Even Faster Ideal Point Estimation

Using Variational Autoencoders to Estimate Item-Response Theory Models

Mason Reece*

December 9, 2025

Abstract

Researchers with discrete outcome data are often interested in estimating *latent* traits characterizing their data. One popular method is Item-Response Theory (IRT). Unfortunately, standard methods of estimating IRT models are ill-equipped for large, modern data or complex models. I introduce a new method to political scientists – developed in computer vision – the Variational Autoencoder (VAE). A particularly structured VAE reliably estimates latent traits and item parameters for millions of respondents with multiple dimensions orders of magnitude faster than traditional MCMC methods. They can also easily be extended to handle missing data, correlated latent traits, and scale easily across all types of problems. I also suggest methods for correcting VAE estimates to match standard methods in finite samples. I demonstrate the method with several empirical applications using the complete ballots of 40 million voters in the 2020 US election.

The author thanks Devin Caughey, Charles Stewart III, Andrea Campbell, Naoki Egami, Shuning Ge, Yuyang Shi, Gabrielle Péloquin-Skulski and participants in MIT's Computational Social Science Workshop for helpful comments.

*PhD candidate, Department of Political Science, MIT. mpreece@mit.edu, https://mreece13.github.io/

1 Introduction

Researchers with discrete data (roll call votes, social media posts, positions taken on bills, etc.) are often interested in mapping those data to a continuous, latent, quantity of interest (ideology, democratic norms, etc.). For example, scholars studying ideological positions of actors began with the spatial voting model developed in Poole and Rosenthal (1985) and have since expanded to a large number of substantive applications using a wide variety of data (e.g., Martin and Quinn 2002; Shor and McCarty 2011; Barberá 2015; Bond and Messing 2015; Lauderdale and Herzog 2016; Burnham 2024; Cowburn and Sältzer 2025; Lewis et al. 2025; Meisels 2025). The size and complexity of this data ranges from analyses of a few thousand votes in the United Nations (Bailey, Strezhnev, and Voeten 2017) to millions of social media followers (Barberá 2015).[1] Standard approaches to estimating IRT models include maximum-likelihood estimation, Hamiltonian Monte-Carlo (HMC), Expectation-Maximization (EM) algorithms (Chalmers 2012; Imai, Lo, and Olmsted 2016; Peress 2022) or variational inference (VI) (Grimmer 2011; Imai, Lo, and Olmsted 2016). Wu et al. (2020) compares the computational time required to fit a model using each algorithm and finds that even the fastest methods take over 6 hours to fit a model with 1.6 million respondents, 100 items, and a single latent dimension. This time cost is perfectly acceptable for researchers who have similarly sized data. However, some modern datasets contain more data or wish to estimate more complex models, and will be prohibitively slow. For example, information on social media users, campaign contributions, or digital trace datasets are essential to modern political science research and are still too large to estimate IRT models on.

In this article I introduce a new technique to political science for accurately and quickly estimating ideal points at scale using *variational autoencoders* (VAEs). VAEs are closely connected to item-response theory and have been shown in the psychometric literature to accurately estimate both ideal points and other parameters of an ideal point model (Converse et al. 2021; Curi et al. 2019; Ma et al. 2024; Cho et al. 2021). The size of the speedup can vary based on the problem, but in the empirical example I explore, VAEs are appropriately accurate and about 5000 times as fast as traditional MCMC algorithms. This

1. Although note that although Barberá (2015) have data on millions of social media followers, they are unable to estimate latent dimensions for all users and instead collapse their model down to just a few thousand estimates.

kind of speedup is backed up by other simulation research in the computer science literature (Wu et al. 2020). Similar to the recommendations of other authors who develop computational improvements to the foundational ideal point method, I caution that this method is only an approximation of the most complete Bayesian inferential approach. However, as I argue below, it is both acceptably accurate in large samples and fast enough to justify as an alternative to other approaches. The use of VAEs dramatically expands the types of data that can be used and the questions that can be asked using latent dimension techniques. Moreover, VAEs are easily deployed using modern machine learning frameworks that include many internal optimizations and can run on large scale computing clusters designed for machine learning, a common resource available to researchers through cheap cloud computing. I also provide code examples and practical tips and diagnostics for researchers looking to implement a VAE.

2 Item-Response Theory

First, a brief overview of the traditional methods to estimate ideal points following Item-Response Theory in the context of political science. IRT models in political science are closely connected to the spatial voting model,[2] where a voter is assumed to choose an option if the utility a voter derives from that option is greater than any other option (Enelow and Hinich 1984). As a running example, related to the empirical application in section 4, consider data on the candidate a given voter voted for in each of the contests on their ballot. Jackman (2009) shows that beginning with the spatial voting model plus the assumption that any stochastic error about this utility is normally distributed results in the following model for the probability that a candidate is selected by a voter:

$$\pi_{jkc} = Pr(y_{jk} = c|\alpha_j, \gamma_{kc}, \beta_{kc}) = F(\alpha_j\gamma_{kc} - \beta_{kc}), \tag{1}$$

where j indexes voters, k indexes the contest, and c indexes the candidate in that contest.[3] This

2. See Conevska and Mutlu (2025) for a critical evaluation of whether spatial voting models at the individual level are the best model for evaluating voters. Given the historical importance of spatial voting and the novelty of the finding in Conevska and Mutlu (2025), I focus here only on spatial voting.

3. Although I don't show it here for clarity, the model can also be multidimensional. The additional dimension is

model is the canonical two-parameter IRT model. There are three categories of parameters of interest. First, $\alpha_j \in \mathbb{R}$ is the unobserved ideal point of voter j. Second, γ_{kc} is an unknown *item discrimination* or slope parameter for each candidate who could be chosen in the contest. γ_{kc} describes the degree to which a vote for a candidate in the contest informs the model's placement of a voter's ideal point. β_{kc} is an *item difficulty* parameter which measures the probability of a certain candidate being chosen, irrespective of the underlying trait. In this context, this is the proportion of the votes that each candidate received. Note that some elements of γ_{kc} must be set to 0 to avoid rotational indeterminancy. In the categorical case, where it may be hard *a priori* to know the direction of effects, this is usually done randomly. Note that this issue of rotational indeterminacy is similar to typical identification issues in ideal point models (Rivers 2003). $F(\cdot)$ is a monotone function mapping the equation to the probability line. The simple IRT model typically chooses the inverse logit function for $F(\cdot)$, but for a more general treatment I rely on the softmax function, a multivariate generalization of the inverse logit.[4] This choice reflects the reality that voters often have more than two candidates to choose among on the ballot.

Combining the choice probabilities into a full likelihood yields the following model:

$$L(\mathbf{\Omega}, \alpha_j | \mathbf{y}_j) = \prod_{k=1}^{K} \pi_{jky_{jk}} = \prod_{k=1}^{K} F(\alpha_j \gamma_{ky_{jk}} - \beta_{ky_{jk}}), \tag{2}$$

where $\mathbf{\Omega}$ includes both the discrimination and difficulty parameters and the notation $[\cdot]_{y_{jk}}$ is used to indicate which candidate voter j actually chose in contest k. In the Bayesian sense, we can then compute the posterior: $p(\boldsymbol{\alpha}, \mathbf{\Omega} | \mathbf{Y}) \propto L(\mathbf{\Omega}, \boldsymbol{\alpha} | \mathbf{Y}) \cdot p(\boldsymbol{\alpha}) \cdot p(\mathbf{\Omega})$, where $p(\boldsymbol{\alpha})$ and $p(\mathbf{\Omega})$ are priors set by the researcher.

There are a number of algorithms to estimate all of the parameters of interest, including Hamil-

reflected in the ideal point α_{jd} and in the discrimination parameter, γ_{kcd} where d indexes the latent dimension. The difficulty parameter, β_{kc}, is the same for all dimensions.

4. Formally, the softmax is defined as

$$F(\alpha_j \gamma_{kc} - \beta_{kc}) = \frac{\exp(\alpha_j \gamma_{kc} - \beta_{kc})}{\sum_{c'=1}^{C_k} \exp(\alpha_j \gamma_{kc'} - \beta_{kc'})},$$

where C_k is the is the number of candidates in contest k.

tonian Monte Carlo (HMC) as implemented in Stan, expectation maximization (EM) algorithms for certain types of data (Chalmers 2012; Imai, Lo, and Olmsted 2016; Peress 2022), and variational inference (VI) methods (Grimmer 2011; Imai, Lo, and Olmsted 2016). HMC and EM algorithms work by directly seeking to optimize the marginal log-likelihood $p(\boldsymbol{y}|\boldsymbol{\alpha})$, whereas VI instead learns a different function $q(\boldsymbol{\alpha})$ for the posterior distribution. VI has theoretical guarantees because it seeks to maximize the evidence lower bound (ELBO), formally defined as:

$$\text{ELBO} = \mathbb{E}_{q(\boldsymbol{\alpha})}[\log p(\boldsymbol{Y}|\boldsymbol{\alpha}; \boldsymbol{\Omega})] - \text{KL}[q(\boldsymbol{\alpha}) \,||\, p(\boldsymbol{\alpha})], \tag{3}$$

where $\text{KL}(\cdot)$ is the Kullback–Leibler divergence between two distributions. By applying Bayes Rule to Equation 3, we can see that

$$\log p(\boldsymbol{Y}) = \text{ELBO} + \text{KL}[q(\boldsymbol{\alpha}) \,||\, p(\boldsymbol{\alpha}|\boldsymbol{Y})]. \tag{4}$$

Therefore, as the function $q(\boldsymbol{\alpha})$ more closely resembles $p(\boldsymbol{\alpha}|\boldsymbol{Y})$, we return to directly optimizing the marginal likelihood. By selecting a simpler $q(\boldsymbol{\alpha})$, VI makes estimation a more tractable problem. We then estimate $\hat{\boldsymbol{\alpha}} = \text{argmax(ELBO)}$ to obtain the latent points for each voter. Although I only reference the ideal point $\boldsymbol{\alpha}$ here, we can also fit approximation functions for each of the other parameters in the model.

Unfortunately, HMC, EM, and VI are all still insufficient for large data sets. To see why, consider the data set of cast vote records released in Kuriwaki et al. (2024). Cast vote records are digital representations of a ballot cast by a voter and they contain the candidate each voter voted for in each contest on the ballot. They match perfectly with the setup at the start of this section. The data set includes the ballots from 40 million voters and over 7000 unique contests. Estimating an IRT model using this data combined with any of the three methods outlined above would involve estimating a posterior, or an approximation of a posterior, for every single voter. Furthermore, for each additional contest, we need to also calculate a discrimination and a difficulty parameter. As the number of parameters increases, the estimation of each individual parameter becomes more challenging since we

are estimating a joint distribution. Finally, if researchers are interested in estimating additional latent dimensions, the problem increases by a factor of the number of additional dimensions. Although technically possible, the computation of this model can be time and resource intensive. For medium sized data, some researchers may be happy to simply wait the computational time required. But, for large data where the time required is too much, or in cases where researchers may want to iterate, the time cost can be prohibitive. In the next section I introduce a new, faster, method – variational autoencoders.

3 Variational Autoencoders

Variational autoencoders (VAEs) are closely connected to variational inference. The difference is that instead of choosing a distribution $q(\boldsymbol{\alpha})$ to approximate each posterior, we instead build a single flexible inferential model that predicts the posteriors for each ideal point. The parameters of this model are shared by all observations in the data, making the size and complexity of the model or the number of parameters to estimate simply a function of researcher choice instead of scaling with the number of observations. By carefully constructing the model such that it is both interpretable and sufficiently flexible, we can recover all three types of parameters of interest from the standard IRT model. It turns out that a good choice for a VAE.

In general, VAEs are a semi-supervised machine learning method. Intuitively, an autoencoder *encodes* the input (the choices of each voter in each contest) to a latent dimension(s) using a learned neural network, then *decodes* this latent representation to reconstruct the original input using a second, connected, learned neural network. Autoencoders have been in use in the machine learning literature for over a decade. Originally developed for image recognition, their use has since expanded into other areas (Kingma and Welling 2019). We make the autoencoder "variational" by using it to approximate the true posterior.

Formally, assume the posterior distribution of the latent variables comes from a multivariate normal distribution:

$$\boldsymbol{\alpha_j} \sim \text{MVN}(\boldsymbol{\mu_j}, \text{diag}(\boldsymbol{\sigma_j})), \text{ with } (\boldsymbol{\mu_j}, \boldsymbol{\sigma_j}) = \text{NeuralNet}_{\phi}(\boldsymbol{y_j}), \tag{5}$$

where $\boldsymbol{\mu_j}$ and $\boldsymbol{\sigma_j}$ characterize the mean and standard deviation of the multivariate normal (MVN), and ϕ contains all the parameters of the neural network model. Together, these constitute the entire "encoder" part of the VAE. To optimize the new parameters we now maximize a slight variation of the ELBO shown in Equation 3:

$$\text{ELBO} = \mathbb{E}_{q_\phi(\boldsymbol{\alpha})}[\log p(\boldsymbol{Y}|\boldsymbol{\alpha}; \boldsymbol{\Omega})] - \text{KL}[q_\phi(\boldsymbol{\alpha}) \,||\, p(\boldsymbol{\alpha})], \tag{6}$$

where instead we are maximizing parameters of the neural network instead of directly maximizing voter posteriors.

A VAE model is trained by minimizing the loss between the original data and the learned data construction, where the parameters of both the decoder and encoder networks are adjusted using standard methods stochastic gradient descent and backpropagation. I reparameterize the model to ensure backpropagation can properly occur because taking gradient of the ELBO with respect to the parameters ϕ is not possible (Wu et al. 2020; Urban and Bauer 2021). To see this, note that

$$\begin{aligned} \nabla_\phi \text{ELBO} &= \nabla_\phi \mathbb{E}_{q_\phi(\boldsymbol{\alpha})}[\log p(\boldsymbol{\alpha}) - \log q_\phi(\boldsymbol{\alpha})] \\ &\neq \mathbb{E}_{q_\phi(\boldsymbol{\alpha})}[\nabla_\phi \log p(\boldsymbol{\alpha}) - \nabla_\phi \log q_\phi(\boldsymbol{\alpha})], \end{aligned} \tag{7}$$

because the expectations are taken with respect to $q_\phi(\boldsymbol{\alpha})$, which is a function of ϕ. We resolve this problem by reparameterizing $\boldsymbol{\alpha}$ as

$$\begin{aligned} \boldsymbol{\varepsilon} &\sim \text{MVN}(\boldsymbol{\varepsilon}) \\ \boldsymbol{\alpha} &= \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\varepsilon}, \end{aligned} \tag{8}$$

where $\boldsymbol{\varepsilon}$ is a $D \times 1$ sample from a standard multivariate normal distribution and $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are

outputs from the NeuralNet above. This removes any of the "randomness" in $\boldsymbol{\alpha}$ by rewriting it as a deterministic function of the NeuralNet parameters ϕ. Therefore, we can now calculate the second line in Equation 7 where the expectation is now taken with respect to MVN($\boldsymbol{\varepsilon}$). These models can easily be implemented in PyTorch, Keras, or any other standard machine learning library.

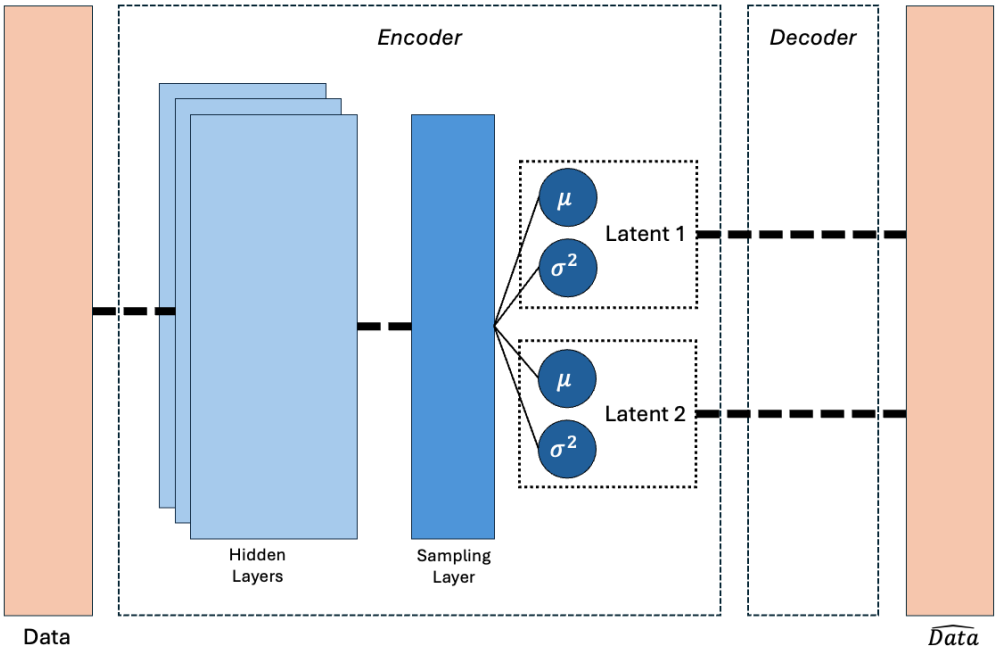

Figure 1: Architecture of a 2-Dimension VAE. Depicts the architecture of a VAE with two latent dimensions. Orange boxes represent both inputted data (leftmost box) and the learned representation of that data (rightmost box). From left to right, the light blue boxes represent a user-specified number of hidden layers of a user-specified size, then a user-specified sampling layer, then the output of the encoder network in the form of two parameters, μ and σ^2 that characterize the distribution of the latent normal distribution for each of the two latent parameters. For each latent dimension, the set of two parameters are then fed through the decoder network. Heavy dashed lines represent a densely connected network, where every node or neuron on one side of the line is connected to every other node on the other side. Solid lines simply represent a connection between one node and another. Light dashed lines outline the components of the architecture that are referred to, respectively, as the *encoder* and *decoder*.

Figure 1 visually represents a basic VAE with two latent dimensions. Beginning from the left side of the diagram, data are fed as inputs to the encoder model, which is composed of some number of hidden layers before passing through a sampling layer. The sampling layer produces the parameters μ and σ^2 when a normal distribution is imposed. Those latent dimensions are then passed through the

decoder network to reconstruct the original data.

There are a couple points to make about the connection between the VAE and IRT model. First, VAEs add an assumption that the latent dimension(s) follow some probability distribution, in many cases, a multivariate normal distribution. Notice the similarity between this assumption and the assumption in a classical spatial voting model that errors are distributed normally. This assumption is encoded as a "sampling layer" at the end of the encoding model which then outputs parameters describing the distribution of the model (in the standard normal case, a parameter μ and σ). The second modification is that the decoder network has only a single layer that passes the learned latent variables through the the exact same $F(\cdot)$ function as in Equation 1. For example, if the $F(\cdot)$ function is a sigmoidal activation function, then $\sigma(z_i) = (1 + e^{-z_i})^{-1}$ where $z_i = \sum_{k=1}^{K} w_{ki}\gamma_k + \beta_i$. Here, w_{ki} is the weight between the k-th and i-th nodes in the second-to-last and output layer, respectively. γ_k is the activation of the k-th node in the second-to-last layer, and β_i is the bias value of the i-th node in the output layer (c.f. Converse et al. 2021). Notice the intentional use of the same parameters γ and β as in the IRT setup – these weights and biases of the VAE can be interpreted in the same manner as the discrimination and difficulty parameters in the IRT model (Curi et al. 2019).

Model Architecture Of course, a key component of a VAE is how to construct the model architecture. Unfortunately, there are not absolute recommendations, but I'll provide some suggestions. For the hidden layers in the encoder, I generally recommend multiple hidden layers and to choose a size for each hidden layer as a multiple of 2 (in the models I fit below, I choose 64). The size and number of these hidden layers is a balancing act of computational time and GPU memory – larger layers with more hidden layers tend to be more "flexible" models but are also more challenging to fit. The number of latent dimensions should be theoretically driven depending on researcher discretion. In machine learning it is also common to batch responses together, a process called "mini-batching." Again, larger batches are generally "better" but more computationally challenging to evaluate. Here, I use batches of size 256. I set a standard learning rate of 0.001.

3.1 Extensions

The literature on VAEs in general is rich and contains many optimizations to the basic model presented here or specific adaptations for other circumstances. Here, I overview two relevant extensions for political scientists: missing data and correlated latent dimensions.

Missing Data Missing data are common in political science applications, but unfortunately due to its neural network structure are a challenging application for a VAE. The problem lies in the step within a neural network where gradients are computed with respect to each of the network parameters. Where parameters are missing, the gradient cannot be computed. Similar to other missing data work, listwise deletion or otherwise just dropping missing data is not advised for statistical efficiency in general. In the running example of voter level ideal points based on their votes on their ballot, deletion is impossible – across the full set of voters only the presidential contest is shared by all voters. All contests that a voter could not have cast a ballot in are labeled "missing." Previous work on voter level ideal points has obviated this problem by constructing the data such that all voters have the same choice set (e.g., Lewis 2001). Models to handle this kind of data are only rarely addressed in political science, with one notable exception constructing a different kind of multinomial logit estimator for the varying choices (Yamamoto 2014).

There are several solutions to missing data proposed in the VAE literature. Veldkamp, Grasman, and Molenaar (2025) review the options and conduct simulation of each method. Of the methods they tested, the "conditional VAE" had the lowest error compared to the true values and was also one of the fastest methods. Conditional VAEs are intuitively simple. Instead of dropping missing data or imputing missingness, we can simply pass a mask of 1s and 0s to the VAE as an additional input (Collier, Nazabal, and Williams 2020). The mask is used in two steps – first in the initial hidden layers to zero out any inputs that are missing, and then again in the loss function to ensure that loss is only computed for values that are actually present in the data.

Correlated Latent Dimensions The current implementation of the VAE forces each of the latent dimensions to be independent. In cases where this might be an unreasonable assumption, researchers could instead allow for, and directly model, the correlation between latent dimensions. For example, in studies of American politics it might be reasonable to allow the first dimension, typically assumed to be ideology, to also correlate with additional dimensions – e.g., economic values or social policy. From a modeling perspective, the change is subtle but impactful. Instead of training our model to fit the latent space to the standard normal distribution, $\mathcal{N}(0, I)$, researchers can instead fit it to $\mathcal{N}(0, \Sigma_0)$ where Σ_0 is a $D \times D$ matrix with entries for each latent dimension (Converse et al. 2021). The estimation of the VAE does not substantively change, except that we need to additionally pass a correlation matrix to the model, typically denoted as Σ_1, to prevent identification/rotation issues.

4 Empirical Application

4.1 Code and Implementation Details

I estimate these models using a bespoke model, fit on ballot level data from the United States in the 2020 election.[5] More details on the data can be found in Appendix A. The model is fit with a standard categorical cross-entropy loss function, 20 epochs, mini-batching, and an AMSGrad optimizer. I implement early stopping on the training loss for additional efficiency. As a rough comparison in computation time, it took the MCMC algorithm 3 hours and 45 minutes to fit a single-dimensional model on a subset of the data using part of Adams County, Colorado, including 250,000 votes.[6] A two-dimensional model does not complete warmup in 48 hours on the same dataset. In contrast, it took the VAE 90 seconds (1/150th of the time) to fit a single-dimensional model using all 6.4 million votes in Adams County, Colorado. To be clear, despite the model using the same architecture on the full data, the actual evaluation of the close to 1.3 billion votes in my full data precludes the model from running in the same exact time as the smaller subset. However, the run time increases much

5. The code can be found at https://github.com/mreece13/cvr-ml/blob/main/main_lightning.py.

6. Note that this model is already highly optimized and was run on a high-powered GPU using variational inference as initialization values.

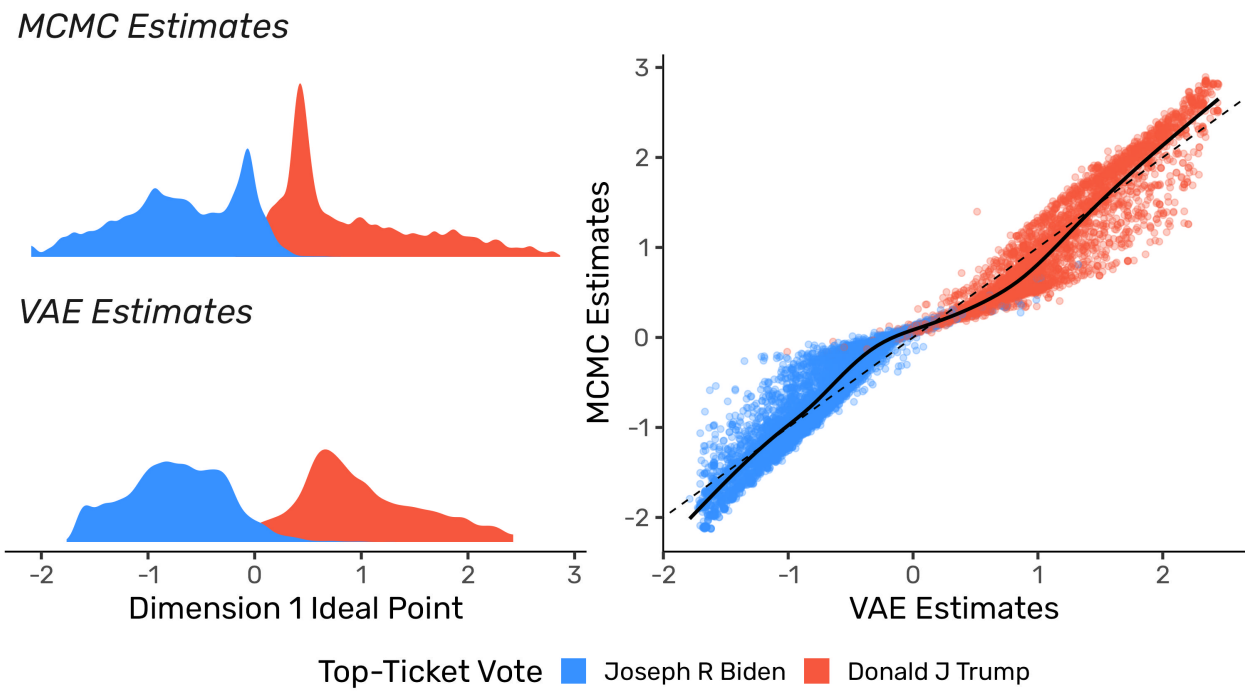

Figure 2: Comparison of MCMC and VAE Methods. The left plots display the the densities of the mean ideal point for voters who selected Joseph R Biden or Donald J Trump in the presidential contest. The right plot directly plots each matched voter's mean ideal point from the VAE algorithm and the MCMC algorithm. The solid black line is a loess line fit to the data and the dashed black line is the 45 degree line.

more slowly than the number of observations increases. For example, fitting a VAE on the full state of Colorado (26 million votes) only took 4 minutes to reach optimal convergence whereas the MCMC algorithm cannot finish warmup in 48 hours.

4.2 Validation

My initial visual inspection of the model is positive. I fit a single dimensional model on a subset of the data using MCMC and VAE and then plot the ideal points from both models in the left side of Figure 2.[7] The plot displays the distribution of ideal points among Biden (Democratic) and Trump (Republican) voters in the 2020 election. Both the MCMC and VAE plots are clearly bimodal and place

7. The MCMC model can only be run on a subset of voters from Adams County, Colorado, a suburb of Denver. The VAE model is fit on the entire state of Colorado, then subset down to just the same subset of voters in Adams County for comparison.

Biden (Trump) voters on the left (right) side of the axis. This matches the findings from other research in this area.[8] In the MCMC models, the two clear peaks around 0 and 0.7 are primarily straight-ticket voters for Democratic and Republican candidates respectively. For those voters, it can be hard to ascertain exactly where they are in the space relative to other strong party voters, a common criticism of ideal point models. In the right plot of Figure 2, I directly plot the estimates from each model against one another. Again, there is clear separation between the two parties, and the estimated ideal points correlate at 0.97. As can be seen in the plot, the models fit very similarly to one another.

Comparison of candidates across contests also support the model's success. For example, Diana de Gette, a House of Representative candidate from Denver, is the candidate with the most similar features to Joe Biden. Diana de Gette voted with Joe Biden's position 100% of the time in the 117th Congress, served as a deputy whip in the House, and is roughly in the middle of the DW-NOMINATE specification from the 117th Congress (*Does Your Member Of Congress Vote With Or Against Biden?* 2021; Lewis et al. 2025). These results suggest that the model is correctly grouping both voters and candidates.

Correcting Bias If researchers are unconvinced by the method, they could also use the following procedure to "de-bias" the VAE estimates: (1) estimate an MCMC model on a small random subset of the data; (2) estimate a VAE model on the full data; (3) using the overlapping points, build a predictive model that maps the VAE estimates to the MCMC estimates; (4) using that predictive model, correct the remaining VAE estimates. Randomly selecting data to be included in the random subset is challenging, but I suggest using a method that well-estimates specific contests using MCMC. For example, researchers could randomly select precincts, which all vote on the same contests, then select some (or all) voters within that precinct. The choice of prediction model is also important, but likely follows from standard recommendations for predictive models – choose a flexible model and rely on cross-fitting to avoid overfitting. Armed with the validation of VAE models and correction from a predictive model, researchers should feel comfortable to proceed with VAE estimates as if they were

8. For example, Bafumi and Herron (2010) plot the distribution of voter ideal points measured using the 2006 CCES in Figure 5, where we can clearly see a roughly bimodal, although largely inseparable distribution.

MCMC estimates.

4.3 Example Substantive Uses

These data can be used for interesting substantive evaluations in political science. For example, in Figure 3 I show a comparison of the underlying "ideology" of each of Colorado's state house districts. In the top plot are estimates of the mean ideal point (or, using the terminology of spatial voting scholars, the pivotal voter) in the district, including an inset of the Denver metro area to see the districts more clearly. In the bottom plot is the same area but instead 'ideology' is computed as the difference in vote share between the Republican and Democrat in the contest, using precinct level results (Baltz et al. 2022). Setting aside uncontested elections and missing counties (represented with grey districts), there are some interesting differences in the plots. For example, despite electing a Democrat in a southwestern district (District 62) of Colorado, the underlying ideal point model suggests that the district trends more Republican in general. Donald Valdez was the Democratic candidate in District 62, who received 57.8% of the vote compared with the 50.5% Joe Biden received from the same voters. This could suggest that Valdez is a popular candidate among both parties who gets votes from people who are otherwise Republican voters. It's hard to directly make comparisons between the two methods since they are inherently on different scales, but this kind of analysis is suggestive of the greater power and usefulness of the model. This analysis also serves as an additional validation of the model using real vote shares instead of just comparing to the MCMC method.

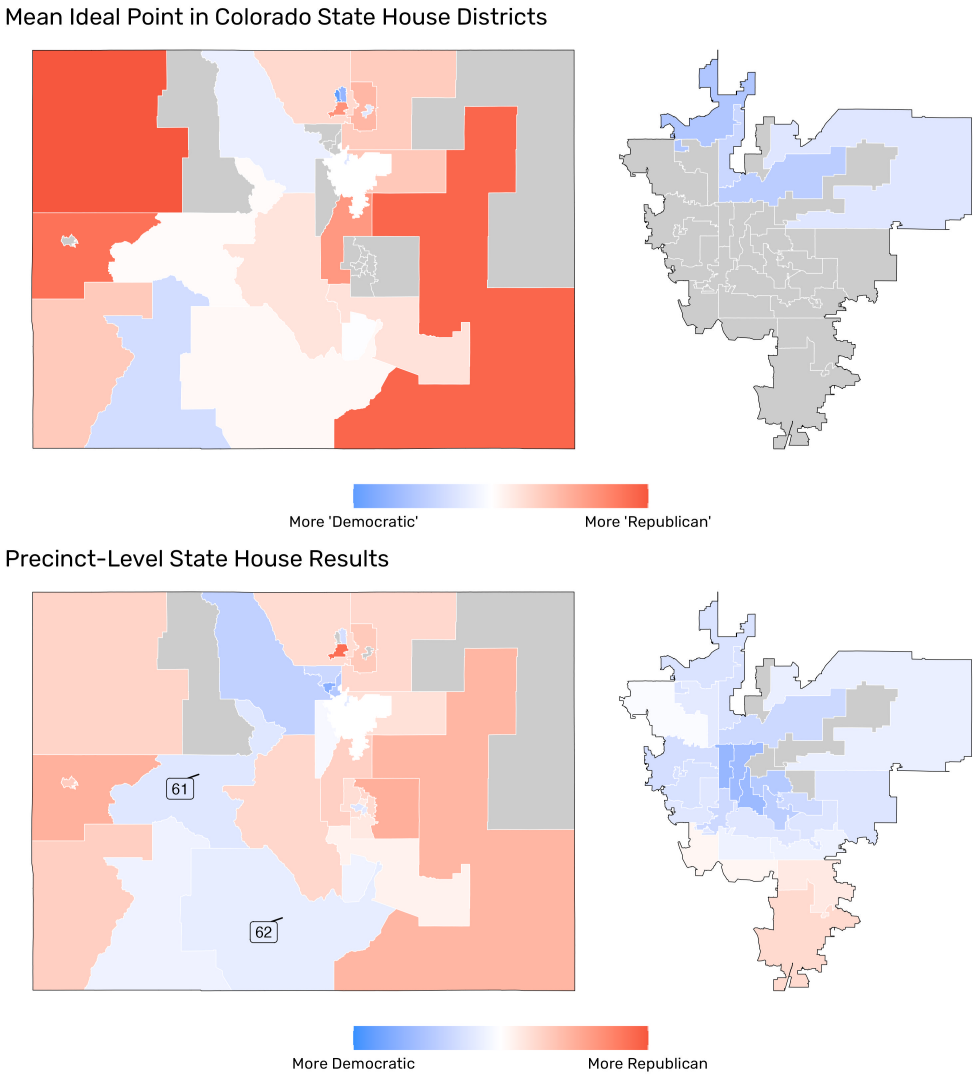

Figure 3: Comparison of Vote Share in State House Elections and Underlying Ideal Point Estimates. The top plot displays the mean ideal point among voters in each state house district in Colorado. Districts that are more blue lean more towards the Democrats, whereas districts that are more red lean towards the Republicans. On the left side is a map of the entire state, sans Denver, whereas on the right side is a zoomed in figure of the greater Denver area. This structure and label is then repeated in the bottom plots, which use precinct-level results in state house elections (Baltz et al. 2022). In the bottom plot, I have labeled two state house districts where the methods report different underlying 'ideologies' of the district. In the bottom plots, gray districts represent uncontested elections where a vote share breakdown does not make sense. In the top plots, gray districts can represent either uncontested elections (the exact same elections as in the bottom plots) or districts where entire counties are missing from the CVR data. A clear example of the missing counties can be seen in the Denver inset in the top plots, where all of the districts in the bottom right are missing – these are part of Arapahoe County, a county which is missing valid data in the CVR dataset. See Kuriwaki et al. (2024) for more information on what defines "valid" data in this context.

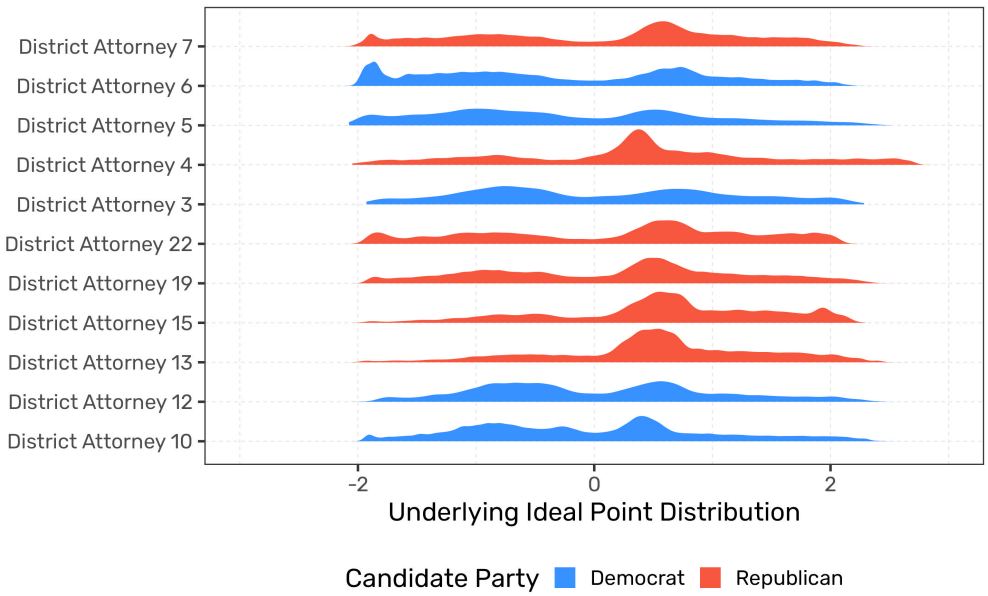

Figure 4: Underlying Latent Ideology of Uncontested Elections. The plot displays, for each uncontested District Attorney contest in Colorado, the distribution of each ideal point for voters who could've cast a ballot in the contest. The distributions are colored by the party of the candidate who won.

Equipped with a measure of the underlying "ideology" of a district, I can also evaluate contests with only one candidate. Using precinct-level data would be uninformative here, since all candidates receive (notwithstanding write-ins), 100% of the vote. In Figure 4, I plot the underlying ideal point distribution for all uncontested District Attorney contests in Colorado. Although mostly the plot supports the hypothesis of some scholars that elections are uncontested when the district is highly skewed towards one party (Hall 2019), there are also examples of districts that appear to be competitive. District 10, at the bottom of the plot, elected a Democrat but has a clear spike on the right side of the distribution as well. I'm not sure that a Republican would've necessarily won the election, but a contest would've likely at least been competitive.

Alternatively, we might be interested in analysis beyond the first latent dimension. For example, are ballot questions ideological in nature or do they load on different dimensions? Without estimates of multidimensional latent models which also include top of the ticket, and clearly ideological, models, it is hard to quantitatively link these concepts together and test propositions for ideological voting. In Figure 5, I display the point estimates for the discrimination parameters for the "yes" votes on

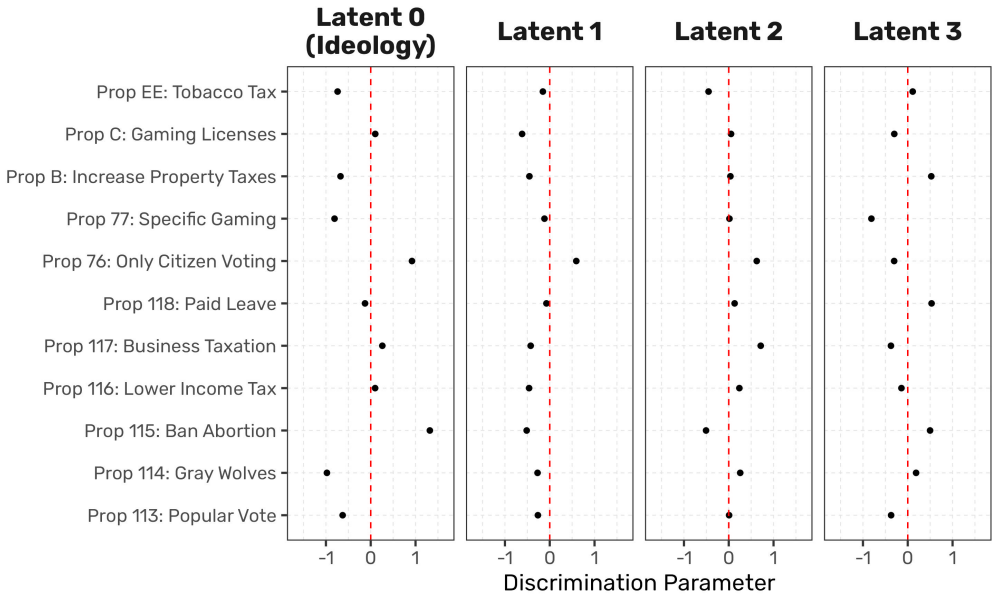

Figure 5: Distribution of γ Among Statewide Propositions Using Four Latent Dimensions. Displays the estimated discrimination parameter, γ, for a "yes" vote on each of the statewide propositions in Colorado, when the model is estimated using four latent dimensions. The first column, latent dimension 0, is colloquially referred to as ideology, both because it clearly separates the presidential candidates Donald J. Trump (Republican) and Joseph R. Biden (Democrat), and because of the historical reasoning to refer to the first dimension as ideology. Given this interpretation, points that are to the left of the red dashed zero line indicate that a "yes" vote on this proposition is the more "liberal" or "Democrat" vote. Latent dimensions 1-3 do not have a common interpretation.

every statewide proposition in the 2020 Colorado general election. The model was fit with four latent dimensions. In latent dimension 0, which I have termed ideology, we can see clearly that some of the propositions are highly informative about a voter's placement on the ideological scale. These include contests that we might intuitively expect to be on this scale – banning abortion and ensuring only citizens vote. However, the model also reveals unexpected ideological patterns on the re-introduction of gray wolves to the wild in Colorado. Latent dimensions 1-3 are more challenging to interpret and require further analysis to make substantive claims, but researchers who might be more informed on the minutiae of ballot propositions could have more to say.

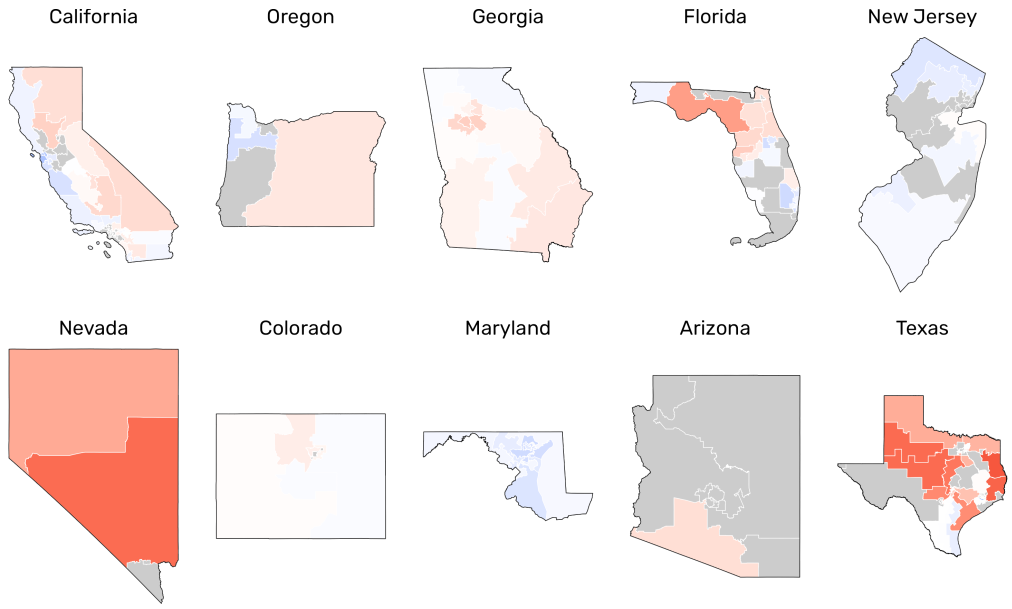

Figure 6: Underlying Ideal Point Estimates in US House Districts. The plot displays the mean ideal point among voters in each House district in the US. Districts that are more blue lean more towards the Democrats, whereas districts that are more red lean towards the Republicans. Gray districts can represent either uncontested elections or districts where entire counties are missing from the CVR data.

Finally, the speed of estimation allows me to fit an IRT model to votes from the entire nation and jointly scale all voters. The bridging contest in this case is only the US President, but initial results suggest this is sufficient. Using a random sample of precincts from every county in the country (~44 million votes), I fit a one-dimensional IRT model, then plot the underlying ideal point distribution for every US House district in Figure 6. Facially, the results seem to line up with our expectation, where

more rural districts are more Republican and conservative than urban districts.

5 Conclusion

This paper has introduced and validated a new method for estimating item-response theory (IRT) models – the variational autoencoder (VAE). VAEs are fast and accurate and open up the possibility for researchers to use new big datasets in combination with a popular method for scaling. I show using three empirical examples of how VAEs can be used to reveal relationships that were previously challenging to uncover using traditional methods. The literature on VAEs in computer science is vast, and many developments on the base model could continue to refine and develop this method. Future research should explore these possibilities.

References

Bafumi, Joseph, and Michael C. Herron. 2010. "Leapfrog Representation and Extremism: A Study of American Voters and Their Members in Congress." *American Political Science Review* 104, no. 3 (August): 519–542.

Bailey, Michael A., Anton Strezhnev, and Erik Voeten. 2017. "Estimating Dynamic State Preferences from United Nations Voting Data." *Journal of Conflict Resolution* 61, no. 2 (February): 430–456.

Baltz, Samuel, Alexander Agadjanian, Declan Chin, John Curiel, Kevin DeLuca, James Dunham, Jennifer Miranda, Connor Halloran Phillips, Annabel Uhlman, and Cameron Wimpy. 2022. "American election results at the precinct level." *Scientific Data* 9 (1): 651.

Barberá, Pablo. 2015. "Birds of the same feather tweet together: Bayesian ideal point estimation using Twitter data." *Political analysis* 23 (1): 76–91.

Bond, Robert, and Solomon Messing. 2015. "Quantifying social media's political space: Estimating ideology from publicly revealed preferences on Facebook." *American Political Science Review* 109 (1): 62–78.

Burnham, Michael. 2024. *Semantic Scaling: Bayesian Ideal Point Estimates with Large Language Models,* May.

Chalmers, R. Philip. 2012. "mirt: A multidimensional item response theory package for the R environment." *Journal of statistical Software* 48:1–29.

Cho, April E., Chun Wang, Xue Zhang, and Gongjun Xu. 2021. "Gaussian variational estimation for multidimensional item response theory." *British Journal of Mathematical and Statistical Psychology* 74, no. S1 (July): 52–85.

Collier, Mark, Alfredo Nazabal, and Christopher K. I. Williams. 2020. *VAEs in the Presence of Missing Data,* June.

Conevska, Aleksandra, and Can Mutlu. 2025. *When Do Voters Stop Caring? Estimating the Shape of Voter Utility Function,* February.

Converse, Geoffrey, Mariana Curi, Suely Oliveira, and Jonathan Templin. 2021. "Estimation of multidimensional item response theory models with correlated latent variables using variational autoencoders." *Machine Learning* 110, no. 6 (June): 1463–1480.

Cowburn, Mike, and Marius Sältzer. 2025. "Partisan communication in two-stage elections: the effect of primaries on intra-campaign positional shifts in congressional elections." *Political Science Research and Methods* 13 (2): 392–411.

Curi, Mariana, Geoffrey A. Converse, Jeff Hajewski, and Suely Oliveira. 2019. "Interpretable Variational Autoencoders for Cognitive Models." In *2019 International Joint Conference on Neural Networks (IJCNN),* 1–8. July.

Enelow, James M., and Melvin J. Hinich. 1984. *The spatial theory of voting: An introduction.* CUP Archive.

Does Your Member Of Congress Vote With Or Against Biden? | FiveThirtyEight. 2021, April.

Grimmer, Justin. 2011. "An Introduction to Bayesian Inference via Variational Approximations." *Political Analysis* 19 (1): 32–47.

Hall, Andrew B. 2019. *Who Wants to Run?: How the Devaluing of Political Office Drives Polarization.* University of Chicago Press, March.

Imai, Kosuke, James Lo, and Jonathan Olmsted. 2016. "Fast Estimation of Ideal Points with Massive Data." *American Political Science Review* 110, no. 4 (November): 631–656.

Jackman, Simon. 2009. *Bayesian Analysis for the Social Sciences.* Hoboken, NJ: Wiley.

Kingma, Diederik P., and Max Welling. 2019. "An Introduction to Variational Autoencoders." *Foundations and Trends® in Machine Learning* 12 (4): 307–392.

Kuriwaki, Shiro, Mason Reece, Samuel Baltz, Aleksandra Conevska, Joseph R. Loffredo, Can Mutlu, Taran Samarth, et al. 2024. "Cast vote records: A database of ballots from the 2020 U.S. Election." *Scientific Data* 11, no. 1 (November 28, 2024): 1304.

Lauderdale, Benjamin E., and Alexander Herzog. 2016. "Measuring political positions from legislative speech." *Political Analysis* 24 (3): 374–394.

Lewis, Jeffrey B, Keith Poole, Howard Rosenthal, Adam Boche, Aaron Rudkin, and Luke Sonnet. 2025. *Voteview: Congressional Roll-Call Votes Database.*

Lewis, Jeffrey B. 2001. "Estimating Voter Preference Distributions from Individual-Level Voting Data." *Political Analysis* 9, no. 3 (January): 275–297.

Ma, Chenchen, Jing Ouyang, Chun Wang, and Gongjun Xu. 2024. "A note on improving variational estimation for multidimensional item response theory." *psychometrika* 89 (1): 172–204.

Martin, Andrew D., and Kevin M. Quinn. 2002. "Dynamic ideal point estimation via Markov chain Monte Carlo for the US Supreme Court, 1953–1999." *Political analysis* 10 (2): 134–153.

Meisels, Mellissa. 2025. "Candidate Positions, Responsiveness, and Returns to Extremism." *The Journal of Politics* (December): 000–000.

Peress, Michael. 2022. "Large-Scale Ideal Point Estimation." *Political Analysis* 30, no. 3 (July): 346–363.

Poole, Keith T., and Howard Rosenthal. 1985. "A spatial model for legislative roll call analysis." *American journal of political science,* 357–384.

Rivers, Douglas. 2003. *Identification of Multidimensional Item Response Models.*

Shor, Boris, and Nolan McCarty. 2011. "The ideological mapping of American legislatures." *American Political Science Review* 105 (3): 530–551.

Urban, Christopher J., and Daniel J. Bauer. 2021. "A deep learning algorithm for high-dimensional exploratory item factor analysis." *Psychometrika* 86 (1): 1–29.

Veldkamp, Karel, Raoul Grasman, and Dylan Molenaar. 2025. "Handling missing data in variational autoencoder based item response theory." *British Journal of Mathematical and Statistical Psychology* 78 (1): 378–397.

Wu, Mike, Richard L. Davis, Benjamin W. Domingue, Chris Piech, and Noah Goodman. 2020. *Variational Item Response Theory: Fast, Accurate, and Expressive,* March.

Yamamoto, Teppei. 2014. "A Multinomial Response Model for Varying Choice Sets, with Application to Partially Contested Multiparty Elections" (April).

Appendix

A Cast Vote Record Data

I use ballot-level data, commonly called "cast vote records" (CVRs), from the 2020 general election. CVRs are anonymized records of all the choices made by each voter on their ballot. The data is fully anonymous and I am unable to identify anything about the voters except where they voted. The data has been standardized into a common format across the nation to facilitate analysis.[1]

CVRs from the 2020 election were released by some local election administrators in response to increasing calls from election activists to be able to independently verify the results of the 2020 election.[2] I do not know why each county in my data released their data so I treat my data as a non-random sample of CVRs. Nevertheless, it contains over 600 million choices in elections at all levels of government and in localities of all kinds from all over the country, representing the votes of about 28 million voters. I first remove all candidates who received less than 50 votes in the data, then remove all races where only one candidate was contesting the election, and finally remove all races where a voter could select more than one candidate.[3] The full set of CVRs covers 20 states, 327 counties, 28 million voters and 7,000 unique offices. See Figure A.1 for the full distribution of the data. This data allow me to simultaneously study local-level heterogeneity and make comparisons between more jurisdictions.

1. More details on this process can be found in Kuriwaki et al. (2024)

2. Some states prohibit releasing CVRs in any form, and for other states the ability to do so depended on whether a group requested them from a certain county, whether the county used technology enabling this data to be easily compiled, and whether the county election office had the capacity to even complete the request.

3. Future research should model these types of races since they represent about 10% of all elections in the data

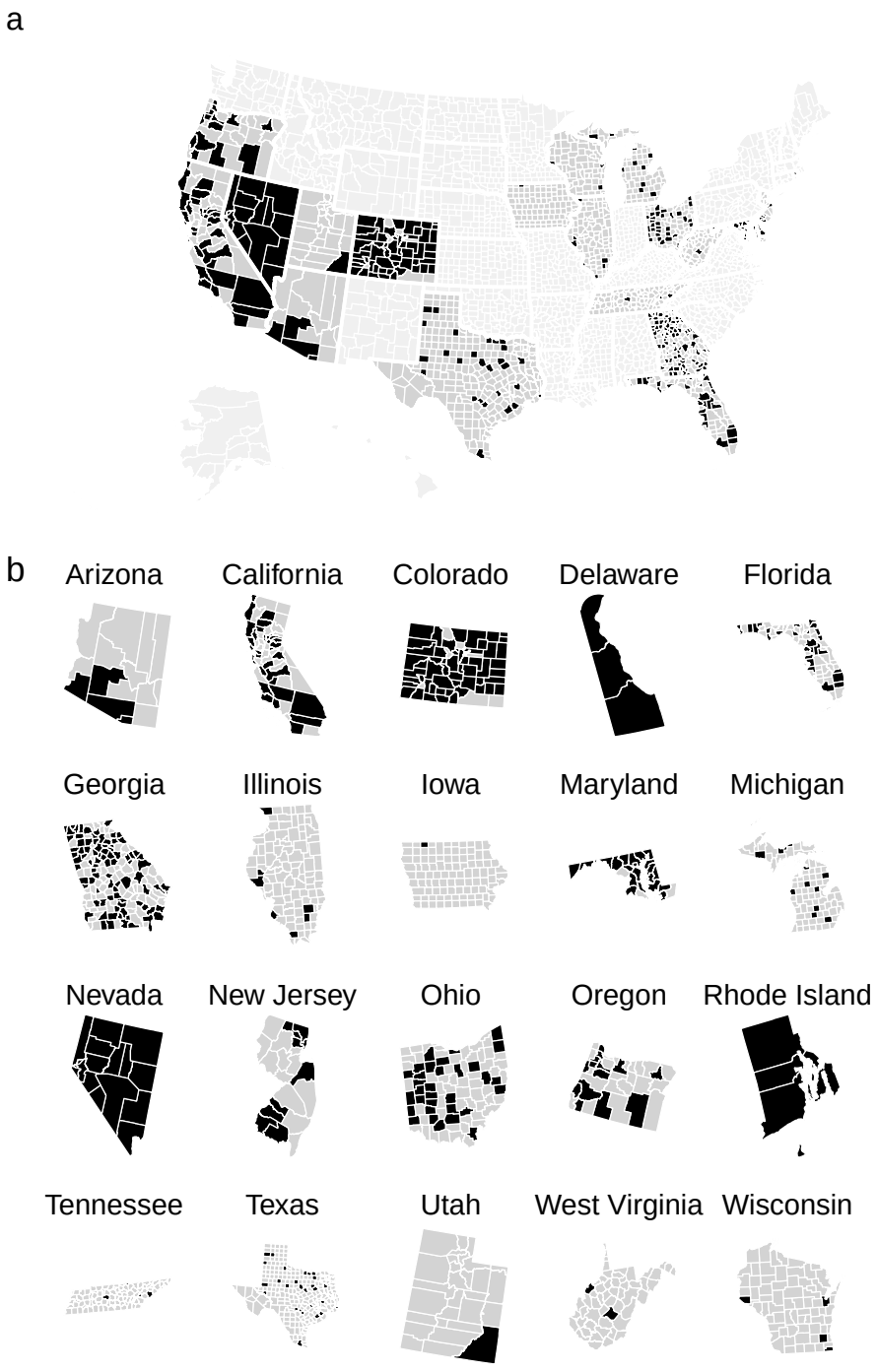

Figure A.1: Cast Vote Record Distribution. Figure reproduced from Kuriwaki et al. (2024). This is a map of the United States subdivided by state and county. Panel A shows the full map of the United States, whereas Panel B zooms in on only the states where I have coverage.