gamma}},\, \hat{R})}$$

This is the conditional probability formula from (1), evaluated at the estimated parameters. The point estimate for item i's label is:

$$\widehat{\text{Label}}(\mathbf{y}_i) = \mathbb{I}\Big[\hat{P}(c_i = 1 \mid \mathbf{y}_i, \hat{\pi}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\gamma}}, \hat{R}) > 0.5\Big]$$

This classification incorporates annotator accuracy and dependence. When two annotators with high estimated correlation $\hat{\rho}_{jj'}$ both vote 1, their agreement contributes less evidence for $c_i = 1$ than when two independent annotators agree, because part of their agreement is explained by correlation rather than by the true label.

LLM biases Since we have votes and estimated ground truth classification probabilities, it is straightforward to calculate a vector of residuals for each model:

$$\widehat{\text{Bias}_j} = \boldsymbol{y}_j - \hat{P}(\boldsymbol{c} \mid \mathbf{y}_i, \hat{\pi}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\gamma}}, \hat{R})$$

This is useful both if we are interested in the performance of a particular LLM or LLM model family on a given classification task, and if we are interested in assessing particular LLMs for item-level biases on particular items. We discuss this last point further in Section 4.

Flags for follow-up. The classification probability itself measures how ambiguous item i is given the votes: values near 0.5 indicate the votes are inconclusive. We flag items for follow-up when its standard deviation exceeds a user-specified threshold, parameterized by τ. Write:

$$\hat{\sigma}_i = \text{sd}\big(\hat{P}(c_i = \cdot \mid \mathbf{y}_i)\big).$$

Then, we flag items whose classification standard deviation exceeds the threshold:

$$\widehat{\text{Flag}}(\mathbf{y}_i) = \mathbb{I}\{\hat{\sigma}_i > \tau\}.$$

Flagged items can be sent either for manual follow-up, or automated review by LLMs. Note that the classification uncertainty holds the parameter values fixed at their estimated value. That is, it reflects uncertainty that takes the model's estimates of the rater dependencies and accuracies as a given. It does not reflect the *estimation* uncertainty in the parameters.

Confidence intervals. To quantify how sensitive this classification is to parameter uncertainty, we use the nonparametric bootstrap. For each replicate $b = 1, \ldots, B$: resample I items with replacement, re-run the full estimation pipeline, and recompute each item's classification probability at the new parameter estimates $(\hat{\pi}^{(b)}, \hat{\boldsymbol{\alpha}}^{(b)}, \hat{\boldsymbol{\gamma}}^{(b)}, \hat{R}^{(b)})$. The 95% confidence interval for item i's classification is:

$$\text{CI}_{0.95}(i) = \Big[Q_{0.025}\Big(P^{(b)}(c_i = 1 \mid \mathbf{y}_i, \hat{\pi}^{(b)}, \hat{\boldsymbol{\alpha}}^{(b)}, \hat{\boldsymbol{\gamma}}^{(b)}, \hat{R}^{(b)})\Big),\; Q_{0.975}\Big(P^{(b)}(c_i = 1 \mid \mathbf{y}_i, \hat{\pi}^{(b)}, \hat{\boldsymbol{\alpha}}^{(b)}, \hat{\boldsymbol{\gamma}}^{(b)}, \hat{R}^{(b)})\Big)\Big]$$

where $Q_p(\cdot)$ denotes the pth quantile across bootstrap replicates. The interval width measures sensitivity of the label to the estimated parameters. An item can have a narrow interval and a large $\hat{\sigma}_i$, when the parameters are well estimated but the votes are split, and these are the items the flag is designed to catch.

3.4 Identification of the Correlated Dawid-Skene Model

Theorem 1 (Identification, informal statement)**.** *Suppose any one of the following holds:*

(i) Assumption 1: ground truth labels are observed on a subset.

(ii) Assumption 2: the raters are reliable (either per se or via conditional independence).

(iii) Assumptions 3, 4, 5, 6: feature extraction is mechanical, the recipe satisfies semantic exclusion, the fitted score is consistent, and the score varies across items.

Then as $I \to \infty$:

$$\hat{\alpha}_j \xrightarrow{p} \alpha_j, \quad \hat{\gamma}_j \xrightarrow{p} \gamma_j, \quad \hat{\rho}_{jj'} \xrightarrow{p} \rho_{jj'}$$

for every j and every pair $j \neq j'$. Under conditional independence (Assumption 2(b)), the limits hold as $K \to \infty$ in addition: with K fixed, the estimators have a bias that converges to zero as K grows. The classification probabilities, flags, and LLM biases converge to their population counterparts.

The proof, which is given in Appendix A.4 together with a formal statement of the theorem, has two steps. The first step is path-specific: each assumption identifies the accuracies $(\alpha_j, \gamma_j)_{j=1}^K$ by one of the identification arguments above. Then, given the identified accuracies, we can find a unique solution for each pairwise correlation term $\rho_{jj'}$. Collecting these together gives us the estimated correlation matrix $\hat{R}$, which we can plug into the multivariate probit model described above.

When the leave-one-out majority vote is wrong on some items, the accuracy estimators under Assumption 2 are biased. Proposition 1 in Appendix A.3 gives the exact form of this bias. The bias of $\hat{\alpha}_j$ depends on two quantities: the false positive rate of the leave-one-out majority vote, and the covariance between that vote and the vote of LLM j within each class. The covariance is zero when the LLMs are conditionally independent. The bias is a concern when the LLMs are correlated, their accuracy is low, and the leave-one-out majority vote is wrong on many items. Corollary 1 bounds the bias in terms of the error rate of the leave-one-out majority vote.

4 Post-Hoc Interpretability via Sparse Autoencoders

The previous section produces numerical outputs that are useful for statistical analysis, and for downstream inference as part of a larger task. But in practice, we are interested both in qualitative answers to substantive questions, and in understanding why models made the decisions they made.

Here, we use sparse autoencoders (SAEs) to interpret the outputs from the previous section. We adapt the HypotheSAEs method of Movva et al. [15]. We differ in that we use SAEs to interpret the outputs of the model in the previous stage, which gives us both interpretable answers to substantive questions, and model diagnostics, that are grounded in the contents of the original documents.

The procedure answers a question of the form: which concepts in the documents predict a given output of the model? It works in three steps. First, it learns a dictionary of concepts from the text, where each concept is a pattern that recurs across sentences, such as references to taxation or to national sovereignty. Second, it asks which of these concepts predict the output of interest, whether that is the classification of an item, the flag on an item, or the deviation of one rater from the others. It keeps the few concepts that predict the output and discards the rest. Third, it describes each surviving concept by collecting the sentences that express it most strongly from the documents, and, additionally, asking an LLM to state what they have in common. The model provides both the sampled sentences and the LLM annotations as outputs. This step is descriptive, not causal: it is intended to aid the user in understanding output and to automate the process of drawing qualitative conclusions from data, as well as in diagnosing the performance of the annotation process.

A sparse autoencoder is a pair of linear functions, an encoder and a decoder, combined with a sparsity constraint that is intended to preserve a small number of interpretable features.

The encoder maps an embedding to a wide hidden layer and keeps its k largest entries, so each embedding is represented by k active units. The decoder maps the hidden layer back to the embedding space and reconstructs the embedding from those k units. The goal is a representation of each embedding in terms of a few units drawn from a fixed set shared across the corpus.

Training adjusts both maps to minimize the squared distance between each embedding and its reconstruction. The sparsity constraint forces the model to reconstruct each embedding from a few units, so a unit earns its place only if it captures variation shared across many embeddings. A unit that fires for one embedding alone does not reduce the total error enough to survive. Each unit therefore corresponds to a pattern that recurs in the corpus, and we treat these units as semantic features.

We train the autoencoder on sentence embeddings rather than document embeddings. A document contains a large amount of semantic content, and a single embedding of it averages that content. A sentence contains less, so a feature trained on sentence embeddings corresponds to a narrower piece of semantic content.

To connect features to a model output, we pool each feature's sentence activations to the document, regress the output on the pooled activations with an L^1 penalty, and keep the features with non-zero coefficients. These are the features that most strongly explain the output.

We then label each kept feature by its content. We collect the sentences that activate it most and least, pass these to an LLM, and record the concept the LLM reports as separating the two sets. The researcher also reads the sampled sentences directly, and so sees which content activated the feature rather than relying on the label alone.

As an example, in the manifesto case study below, the sentence 'Tougher criminal sentencing and penalties' is an LLM-generated label that describes the feature that predicts that annotators assign a high probability that a given manifesto is conservative on social policy, and "Att kriminella handlingar får konsekvenser. [Criminal acts have consequences.]" is a sentence sampled from the 2010 manifesto of Swedish Christian Democrats that exemplifies the concept.

4.1 SAE Setup

Recall that we have a corpus of documents $\{D_i\}_{i=1}^{I}$. We split each document into sentences and write $D_i = \{s_{i1}, \ldots, s_{in_i}\}$ for its n_i sentences. A pretrained embedding model Emb maps each sentence to a vector:

$$v_{ig} = \mathrm{Emb}(s_{ig}) \in \mathbb{R}^d$$

This gives us a matrix V of sentence embeddings, with one row per sentence. Next, we train a k-sparse autoencoder [13] on the embeddings V. The autoencoder encodes each embedding v_{ig} as a sparse vector of feature activations $w_{ig} \in \mathbb{R}^m$, then decodes w_{ig} back to the embedding space:

$$\begin{aligned} w_{ig} &= \mathrm{ReLU}\big(\,\mathrm{TopK}_k\big(W_{\text{enc}}(v_{ig} - b)\big)\big), \\ \hat{v}_{ig} &= W_{\text{dec}}\, w_{ig} + b, \end{aligned}$$

where $W_{\text{enc}} \in \mathbb{R}^{m \times d}$, $W_{\text{dec}} \in \mathbb{R}^{d \times m}$, and $m > d$. The operator TopK_k sets all but the k largest entries to zero, so w_{ig} has at most k non-zero entries. The parameters are fit by minimizing the reconstruction error $\sum_{i,g} \|v_{ig} - \hat{v}_{ig}\|_2^2$ by stochastic gradient descent. The TopK selection is held fixed within each gradient step, so on each sentence the gradient updates only the features that are active for it. Here m is the number of features the autoencoder can represent across the corpus, and k is the number that can be active in any one sentence.

The quantities of interest are defined at the level of the document, so we pool the sentence activations to the document. For feature f and document i, take the activations of that feature across the document's sentences, keep the κ largest, and average them:

$$A_{if} = \frac{1}{\kappa} \sum_{g \in \text{top-}\,\kappa(i,f)} w_{igf},$$

where top- $\kappa(i, f)$ indexes the κ sentences of document i with the largest activation of feature f. This gives a document-level activation matrix $A \in \mathbb{R}^{I \times m}$, with row i the pooled activations for document i and column f the activation of feature f across the corpus.

Then, we find the features that predict a quantity of interest. Let t_i be a quantity associated with document i, collected into the vector $t \in \mathbb{R}^I$. We fit an L^1-penalized regression of t on the activation matrix:

$$\hat{\beta} = \arg\min_{\beta}\ \frac{1}{I}\big\|t - A\beta\big\|_2^2 + \lambda\|\beta\|_1,$$

and set the penalty λ by binary search, so that a chosen number of coefficients are non-zero. The features with non-zero coefficients are the features that predict the quantity.

Finally, each selected feature is described by sentences sampled from the documents themselves. We rank the sentences by their activation of the feature, conditional on the activation being nonzero. The procedure then samples sentences from the high and low ends of that ranking, and prompts an LLM to state the concept that is present in the high-activation sentences and absent in the low-activation ones. We report this concept label together with the highest-activation sentences, so the analyst observes 'primary source material' sampled from the documents, in addition to the LLM-generated label.

4.2 Applying the SAE to outputs of the Correlated Dawid-Skene Model

The previous section describes an approach applied generically to any output. But we specifically use SAEs to provide interpretable summaries of the output of the Correlated Dawid-Skene model. We run the feature-selection regression once for each of the three quantities below, and report the features it selects. Each regression is a Lasso of the target on the feature activations, with the penalty set so that a small number of features are selected.

- **Classification probability** $\hat{P}(c_i = 1 \mid \boldsymbol{y}_i, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\gamma}}, \hat{R})$. Regressing this classification probability on the features explains which document features were relevant to the probability of being assigned the positive class. In the manifesto case they are the content that separates left from right, so the regression states, in terms of the documents, which manifesto contents were relevant to describing a document as left-wing on a particular issue.
- **Rater deviation** $\widehat{\text{Bias}}_{ij} = y_{ij} - \hat{c}_i$. Regressing rater j's deviation on the features recovers the content to which rater j responds beyond the consensus. If a given LLM is biased on specific content, this provides the user with diagnostic information that this may be the case.
- **Flag** $\widehat{\text{Flag}}_i = \mathbb{I}\{\hat{\sigma}_i > \tau\}$. Regressing the flag on the features recovers the items on which the LLMs disagree, and the model assigns high uncertainty. The selected features are those that recur in the items with high classification standard deviation. This tells us which document features are hard for the annotators collectively to explain.

The bias regression connects to the item model of Section 2.3. A feature that predicts rater j's deviation from the consensus is a candidate for a feature that violates the semantic exclusion restriction for rater j: it is associated with how rater j votes, beyond the truth as proxied by the consensus. The regression detects features to which raters respond differently from one another. It does not detect features to which all raters respond in the same way, because a common response is absorbed into the consensus $\hat{c}_i$. The layer can therefore falsify the semantic exclusion restriction for a specific rater, by showing that a given feature that predicts that rater's deviation.

Note that the ex post interpretation layer is *not* a causal inference layer: it is intended to produce qualitative, descriptive output to provide diagnostics for the model. The idea here is to associate the outputs of the previous layer with semantic information from the documents themselves. This gives us decisions that are associated with contrasts between positive and negative examples, and helps us to understand why a particular decision was made.

5 Empirical Demonstration

In this section, we assess the empirical performance of the method via simulation, and apply the method to the manifesto classification exercise described in [1].

First, via simulation, we highlight the case in which this method will improve the accuracy of the resulting point estimates: namely, when LLMs are correlated, and a simple average is overconfident about the number of independent sources of information provided by the LLM raters. We then show that this matters for uncertainty quantification: generating confidence intervals under the assumption that the individual raters are independent is anticonservative, leads to coverage below the desired level, and to confidence intervals that are narrow relative to the truth.

Second, we apply our method to the problem of classifying the ideological position of a set of manifestos collected by the Manifesto Project. We highlight how our method provides point estimates,

confidence intervals, estimated correlation matrices, and flags for user and automated follow-up to the user.

We then highlight the utility of the sparse autoecoder layer. We show that the SAE layer provides interpretable explanations of specific semantic content that explain why a given classification decision was made. The LLM labels map clearly onto specific issue dimensions that are recognizable to political scientists, and the example quotes are both relevant to the identified issue dimensions, and are sourced from parties from party families that plausibly have the relevant ideological stance on that issue.

We also detect biases in LLMs. Examining the estimated correlation matrix on the social dimension highlights that Deepseek models are least correlated with the decisions of other LLM models. Our qualitative results show that this disagreement in especially pronounced in the classification of nationalist and authoritarian content.

Finally, we highlight specific manifestos that are difficult to classify. These include two Greek parties from 2019, the far-right Golden Dawn, and Yanis Varoufakis's MeRa25 party, which was formed primarily in opposition to the terms of the Greek bailout, the Icelandic Liberal party in 2003, whose main issue was EU fisheries policy, and the Sweden's Left party in 2006, amid an internal split of the party into two nationwide organizations.

5.1 Simulation study: Aggregation of Correlated Raters

We run three simulations.

The first studies the case where we have 8 LLMs that are all reliable. In this setup, the 8 LLMs are equicorrelated with $\rho = .3$. Here, we can essentially trust the LLMs to make correct classification decisions. In this situation, majority vote is a good enough proxy for the vote. Here, there is a small gain to be had by adjusting for the accuracies of the LLMs, and there is no particular gain to be had by estimating the correlation matrix R in addition.

Table 1: Sim A: Reliable LLMs regime. $K = 8$, $I = 1000$, $\pi = 0.5$, accuracies drawn from $\mathrm{U}[0.70, 0.90]$, equicorrelation $\rho = 0.3$, $N_{\mathrm{sim}} = 100$. The three methods classify at approximately equal accuracy. In this regime the mean/majority vote is accurate, and the model contributes the interval and the flags rather than a better point estimate.

ρ	Majority	Dawid-Skene ($R = I$)	Correlated DS ($\hat{R}$)
0.3	0.909 (0.001)	0.915 (0.001)	0.914 (0.001)

Accuracy against the known labels, with Monte Carlo standard error in parentheses. Paired differences against the mean: correlated Dawid-Skene 0.005 (0.001), Dawid-Skene 0.006 (0.001).

In the second simulation, we study a case in which there is a fraction of adversarial LLMs, who err systematically. We first vary the fraction of adversarial LLMs, and second vary the size of the gold standard subset of labels, which is a way of varying the precision with which accuracies are estimated.

In the first case, gold standard labels improve the performance of both Dawid-Skene methods, because they are now able to use the information from the adversarial labelers as signal, where majority vote would instead incorrectly take the votes of the adversarial labelers as signal. Second, the gain from estimating the correlation matrix is noticeable when either the adversarial proportion is high, or, in this simulation, when the gold standard subset size reaches 25% of the sample size. That is because there is a finite sample cost to estimating $\hat{R}$, and it reduces overall error only when it can be estimated precisely enough.

Third, we consider a case where there are highly correlated labellers, and a small minority of independent labellers. Here, the independent labellers collectively have more signal than the dependent labellers, but majority vote treats the five dependent labellers as having more weight. Here, using the correlated Dawid-Skene model helps relative to majority vote when the identification assumptions

Table 2: Sim B: adversarial majority with a gold anchor. $K = 8$, $I = 1000$, $\pi = 0.5$. Good raters are drawn from $\mathrm{U}[0.75, 0.85]$ and are independent. Adversarial raters are drawn from $\mathrm{U}[0.15, 0.35]$ and are correlated at $\rho = 0.6$. Accuracies are anchored to a gold subset of size $|S|$. $N_{\mathrm{sim}} = 100$.

	Vote	Model-based		Paired diff.	
	Majority	DS ($R = I$)	Corr DS ($\hat{R}$)	Gain from accuracy	Gain from $\hat{R}$
Panel 1: fraction adversarial, $\lvert S\rvert = 100$					
0.25	0.854 (0.001)	0.963 (0.001)	0.930 (0.003)	0.109 (0.001)	−0.034 (0.003)
0.50	0.515 (0.001)	0.931 (0.001)	0.917 (0.003)	0.415 (0.002)	−0.014 (0.003)
0.625	0.372 (0.002)	0.872 (0.002)	0.875 (0.002)	0.499 (0.003)	0.003 (0.003)
0.75	0.251 (0.002)	0.855 (0.001)	0.859 (0.002)	0.604 (0.003)	0.004 (0.002)
Panel 2: gold-subset size $\lvert S\rvert$*,* $5/8$ *adversarial*					
20	0.330 (0.002)	0.895 (0.002)	0.865 (0.004)	0.565 (0.003)	−0.030 (0.004)
50	0.330 (0.002)	0.901 (0.002)	0.886 (0.003)	0.571 (0.003)	−0.015 (0.003)
100	0.330 (0.002)	0.905 (0.001)	0.904 (0.002)	0.575 (0.002)	−0.001 (0.002)
250	0.330 (0.002)	0.904 (0.001)	0.919 (0.002)	0.574 (0.002)	0.015 (0.002)

Accuracy against the known labels, with Monte Carlo standard error in parentheses. The first column is the fraction adversarial in Panel 1 and the gold-subset size in Panel 2.

hold: when we have access to one source of anchoring, the correlated Dawid-Skene model outperforms both the Dawid-Skene model and majority vote. The method performs worse than majority vote when the assumptions do not hold.

Table 3: Sim C: Correlated block with independent dissenters. $K = 8$, $I = 2000$, $\pi = 0.5$, accuracies drawn from $\mathrm{U}[0.65, 0.75]$, a block of five raters correlated at $\rho = 0.7$ ($K_{\mathrm{eff}} = 2.91$), $N_{\mathrm{sim}} = 100$.

	Vote	Model-based			Paired diff.
Accuracies	Majority	DS ($R = I$)	Corr DS ($\hat{R}$)	Corr DS (R true)	corr−naive
LOO	0.773 (0.001)	0.739 (0.001)	0.728 (0.001)	0.787 (0.001)	−0.011 (0.001)
Gold	0.773 (0.001)	0.781 (0.001)	0.802 (0.002)	0.822 (0.001)	0.021 (0.002)
Oracle	0.773 (0.001)	0.780 (0.001)	0.826 (0.001)	0.827 (0.001)	0.046 (0.001)

Accuracy against the known labels, with Monte Carlo standard error in parentheses. The leave-one-out row is the correlated block under the reliable-rater proxy, where the assumption does not hold. The gain from the correlation matrix appears when the accuracies are identified from valid labels.

Taken together, we see that the correlated Dawid-Skene model is more accurate than majority vote or the Dawid-Skene model when there is sufficient data to estimate $\hat{R}$ precisely; when the identification assumptions hold; when the majority actually contains less information than the minority, due to their being correlated; and in the presence of adversarial labels.

5.2 Manifestos: Correlated Dawid-Skene Model

Classification probabilities We applied the correlated Dawid-Skene model to the Manifesto classification task of [1]. In this study, LLMs were asked to classify political party manifestos along six issue dimensions (Economic, Immigration, Social, EU, Decentralization and the Environment) on an ordinal scale from 1-7. These were compared to gold standard labels from the Chapel Hill Expert Survey.

The first key output of the paper is the 'operating characteristic' display from the correlated Dawid-Skene model. This illustrates, for each issue area, the estimated probability that a given item belongs to a specific class, given which rater voted for it, and their estimated accuracy and dependence structure. It also illustrates the level of conditional uncertainty associated with each vote, where the conditioning involves taking the parameters as fixed (that is, uncertainty that does not include estimation error). This reflects both which rater voted for which items, and the aggregation model's

parameters that encode how correlated that vote is with other votes, and how accurate that LLM is. LLMs that are high accuracy and low dependence are upweighted in the classification decision.

This illustration also highlights which items are flagged for review. We return as a list items that have large estimated standard deviation. These are items for which rater aggregations express high uncertainty, given the estimated dependence structure and accuracies.

Correlation Matrices Next, we can also return the estimated correlation matrices. These illustrate, for given issue dimensions, the pattern of LLM conditional correlation, where, again, conditional correlation means *residual agreement after the ground truth labels* have been taken into account. First, a note on these correlation matrices: these are marginalized over multiple scoring runs, so the diagonal entries do not sum to 1: they instead represent LLM correlations with themselves over many runs.

Two patterns are worth remarking on. First, the Deepseek models make noticeably different predictions to other LLMs on the social dimension, which is suggestive evidence of an 'in-house' bias on the classification of social issues. Second, correlation varies across issue dimensions: LLMs should not be thought of as possessing specific correlation relationships with one another, but should be understood as having *task-specific* correlations, where LLM characteristics are a function of the task they have been assigned. Third, the *magnitude* of LLM correlation also varies across issue dimensions: the level of agreement on classification decisions varies significantly, so that LLMs could be treated as 'more independent' for some tasks, and 'less independent' for others. However, it is worth stressing that this cannot necessarily be known ex ante, because these are the conditional correlation matrices given the accuracy parameters. Differences in accuracies across tasks can also explain differences in apparent correlation across LLMs.

5.3 Manifestos: Interpretability Outputs

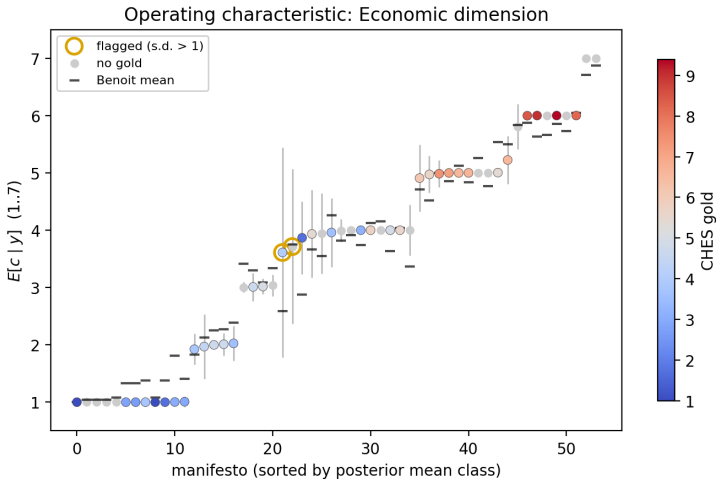
Next, we consider the outputs of the SAE layer. We have three tables (Tables 4 to 6), corresponding to our three regression targets of interest. The tables follow the conclusion.

First, in terms of class predictions, we should notice that the identification of issues clusters well into issues traditionally associated with the issue dimension. This 'tree' structure rationalizes the LLM summary step: we get out specific issues that are connected relevantly to the dimension of interest. So the economic dimension loads on sentences that refer to taxation and inequality: the VVD, a Dutch, right-libertarian party, wants lower taxes; LMP, a Green party in Hungary, mentions social justice in their manifesto. Second, the content of the exemplar sentences is clearly relevant to both the issue and the dimension of political competition. Third, the party families and electoral contexts of the exemplar sentences are aligned with the relevant issue positions and dimensions of political competition.

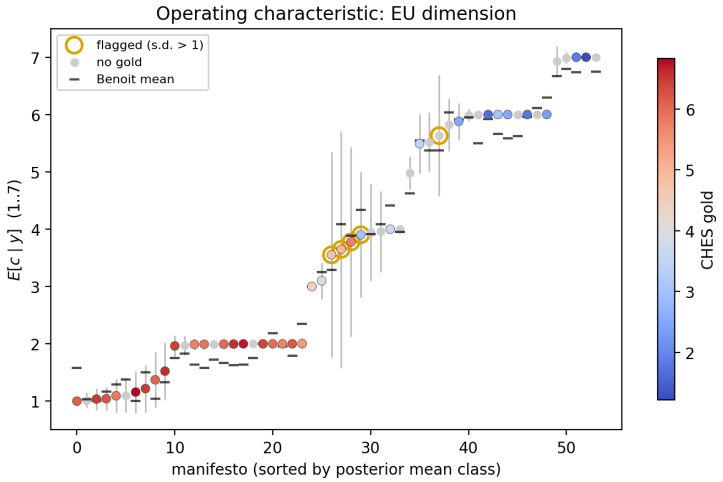
Next, we extract features that explain LLM biases. Here, we find that LLMs struggle with idiosyncratic sub-issues: Hungarian nationalism, Bulgarian nationalism, and Plaid Cymru's dissatisfaction with the United Kingdom's internal fiscal transfers are sources of disagreement for the individual raters. Two items, the FPÖ's proposal to raise a national militia, is mildly suggestive of a possible conservative-authoritarian bias of Deepseek, while Deepseek also thinks that the Hungarian Socialist Party's nationalistic rhetoric is relevant to the economic dimension (in the right-wing direction).

Third, the model also returns individual manifestos that were difficult for the raters to classify. Two of the difficult manifestos to classify are from extreme right and extreme left parties in the context of the Greek 2019 election, which was a direct response to the Greek Bailout in 2018. These parties took extreme positions on an issue that was fairly idiosyncratic in global terms. The Liberal Party in Iceland, despite its name, is a single-issue party primarily concerned with EU fishing quotas; and the Swedish Left experienced an internal party split in 2006, with two separate branches fielding candidates in the national election of that year.

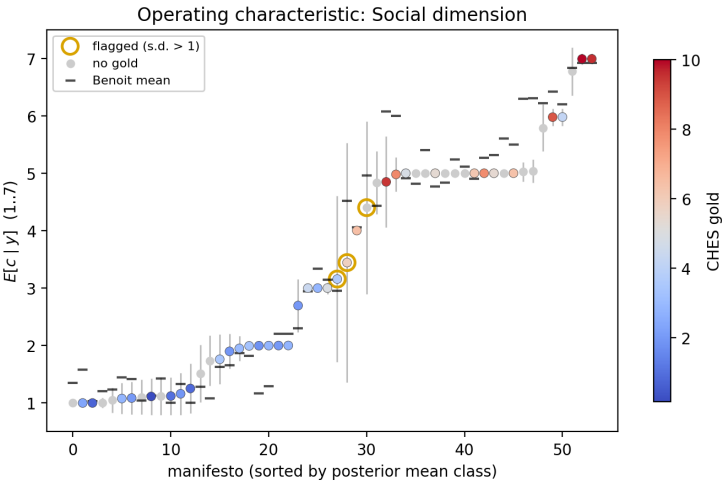
This output is of course qualitative, and it does not, by itself, definitively resolve experimental hypotheses. But it is *illustrative*, and it produces output from the data that generates substantive content from the classification exercise, and it does so in an automated and systematic way. In political science specifically, which is concerned with theory-building as well as theory-testing, this content is practically useful.



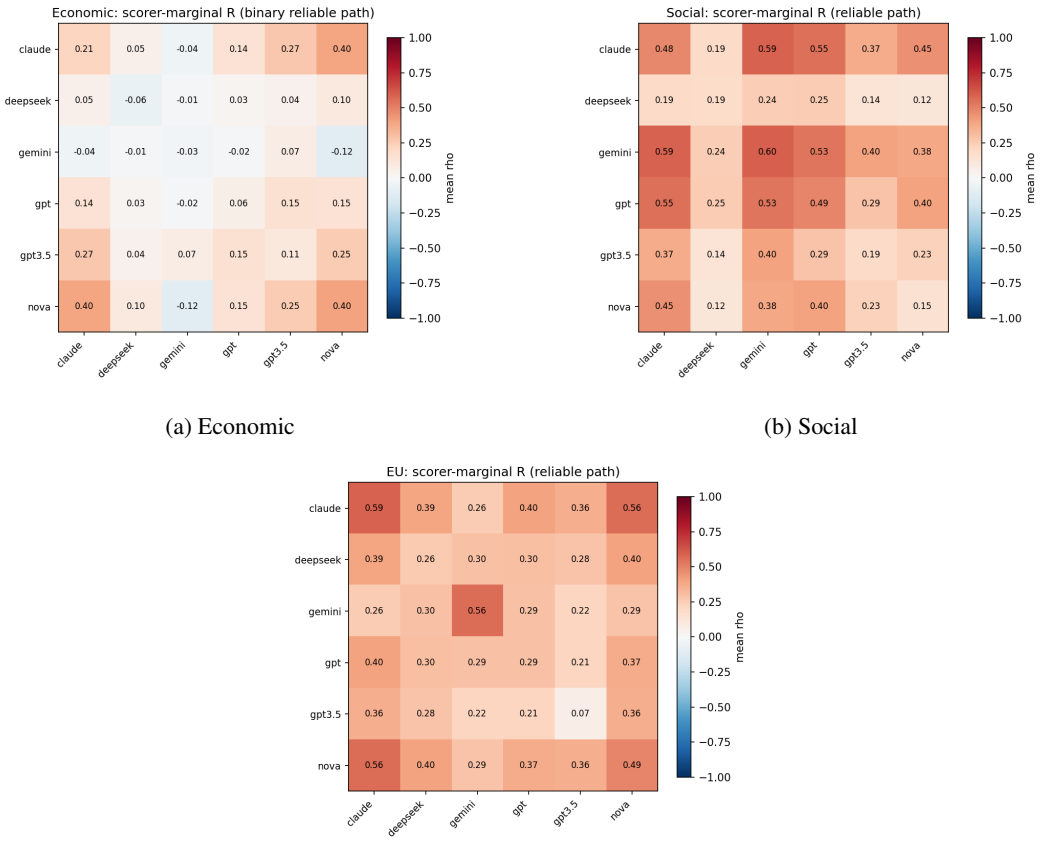
(a) Economic



(b) EU



(c) Operator characteristics: class probabilities, uncertainties, and flags.



(a) Economic

(b) Social

(c) EU

Figure 4: Estimated rater correlation R by dimension, scorer-marginal.

6 Conclusion

We study the problem of aggregating dependent LLM annotators. We describe the circumstances under which this is possible, showing that a researcher must make some assumptions on the annotators, or on items, or must have access to ground truth labels. We modify the Dawid-Skene model of crowdsourced annotations to incorporate an estimated correlation matrix, and use sparse autoeconders to generate natural language output that aids researchers in understanding classification decisions, highlights why individual annotators diverge from the consensus, and explain what items annotators are likely to diverge on. These tools are intended to be useful both for downstream causal inference, in which annotation is used to generate treatment indicators or covariates, and where uncertainty estimates associated with these measures is necessary. They also provide quaititative information that is helpful for theory-building in a substantive domain. While the focus of this article is on annotation decisions, the framework applies broadly to any context in which an ensemble of LLM agents is used to make decisions or predictions. As a result, it is relevant to the domain of automated decision-making with LLM agents, and to a wide range of practical applications. In future work, we will study the problem of combining fairness constraints with aggregation decisions.

Table 4: Class-loading features. Content predicting the aggregated score, recovered without supervision by a sparse autoencoder. The strongest positively loaded feature in each dimension is the substantively correct concept at the correct pole.

Dimension, pole	Issue	Topic (feature)	Example sentence (source; English)	η^2
Economic, high	Taxation	Tax systems and levels	*"De VVD wil een rechtvaardig belastingstelsel."* (Netherlands, VVD; the VVD wants a fair tax system)	0.21
Economic, low	Inequality	Social problems framed as suffering	*"Egy igazságtalanságokkal súlyosan terhelt társadalom nem fenntartható."* (Hungary, LMP; a society heavily burdened by injustice is not sustainable)	0.37
EU, anti	Sovereignty	Rejection of supranational or central authority	*"Sverige ska inte gå med i Nato."* (Sweden, Greens; Sweden should not join NATO)	0.54
Social, high	Family	Family as the foundation of society	*"Den viktigaste är familjen."* (Sweden, Christian Democrats; the most important thing is the family)	0.39
Social, high	Law and order	Tougher criminal sentencing	*"Att kriminella handlingar får konsekvenser."* (Sweden, Christian Democrats; that criminal acts have consequences)	0.51

The Economic taxation feature ($\eta^2 = 0.21$) is the cleanest result: content with little country-identity load. The remaining features are on-dimension but carry more identity, reflected in higher η^2.

Table 5: Sign-disagreement features. Content that raters place in opposite directions relative to the consensus. The disagreement concentrates on nationhood and sovereignty, and recurs across dimensions.

Dimension	Issue	Topic (feature)	Raters (raise / lower)	Example sentence (source; English)
Economic	Nationhood	Nation invoked as national subject	raise: gpt_deepseek; lower: gemini_gpt, deepseek_nova, deepseek_claude	*"Igen az erős köztársaságra, és igen a sikeres Magyarországra."* (Hungary, MSzP; yes to a strong republic, yes to a successful Hungary)
Economic	EU role	Bulgaria's strategic role within the EU	raise: gemini_deepseek, claude_gemini; lower: deepseek_nova, gemini_nova, gpt_claude	Bulgarian sovereignty-in-Europe content (Bulgaria)
Social	Protective state	Strong protective state defending homeland	raise: deepseek_deepseek, gemini_deepseek, gpt_gpt; lower: gpt3.5_gpt	*"Starke freiwillige Miliz-Komponente für den Heimatschutz."* (Austria, FPÖ; strong voluntary militia for homeland defence)
EU	Fiscal transfers	Budgets and funding allocations	split across raters	*"We have a worse funding settlement and fewer of the economic levers over tax and borrowing…"* (United Kingdom, Plaid Cymru)

The nationhood, homeland, and sovereignty features form a cluster, which is the evidence that the disagreement is substantive rather than a single-feature artifact. Nationalist content is both country-identifying (high η^2) and rater-dividing.

Table 6: Most-contested items, measured by the dispersion of the raw votes (threshold-free, independent of the aggregation model). The two most-contested manifestos are Golden Dawn and MeRA25, the radical-right and radical-left-Europeanist poles of Greek politics.

Dimension	Manifesto (country, year, party)	Vote split (score:count)	SD	Contested feature
Economic	Greece, 2019, Golden Dawn (34720)	1:7 2:5 4:1 5:2 6:1 7:1	1.94	Nation invoked as national subject
EU	Greece, 2019, MeRA25 (34215)	1:4 2:3 3:1 4:1 5:8 6:5	1.87	Budgets and funding allocations
Social	Iceland, 2003, Liberal Party	1:1 2:1 3:1 4:1 7:3	2.38	Rejection of supranational authority
Social	Sweden, 2006, Left (11220)	1:19 2:4 3:1 4:1 7:1	1.31	Rejection of supranational authority

Contestation is the standard deviation of the raw 1–7 votes.

References

[1] Kenneth Benoit, Scott De Marchi, Conor Laver, Michael Laver, and Jinshuai Ma. Using large language models to analyze political texts through natural language understanding. *American Journal of Political Science*, 2025.

[2] Adam Bouyamourn. Why LLMs hallucinate, and how to get (evidential) closure: Perceptual, intensional, and extensional learning for faithful natural language generation. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3181–3193, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.192. URL https://aclanthology.org/2023.emnlp-main.192/.

[3] Emmanuel J. Candès, Andrew Ilyas, and Tijana Zrnic. Probably approximately correct labels, 2025. URL https://arxiv.org/abs/2506.10908.

[4] Jerome Cornfield, William Haenszel, E. Cuyler Hammond, Abraham M. Lilienfeld, Michael B. Shimkin, and Ernst L. Wynder. Smoking and lung cancer: Recent evidence and a discussion of some questions. *JNCI: Journal of the National Cancer Institute*, 22(1):173–203, 01 1959. ISSN 0027-8874. doi: 10.1093/jnci/22.1.173. URL https://doi.org/10.1093/jnci/22.1.173.

[5] A. P. Dawid and A. M. Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1): 20–28, 1979.

[6] Naoki Egami, Musashi Hinck, Brandon M. Stewart, and Hanying Wei. Using imperfect surrogates for downstream inference: Design-based supervised learning for social science applications of large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 68589–68601, 2023.

[7] Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120, 2023. doi: 10.1073/pnas.2305016120.

[8] Kristina Gligorić, Tijana Zrnic, Cinoo Lee, Emmanuel J. Candès, and Dan Jurafsky. Can unconfident llm annotations be used for confident conclusions?, 2025. URL https://arxiv.org/abs/2408.15204.

[9] Andrew Halterman and Katherine A Keith. Codebook llms: Evaluating llms as measurement tools for political science concepts. *Political Analysis*, pages 1–17, 2025.

[10] Michael Heseltine and Bernhard Clemm von Hohenberg. Large language models as a substitute for human experts in annotating political text. *Research & Politics*, 11(1):20531680241236239, 2024. doi: 10.1177/20531680241236239.

[11] Elliot Kim, Avi Garg, Kenny Peng, and Nikhil Garg. Correlated errors in large language models. *arXiv preprint arXiv:2506.07962*, 2025.

[12] Gaël Le Mens and Aina Gallego. Positioning political texts with large language models by asking and averaging. *Political Analysis*, 33(3):274–282, 2025.

[13] Alireza Makhzani and Brendan Frey. k-sparse autoencoders, 2014. URL https://arxiv.org/abs/1312.5663.

[14] Iman Modarressi, Jann Spiess, and Amar Venugopal. Causal inference on outcomes learned from text, 2025. URL https://arxiv.org/abs/2503.00725.

[15] Rajiv Movva, Kenny Peng, Nikhil Garg, Jon Kleinberg, and Emma Pierson. Sparse autoencoders for hypothesis generation, 2025. URL https://arxiv.org/abs/2502.04382.

[16] Ulf Olsson. Maximum likelihood estimation of the polychoric correlation coefficient. *Psychometrika*, 44(4):443–460, 1979.

[17] Yinsheng Qu, Ming Tan, and Michael H Kutner. Random effects models in latent class analysis for evaluating accuracy of diagnostic tests. *Biometrics*, pages 797–810, 1996.

[18] Luca Rettenberger, Markus Reischl, and Mark Schutera. Assessing political bias in large language models. *Journal of Computational Social Science*, 8(2):42, 2025.

[19] PAUL R. ROSENBAUM and DONALD B. RUBIN. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 04 1983. ISSN 0006-3444. doi: 10.1093/biomet/70.1.41. URL https://doi.org/10.1093/biomet/70.1.41.

[20] Philipp Schoenegger, Indre Tuminauskaite, Peter S Park, Rafael Valdece Sousa Bastos, and Philip E Tetlock. Wisdom of the silicon crowd: Llm ensemble prediction capabilities rival human crowd accuracy. *Science Advances*, 10(45):eadp1528, 2024.

[21] Petter Törnberg. ChatGPT-4 outperforms experts and crowd workers in annotating political Twitter messages with zero-shot learning. *arXiv preprint arXiv:2304.06588*, 2023.

[22] Petter Törnberg. Large language models outperform expert coders and supervised classifiers at annotating political social media messages. *Social Science Computer Review*, page 08944393241286471, 2025. doi: 10.1177/08944393241286471.

[23] Kai-Cheng Yang and Filippo Menczer. Accuracy and political bias of news source credibility ratings by large language models. In *Proceedings of the 17th ACM Web Science Conference 2025*, pages 127–137, 2025.

A Proofs

In this section, we assume that the items $(c_i, \mathbf{y}_i, x_i)$ are independent and identically distributed across i. We hold the number of raters K fixed and take limits as $I \to \infty$.

A.1 Proof of Lemma 1

Proof. We show that the joint distribution of the features and the votes factorizes given the true label.

$$\begin{aligned} P(x_i,\, \mathbf{y}_i \mid c_i) &= E\big[\, P(x_i \mid R(D_i),\, c_i,\, \mathbf{y}_i)\; P(\mathbf{y}_i \mid R(D_i),\, c_i) \;\big|\; c_i \,\big] \\ &= E\big[\, P(x_i \mid R(D_i))\; P(\mathbf{y}_i \mid R(D_i),\, c_i) \;\big|\; c_i \,\big] && \text{[By Assumption 3]} \\ &= P(\mathbf{y}_i \mid c_i)\; E\big[\, P(x_i \mid R(D_i)) \;\big|\; c_i \,\big] && \text{[By Assumption 4]} \\ &= P(\mathbf{y}_i \mid c_i)\; P(x_i \mid c_i) && \text{[By Assumption 3]} \end{aligned}$$

The first line conditions on $R(D_i)$ and applies the chain rule. The second line uses Assumption 3: the pair $(c_i, \mathbf{y}_i)$ is a function of $(D_i, \mathbf{y}_i)$, so x_i is independent of $(c_i, \mathbf{y}_i)$ given $R(D_i)$. The third line uses Assumption 4, which gives $P(\mathbf{y}_i \mid R(D_i),\, c_i) = P(\mathbf{y}_i \mid c_i)$. The fourth line uses Assumption 3 again, in the form $P(x_i \mid R(D_i)) = P(x_i \mid R(D_i),\, c_i)$, together with the law of total probability. □

A.2 The item-model regression

Lemma 2 (Item-model linear relation)**.** *Under Lemma 1, for any function g of the votes,*

$$E[g(\mathbf{y}_i) \mid \mathbf{x}_i] = \sum_q P(c_i = q \mid \mathbf{x}_i)\, E[g(\mathbf{y}_i) \mid c_i = q].$$

In the binary case this is

$$E[g(\mathbf{y}_i) \mid \mathbf{x}_i] = E[g(\mathbf{y}_i) \mid c_i = 0] + \big(E[g(\mathbf{y}_i) \mid c_i = 1] - E[g(\mathbf{y}_i) \mid c_i = 0]\big)\, e(\mathbf{x}_i).$$

Proof. We condition on the latent label and apply Lemma 1.

$$\begin{aligned} E[g(\mathbf{y}_i) \mid \mathbf{x}_i] &= \sum_q P(c_i = q \mid \mathbf{x}_i)\, E[g(\mathbf{y}_i) \mid \mathbf{x}_i,\, c_i = q] && \text{[Law of total expectation]} \\ &= \sum_q P(c_i = q \mid \mathbf{x}_i)\, E[g(\mathbf{y}_i) \mid c_i = q] && \text{[By Lemma 1]} \end{aligned}$$

In the binary case, $P(c_i = 1 \mid \mathbf{x}_i) = e(\mathbf{x}_i)$ and $P(c_i = 0 \mid \mathbf{x}_i) = 1 - e(\mathbf{x}_i)$, so:

$$\begin{aligned} E[g(\mathbf{y}_i) \mid \mathbf{x}_i] &= \big(1 - e(\mathbf{x}_i)\big)\, E[g(\mathbf{y}_i) \mid c_i = 0] + e(\mathbf{x}_i)\, E[g(\mathbf{y}_i) \mid c_i = 1] \\ &= E[g(\mathbf{y}_i) \mid c_i = 0] + \big(E[g(\mathbf{y}_i) \mid c_i = 1] - E[g(\mathbf{y}_i) \mid c_i = 0]\big)\, e(\mathbf{x}_i) \end{aligned}$$

The constant term is the intercept and the coefficient of $e(\mathbf{x}_i)$ is the slope. With $g(\mathbf{y}_i) = y_{ij}$, the intercept is $1 - \gamma_j$ and the slope is $\alpha_j + \gamma_j - 1$. □

A.3 When the rater-reliability path holds approximately

The oracle condition (Assumption 2(a)) requires that LOO majority vote is correct for every item. In practice it is not. When $\hat{c}_{i,-j} \neq c_i$ for some items, the Hajek estimator is biased. Proposition 1 gives the exact form of the bias, and Corollary 1 bounds it in terms of the error rate of the LOO-proxy. The results hold for any correlation matrix R.

Write FPR_j and FNR_j for the false positive rate and the false negative rate of the LOO-proxy for rater j:

$$\mathsf{FPR}_j = P(\hat{c}_{i,-j} = 1 \mid c_i = 0), \qquad \mathsf{FNR}_j = P(\hat{c}_{i,-j} = 0 \mid c_i = 1).$$

Proposition 1 (Bias of the leave-one-out estimator)**.** *As $I \to \infty$, $\hat{\alpha}_j \xrightarrow{p} \alpha_j + \mathsf{Bias}_j^{\alpha}$ and $\hat{\gamma}_j \xrightarrow{p} \gamma_j + \mathsf{Bias}_j^{\gamma}$, where*

$$\mathsf{Bias}_j^{\alpha} = \frac{-(1-\pi)\,\mathsf{FPR}_j\,(\alpha_j + \gamma_j - 1) \;+\; E\big[\mathrm{Cov}(\hat{c}_{i,-j},\, y_{ij} \mid c_i)\big]}{P(\hat{c}_{i,-j} = 1)},$$

$$\mathsf{Bias}_j^{\gamma} = \frac{-\pi\,\mathsf{FNR}_j\,(\alpha_j + \gamma_j - 1) \;+\; E\big[\mathrm{Cov}(\hat{c}_{i,-j},\, y_{ij} \mid c_i)\big]}{P(\hat{c}_{i,-j} = 0)}.$$

Proof. The items are independent and identically distributed, and $\hat{c}_{i,-j}$ is a function of the votes on item i. The law of large numbers applies to the numerator and to the denominator of the Hajek estimator:

$$\hat{\alpha}_j = \frac{I^{-1}\sum_i \hat{c}_{i,-j}\, y_{ij}}{I^{-1}\sum_i \hat{c}_{i,-j}} \;\xrightarrow{p}\; \frac{E[\hat{c}_{i,-j}\, y_{ij}]}{E[\hat{c}_{i,-j}]}.$$

The bias is the difference between this limit and α_j:

$$\begin{aligned} \mathsf{Bias}_j^{\alpha} &= \frac{E[\hat{c}_{i,-j}\, y_{ij}]}{E[\hat{c}_{i,-j}]} - \alpha_j \\ &= \frac{E[\hat{c}_{i,-j}\,(y_{ij} - \alpha_j)]}{E[\hat{c}_{i,-j}]} \\ &= \frac{E[(\hat{c}_{i,-j} - c_i)\,(y_{ij} - \alpha_j)]}{E[\hat{c}_{i,-j}]} \qquad [E[c_i\,(y_{ij} - \alpha_j)] = 0] \end{aligned}$$

The last line uses the definition of the sensitivity, $E[c_i\, y_{ij}] = \alpha_j\, E[c_i]$. We decompose the numerator by writing $y_{ij} - \alpha_j = \big(E[y_{ij} \mid c_i] - \alpha_j\big) + \big(y_{ij} - E[y_{ij} \mid c_i]\big)$:

$$\begin{aligned} E[(\hat{c}_{i,-j} - c_i)(y_{ij} - \alpha_j)] = {} & \underbrace{E\Big[(\hat{c}_{i,-j} - c_i)\cdot\big(E[y_{ij} \mid c_i] - \alpha_j\big)\Big]}_{\text{(A): false positive rate of the LOO-proxy}} \\ & + \underbrace{E\Big[(\hat{c}_{i,-j} - c_i)\cdot\big(y_{ij} - E[y_{ij} \mid c_i]\big)\Big]}_{\text{(B): within-class covariance}} \end{aligned}$$

For Term (A), the difference $E[y_{ij} \mid c_i] - \alpha_j$ equals zero when $c_i = 1$, and equals $-(\alpha_j + \gamma_j - 1)$ when $c_i = 0$. Therefore:

$$\begin{aligned} \text{(A)} &= -(\alpha_j + \gamma_j - 1)\cdot E\big[\hat{c}_{i,-j}\,(1 - c_i)\big] \\ &= -(1-\pi)\,\mathsf{FPR}_j\,(\alpha_j + \gamma_j - 1) \end{aligned}$$

For Term (B), condition on c_i. Given c_i, the label is a constant and the factor $y_{ij} - E[y_{ij} \mid c_i]$ has mean zero, so:

$$\text{(B)} = E\big[\mathrm{Cov}(\hat{c}_{i,-j},\ y_{ij} \mid c_i)\big]$$

The denominator is $E[\hat{c}_{i,-j}] = P(\hat{c}_{i,-j} = 1)$. The argument for $\hat{\gamma}_j$ replaces $\hat{c}_{i,-j}$, c_i and y_{ij} by $1 - \hat{c}_{i,-j}$, $1 - c_i$ and $1 - y_{ij}$, and the false negative rate of the LOO-proxy takes the place of the false positive rate. □

Term (B): within-class covariance. This covariance decomposes by definition into three factors:

$$\mathrm{Cov}(\hat{c}_{i,-j},\ y_{ij} \mid c_i) = \mathrm{Corr}(\hat{c}_{i,-j},\ y_{ij} \mid c_i) \cdot \sqrt{\mathrm{Var}(\hat{c}_{i,-j} \mid c_i)} \cdot \sqrt{\mathrm{Var}(y_{ij} \mid c_i)}$$

Within-class correlation between the proxy and the vote: $\mathrm{Corr}(\hat{c}_{i,-j}, y_{ij} \mid c_i)$. This is the dependence term. It is zero under conditional independence ($R = I$) and positive when R has positive off-diagonal entries.

Error in the estimated label: $\sqrt{\mathrm{Var}(\hat{c}_{i,-j} \mid c_i)}$. This is large when majority vote is uncertain—when the vote margin is narrow—and small when the majority is decisive.

Low LLM accuracy: $\sqrt{\mathrm{Var}(y_{ij} \mid c_i)}$. This is the variability in annotator j's vote within each class. It is maximised when the annotator is at chance and vanishes as the annotator becomes perfectly accurate.

Term (B) is large when all three conditions hold simultaneously: the annotators are correlated, the average predicted labels are uncertain, and the annotator is inaccurate. It is small when any one of these is small. The bias does not vanish with the number of items I; it is a property of the annotator pool.

Corollary 1 (Bounds on the bias)**.** *Let $\varepsilon_j = P(\hat{c}_{i,-j} \neq c_i)$ denote the error rate of the LOO-proxy for rater j, and suppose $\varepsilon_j < \min(\pi,\, 1 - \pi)$. Then*

$$\big|\mathsf{Bias}_j^{\alpha}\big| \le \frac{\varepsilon_j + \frac{1}{2}\sqrt{\varepsilon_j}}{\pi - \varepsilon_j}, \qquad \big|\mathsf{Bias}_j^{\gamma}\big| \le \frac{\varepsilon_j + \frac{1}{2}\sqrt{\varepsilon_j}}{1 - \pi - \varepsilon_j}.$$

Under Assumption 2(b), the terms $\frac{1}{2}\sqrt{\varepsilon_j}$ are absent, and $\varepsilon_j \le \exp(-2(K-1)\delta^2)$, where $\delta > 0$ is the smallest amount by which the averages in Assumption 2(b)(ii) exceed one half.

Proof. The error rate of the LOO-proxy is $\varepsilon_j = \pi\,\mathsf{FNR}_j + (1 - \pi)\,\mathsf{FPR}_j$. Each part of Bias_j^{α} is bounded in terms of ε_j:

$$\begin{aligned}
|\text{(A)}| &\le (1 - \pi)\,\mathsf{FPR}_j \le \varepsilon_j && [|\alpha_j + \gamma_j - 1| \le 1] \\
|\text{(B)}| &\le \tfrac{1}{2}\Big[\pi\sqrt{\mathsf{FNR}_j} + (1 - \pi)\sqrt{\mathsf{FPR}_j}\Big] && [\text{Cauchy–Schwarz, } \mathrm{Var}(y_{ij} \mid c_i) \le \tfrac{1}{4}] \\
&\le \tfrac{1}{2}\sqrt{\varepsilon_j} && [\text{Concavity of the square root}] \\
P(\hat{c}_{i,-j} = 1) &\ge \pi\,(1 - \mathsf{FNR}_j) \ge \pi - \varepsilon_j
\end{aligned}$$

Combining these bounds gives the bound on Bias_j^{α}, and the same argument gives the bound on Bias_j^{γ}.

Under Assumption 2(b), the LOO-proxy is a function of the votes of the other raters, and these votes are independent of y_{ij} given c_i. Term (B) is therefore zero. The error rates of the LOO-proxy are bounded as follows:

$$\begin{aligned}
\mathsf{FPR}_j &= P\Big(\tfrac{1}{K-1}\textstyle\sum_{k \neq j} y_{ik} > \tfrac{1}{2} \,\Big|\, c_i = 0\Big) \\
&\le P\Big(\tfrac{1}{K-1}\textstyle\sum_{k \neq j} \big(y_{ik} - (1 - \gamma_k)\big) \ge \delta \,\Big|\, c_i = 0\Big) && [\text{By Assumption 2(b)(ii)}] \\
&\le \exp\big(-2(K-1)\delta^2\big) && [\text{By Hoeffding's inequality}]
\end{aligned}$$

The second line uses $\frac{1}{K-1}\sum_{k \neq j}(1 - \gamma_k) \le \frac{1}{2} - \delta$. The same bound holds for FNR_j, so $\varepsilon_j \le \exp(-2(K-1)\delta^2)$. □

Under Assumption 2(a), $\varepsilon_j = 0$ and both biases are zero. Under Assumption 2(b), the bounds converge to zero as $K \to \infty$.

A.4 Formal statement and proof of Theorem 1

Theorem 1 (formal statement). *Suppose the items are independent and identically distributed, that π, α_j and γ_j lie in $(0,1)$ for every j, and that the pairwise agreement equation (2) holds with $\rho_{jj'} \in (-1,1)$ for every pair $j \neq j'$. Let $I \to \infty$ with K fixed.*

(i) Under Assumption 1, with $|\mathcal{S}| \to \infty$, for every j and every pair $j \neq j'$:

$$\hat{\pi} \xrightarrow{p} \pi, \quad \hat{\alpha}_j \xrightarrow{p} \alpha_j, \quad \hat{\gamma}_j \xrightarrow{p} \gamma_j, \quad \hat{\rho}_{jj'} \xrightarrow{p} \rho_{jj'}.$$

(ii) Under Assumption 2(a), the limits of part (i) hold. Under Assumption 2(b), $\hat{\alpha}_j \xrightarrow{p} \alpha_j + \mathsf{Bias}_j^{\alpha}$ and $\hat{\gamma}_j \xrightarrow{p} \gamma_j + \mathsf{Bias}_j^{\gamma}$, where

$$\big|\mathsf{Bias}_j^{\alpha}\big| \le \frac{\varepsilon_K}{\pi - \varepsilon_K}, \qquad \big|\mathsf{Bias}_j^{\gamma}\big| \le \frac{\varepsilon_K}{1 - \pi - \varepsilon_K}, \qquad \varepsilon_K = \exp\big(-2(K-1)\delta^2\big),$$

and δ is defined in Corollary 1. The bounds converge to zero as $K \to \infty$, and the limits of $\hat{\pi}$ and $\hat{\rho}_{jj'}$ converge to π and $\rho_{jj'}$.

(iii) Under Assumptions 3, 4, 5 and 6, the limits of part (i) hold.

When the limits of part (i) hold and R is positive definite, the classification probabilities, the flags, and the LLM biases of Section 3.3 converge in probability to their values at the true parameters.

The proof of Theorem 1 proceeds in three steps. First, we show convergence of the accuracy and prevalence terms under each of the identification paths (Lemma 3). Second, we show that the accuracy and prevalence terms are sufficient to identify the correlation terms (Lemma 4). Finally, we show that the functionals are consistent, given the accuracy, prevalence and correlation terms (Lemma 5).

Lemma 3 (Accuracies and prevalence)**.** *Under Assumption 1, under Assumption 2(a), or under Assumptions 3, 4, 5 and 6, for every j:*

$$\hat{\pi} \xrightarrow{p} \pi, \quad \hat{\alpha}_j \xrightarrow{p} \alpha_j, \quad \hat{\gamma}_j \xrightarrow{p} \gamma_j.$$

Under Assumption 2(b), $\hat{\alpha}_j \xrightarrow{p} \alpha_j + \mathsf{Bias}_j^{\alpha}$ and $\hat{\gamma}_j \xrightarrow{p} \gamma_j + \mathsf{Bias}_j^{\gamma}$, with the biases bounded as in Corollary 1, and the limit of $\hat{\pi}$ differs from π by at most ε_K.

Proof. **Identification with ground truth labels.** The estimator is the Hajek ratio over the labelled items. The subset $\mathcal{S}$ is selected independently of the labels and the votes, so the labelled items are an independent and identically distributed sample of the items. As $|\mathcal{S}| \to \infty$:

$$\hat{\alpha}_j = \frac{|\mathcal{S}|^{-1} \sum_{i \in \mathcal{S}} c_i\, y_{ij}}{|\mathcal{S}|^{-1} \sum_{i \in \mathcal{S}} c_i} \xrightarrow{p} \frac{E[c_i\, y_{ij}]}{E[c_i]} \qquad \text{[Law of large numbers]}$$
$$= \frac{\pi\, E[y_{ij} \mid c_i = 1]}{\pi} \;=\; \alpha_j$$

The argument for $\hat{\gamma}_j$ replaces c_i and y_{ij} by $1 - c_i$ and $1 - y_{ij}$. The prevalence estimator is the mean of the observed labels, and it converges to π by the same argument.

Identification under the LOO-proxy. Under Assumption 2(a), $\hat{c}_{i,-j} = c_i$ for every item. The false positive rate and the false negative rate of the LOO-proxy are zero, and the within-class covariance is zero because the proxy is a constant given c_i. Proposition 1 then gives $\mathsf{Bias}_j^{\alpha} = \mathsf{Bias}_j^{\gamma} = 0$. The full majority vote equals c_i whenever every leave-one-out majority vote does, so $\hat{\pi}$ is the sample mean of c_i and converges to π. Under Assumption 2(b), Proposition 1 gives the limits and Corollary 1 gives the bounds. The full majority vote is wrong with probability at most ε_K by the same application of Hoeffding's inequality, so the limit of $\hat{\pi}$ differs from π by at most ε_K.

Identification under a valid item model. Lemma 1 gives $\mathbf{y}_i \perp\!\!\!\perp \mathbf{x}_i \mid c_i$. Apply Lemma 2 with $g(\mathbf{y}_i) = y_{ij}$:

$$E[y_{ij} \mid \mathbf{x}_i] = (1 - \gamma_j) + (\alpha_j + \gamma_j - 1)\, e(\mathbf{x}_i).$$

The conditional mean is linear in the score, so the population least-squares regression of y_{ij} on $e(\mathbf{x}_i)$ recovers its coefficients:

$$\begin{aligned}
\text{slope} &= \frac{\operatorname{Cov}(y_{ij},\, e(\mathbf{x}_i))}{\operatorname{Var}(e(\mathbf{x}_i))} = \alpha_j + \gamma_j - 1 && \text{[By Assumption 6]} \\
\text{intercept} &= E[y_{ij}] - (\alpha_j + \gamma_j - 1)\, E[e(\mathbf{x}_i)] \\
&= \pi\, \alpha_j + (1-\pi)(1-\gamma_j) - (\alpha_j + \gamma_j - 1)\, \pi && [E[e(\mathbf{x}_i)] = \pi] \\
&= 1 - \gamma_j
\end{aligned}$$

Therefore γ_j equals one minus the intercept and α_j equals the slope plus the intercept. The sample regression uses $\hat{e}(\mathbf{x}_i)$ in place of $e(\mathbf{x}_i)$. The votes and the scores lie in $[0, 1]$, so each sample moment in the least-squares formula changes by at most $2\, I^{-1} \sum_i |\hat{e}(\mathbf{x}_i) - e(\mathbf{x}_i)|$ when $\hat{e}$ replaces e. This quantity converges to zero by Assumption 5. The sample coefficients therefore converge to the population coefficients by the law of large numbers, so $\hat{\alpha}_j \xrightarrow{p} \alpha_j$ and $\hat{\gamma}_j \xrightarrow{p} \gamma_j$. The same bound gives $\hat{\pi} = I^{-1} \sum_i \hat{e}(\mathbf{x}_i) \xrightarrow{p} E[e(\mathbf{x}_i)] = \pi$.

(The accuracies also determine the prevalence, through $P(y_{ij} = 1) = \pi\, \alpha_j + (1-\pi)(1-\gamma_j)$.) □

Lemma 4 (Correlations)**.** *Suppose the pairwise agreement equation* (2) *holds with $\rho_{jj'} \in (-1, 1)$, and that the estimators $\hat{\pi}$, $\hat{\alpha}_j$, $\hat{\alpha}_{j'}$, $\hat{\gamma}_j$ and $\hat{\gamma}_{j'}$ converge in probability to π, α_j, $\alpha_{j'}$, γ_j and $\gamma_{j'}$. Then $\hat{\rho}_{jj'} \xrightarrow{p} \rho_{jj'}$.*

Proof. Write $\hat{h}_{jj'}$ for the function of Section 3.2, which uses the plug-in values of the prevalence and the accuracies, and write $h_{jj'}$ for the same function at the true values. The function $h_{jj'}$ is continuous in ρ, in the prevalence, and in the accuracies. It is strictly increasing in ρ, because the derivative of $\Phi_2(a, b;\, \rho)$ with respect to ρ is the bivariate normal density at (a, b), which is positive. The inverse $h_{jj'}^{-1}$ is therefore continuous in its argument, in the prevalence, and in the accuracies.

$$\begin{aligned}
S_{jj'} &\xrightarrow{p} P(y_{ij} = 1,\ y_{ij'} = 1) && \text{[Law of large numbers]} \\
&= h_{jj'}(\rho_{jj'}) && \text{[By Equation (2)]} \\
\hat{\rho}_{jj'} = \hat{h}_{jj'}^{-1}(S_{jj'}) &\xrightarrow{p} h_{jj'}^{-1}\big(h_{jj'}(\rho_{jj'})\big) = \rho_{jj'} && \text{[Continuity of } h_{jj'}^{-1}\text{]}
\end{aligned}$$

When the estimators of the prevalence and the accuracies converge to biased limits, the same argument shows that $\hat{\rho}_{jj'}$ converges to the inverse of $h_{jj'}$ evaluated at those limits. By the continuity of the inverse, this limit converges to $\rho_{jj'}$ as the biases converge to zero. □

Lemma 5 (Model outputs)**.** *Suppose $\hat{\pi}$, $\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\gamma}}$ and $\hat{R}$ converge in probability to π, $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$ and R, and that R is positive definite. Then the classification probabilities and the LLM biases of Section 3.3 converge in probability to their values at the true parameters. The flag of an item converges when its limiting standard deviation differs from the threshold τ.*

Proof. For a fixed vote pattern $\mathbf{y} \in \{0, 1\}^K$, the classification probability is a continuous function of π, $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$ and R when R is positive definite. There are 2^K vote patterns, so the classification probabilities converge at every pattern by the continuous mapping theorem. The LLM biases and the classification standard deviations are continuous functions of the classification probabilities. □

Proof of Theorem 1. The theorem follows from Lemma 3, Lemma 4 and Lemma 5. Under Assumption 2(b), Lemma 3 gives the biased limits, Corollary 1 shows that the bounds converge to zero as $K \to \infty$, and the final part of Lemma 4 gives the limit of $\hat{\rho}_{jj'}$. □

B The item model

B.1 Fitting the score with a two-class mixture model

In this section, we describe one model for the score $e(x)$ of Section 2.3.

We suppose that the probability of observing specific features varies in each class. Define the class-conditional feature distributions:

$$f_1(x) = P(x_i = x \mid c_i = 1), \qquad f_0(x) = P(x_i = x \mid c_i = 0),$$

where, by hypothesis, $f_1 \neq f_0$, and define the base rate, or prevalence, $\pi = P(c_i = 1)$.

We observe the features, and their frequencies, but not the class labels. The distribution of features in the population is a mixture of the two class-conditional distributions. By the law of total probability, we have:

$$P(x_i = x) = \pi\, f_1(x) + (1 - \pi)\, f_0(x).$$

We do not observe π, f_1 or f_0, so the unsupervised learning task is to recover these parameters from the observed feature frequencies alone.

With these parameters in hand, the score follows by Bayes' rule:

$$e(x) = \frac{\pi\, f_1(x)}{\pi\, f_1(x) + (1 - \pi)\, f_0(x)}.$$

This is the conditional probability of class one given the features.

Whether the mixture can be recovered depends on the form the analyst specifies for f_1 and f_0. The number of components is fixed by the class, and the decomposition above is the law of total probability, so it holds for any f_1 and f_0. The analyst can make different assumptions on the structure of f_1 and f_0. For instance, if f_1 and f_0 are products of independent Bernoullis:

$$f_d(x) = \prod_{m=1}^{M} p_{dm}^{x_m}\, (1 - p_{dm})^{1 - x_m},$$

the items are conditionally independent given the class, which is the Dawid-Skene model for raters, but applied to items. [17] propose a random effects model, in which raters are conditionally independent given a continuous trait whose distribution varies by class: this could be applied to items. A multivariate probit form places a correlation matrix on latent feature scores, which is the item model version of the correlated Dawid Skene proposal of Section 3.

In addition, the mixture model recovers two components but does not say which is the positive class, which is the problem of label-switching. The analyst resolves this by declaring the orientation: in the manifesto case, the component in which "public ownership" occurs at a higher rate is the left. To resolve this, we require the following assumption.

Assumption 7 (Label switching)**.** *The feature named in the orientation declaration is produced at a higher rate by the positive class: $p_{1m^*} > p_{0m^*}$ for the assigned-positive feature m^*.*

In the manifesto case, this states that truly left manifestos mention public ownership at a higher rate than truly right manifestos. This assumption resolves the label-switching indeterminacy that is present in any two-component mixture model.

Given a correctly specified and identified model for f_1 and f_0, and a resolution to the label-switching problem, the fitted score converges to the true conditional probability, which is the requirement of Assumption 5. The orientation fixed by Assumption 7 makes $\hat{e}$ consistent for $e(\mathbf{x}_i) = P(c_i = 1 \mid \mathbf{x}_i)$ rather than $1 - e(\mathbf{x}_i)$, without which the slope and intercept of the regression in Section 3.2 identify α_j and γ_j with their roles exchanged.

C Ordinal data

The application in Section 5 uses votes on an ordinal scale. In this section, we state the ordinal version of the model, show that the binary model of Section 3.1 is the case with two categories, and describe how the estimators and the identification results carry over. The correlation matrix R is the same object in both models.

The ordinal model. The true label takes one of J ordered values, $c_i \in \{1, \ldots, J\}$, with prevalences $\pi_q = P(c_i = q)$. Each rater reports one of the same J values, $y_{ij} \in \{1, \ldots, J\}$. As in the binary model, each item has a vector of latent utilities, with one entry for each rater:

$$z_i \mid c_i = q \;\sim\; \mathcal{N}(\mu_q,\, R)$$

Here $\mu_q = (\mu_{q,1}, \ldots, \mu_{q,K})$ is the mean vector of class q. Each rater j has ordered cut-points $-\infty = \tau_{j,0} < \tau_{j,1} < \cdots < \tau_{j,J-1} < \tau_{j,J} = +\infty$, and reports the category whose interval contains the latent utility:

$$y_{ij} = l \iff \tau_{j,l-1} < z_{ij} \leq \tau_{j,l}$$

The binary model has a single threshold at zero for every rater. The ordinal model replaces this threshold with $J-1$ cut-points for each rater. We fix $\tau_{j,1} = 0$, because a common shift of the cut-points and the means of rater j leaves the votes unchanged.

The sensitivity and the specificity generalize to the confusion matrix of rater j, which records the probability of each vote within each class:

$$\begin{aligned}
\beta_{q,l}^{(j)} &= P(y_{ij} = l \mid c_i = q) \\
&= P(\tau_{j,l-1} < z_{ij} \leq \tau_{j,l} \mid c_i = q) && \text{[Vote rule]} \\
&= \Phi(\tau_{j,l} - \mu_{q,j}) - \Phi(\tau_{j,l-1} - \mu_{q,j}) && [z_{ij} \mid c_i = q \sim \mathcal{N}(\mu_{q,j}, 1)]
\end{aligned}$$

The binary model as a special case. Take $J = 2$, so that each rater has the single cut-point $\tau_{j,1} = 0$. Identify class 2 with $c_i = 1$, class 1 with $c_i = 0$, and the vote $y_{ij} = 2$ with a vote of one. The vote rule becomes $y_{ij} = 2 \iff z_{ij} > 0$, which is the vote rule of Section 3.1. The diagonal of the confusion matrix contains the sensitivity and the specificity:

$$\begin{aligned}
\beta_{2,2}^{(j)} &= 1 - \Phi(-\mu_{2,j}) = \Phi(\mu_{2,j}) = \alpha_j \\
\beta_{1,1}^{(j)} &= \Phi(-\mu_{1,j}) = \gamma_j
\end{aligned}$$

Therefore $\mu_{2,j} = \Phi^{-1}(\alpha_j)$ and $\mu_{1,j} = \Phi^{-1}(1 - \gamma_j)$, which are the class means of Section 3.1. The binary model is the ordinal model with two categories.

Estimation. Each estimator of Section 3.2 has an ordinal form, and each ordinal form returns the binary estimator at $J = 2$.

The LOO-proxy $\hat{c}_{i,-j}$ is the median of the votes of the other $K-1$ raters, with a fixed rule when the median falls between two categories. At $J = 2$ this is the leave-one-out majority vote. The Hajek estimator applies to each cell of the confusion matrix:

$$\hat{\beta}_{q,l}^{(j)} = \frac{\sum_i \mathbb{I}\{\hat{c}_{i,-j} = q\}\, \mathbb{I}\{y_{ij} = l\}}{\sum_i \mathbb{I}\{\hat{c}_{i,-j} = q\}}$$

At $J = 2$, the cells $(2,2)$ and $(1,1)$ are $\hat{\alpha}_j$ and $\hat{\gamma}_j$. The cut-points and the class means are then fitted to $\hat{\beta}^{(j)}$ through the expression for the confusion matrix above. At $J = 2$ this step is $\hat{\mu}_{2,j} = \Phi^{-1}(\hat{\alpha}_j)$ and $\hat{\mu}_{1,j} = \Phi^{-1}(1 - \hat{\gamma}_j)$.

For the correlations, the pairwise agreement equation (2) becomes one equation for each pair of categories (l, m):

$$P(y_{ij} = l,\ y_{ij'} = m) = \sum_{q=1}^{J} \pi_q\, P\big(\tau_{j,l-1} < z_{ij} \leq \tau_{j,l},\ \ \tau_{j',m-1} < z_{ij'} \leq \tau_{j',m} \,\big|\, c_i = q\big)$$

Each term on the right-hand side is a rectangle probability of a bivariate normal distribution with correlation $\rho_{jj'}$. The estimate $\hat{\rho}_{jj'}$ is the value at which these probabilities fit the observed joint frequencies of the two raters' votes, which is the polychoric correlation [16]. At $J = 2$, the pair $(l, m) = (2, 2)$ gives Equation (2).

The classification probability follows from Bayes' rule, as in Section 3.1:

$$P(c_i = q \mid \mathbf{y}_i) = \frac{\pi_q\, P(\mathbf{y}_i \mid c_i = q)}{\sum_{r=1}^{J} \pi_r\, P(\mathbf{y}_i \mid c_i = r)}$$

Here $P(\mathbf{y}_i \mid c_i = q)$ is the probability that every latent utility lies in the interval of the reported category. The binary model integrates the same normal density over half-lines, so the implementation changes in the limits of integration alone.

Identification. Theorem 1 carries over path by path. With ground truth labels, the confusion matrices and the prevalences are class-conditional frequencies on the labelled subset. Under the LOO-proxy, the oracle condition $\hat{c}_{i,-j} = c_i$ makes the Hajek estimator of each cell a class-conditional frequency, as in Lemma 3. Under conditional independence, the requirement that the average accuracy exceeds chance becomes the requirement that, within each class, the median vote of the other raters is the true label. Proposition 1 holds for each cell with $\mathbb{I}\{\hat{c}_{i,-j} = q\}$ in place of $\hat{c}_{i,-j}$, and the bias of each cell is the sum of a misclassification term and a within-class covariance term.

Under a valid item model, the score becomes the vector of class probabilities $e_q(x_i) = P(c_i = q \mid x_i)$. Lemma 2 gives a linear relation for each category of vote:

$$\begin{aligned} E\big[\mathbb{I}\{y_{ij} = l\} \mid x_i\big] &= \sum_{q=1}^{J} P(c_i = q \mid x_i)\; E\big[\mathbb{I}\{y_{ij} = l\} \mid c_i = q\big] && \text{[By Lemma 2]} \\ &= \sum_{q=1}^{J} e_q(x_i)\; \beta_{q,l}^{(j)} \end{aligned}$$

The regression of the vote indicator on the scores recovers column l of the confusion matrix. The regression has no intercept, because the scores sum to one. It requires that the scores are linearly independent across items, which is the ordinal form of Assumption 6. At $J = 2$ this is the regression of Section 3.2.

Given consistent estimates of the confusion matrices and the prevalences, the remaining steps follow the binary case. The argument of Lemma 4 applies to each moment $P(y_{ij} \geq l,\ y_{ij'} \geq m)$, which is a mixture of bivariate normal orthant probabilities and is increasing in $\rho_{jj'}$. The argument of Lemma 5 applies with rectangle probabilities in place of orthant probabilities.

D Estimating $\hat{\rho}_{jj'}$ by bisection

We compute $\hat{\rho}_{jj'} = h_{jj'}^{-1}(S_{jj'})$ by bisection as follows. Fix the pair (j, j') and write $h = h_{jj'}$.

1. Set $\rho_L = -1 + \epsilon$ and $\rho_U = 1 - \epsilon$ for some small $\epsilon > 0$ (we use $\epsilon = 0.001$).
2. Compute the midpoint $\rho_M = (\rho_L + \rho_U)/2$.
3. Evaluate $h(\rho_M)$. This requires two bivariate normal CDF evaluations, available via `pmvnorm` in `R` and `scipy.stats.multivariate_normal` in Python.
4. If $|h(\rho_M) - S_{jj'}| < \delta$ for a convergence tolerance δ (we use $\delta = 10^{-10}$), stop and return $\hat{\rho}_{jj'} = \rho_M$.
5. If $h(\rho_M) < S_{jj'}$, set $\rho_L = \rho_M$. Otherwise, set $\rho_U = \rho_M$.
6. Return to step 2.

Each iteration halves the interval $[\rho_L, \rho_U]$. After n iterations, the interval has width $\dfrac{2 - 2\epsilon}{2^n}$.