Reducing the dimensionality and granularity in hierarchical categorical variables

Paul Wilsens[1], Katrien Antonio[1,2] & Gerda Claeskens[3]
[1]LRisk - KU Leuven
[2]RCLR - University of Amsterdam
[3]ORSTAT - KU Leuven

Insurance Data Science - June 18, 2024

Presenter + Authors

Paul Wilsens

Katrien Antonio

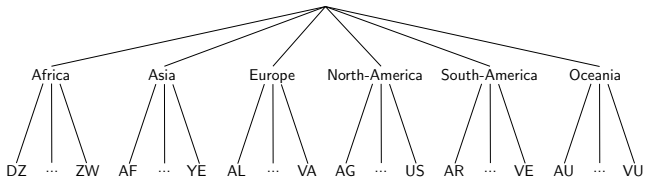
Gerda Claeskens

ArXiv (preprint): https://arxiv.org/abs/2403.03613

GitHub (R code): https://github.com/PaulWilsens/reducing-hierarchical-cat

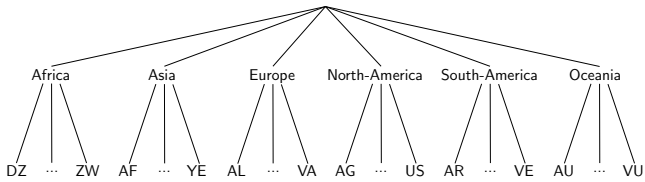
Introduction

Handling categorical variables in predictive modelling can be challenging.



Handling categorical variables in predictive modelling can be challenging.

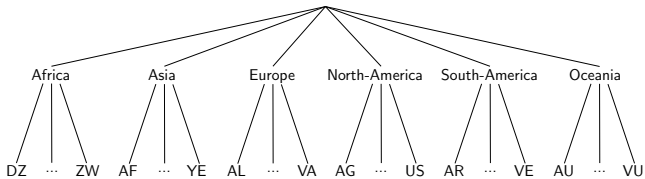
They require numerical encodings to be included in a model, e.g., dummy variables or one-hot encoding.



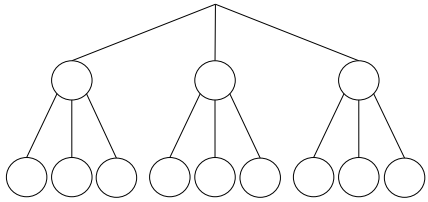
Handling categorical variables in predictive modelling can be challenging.

They require numerical encodings to be included in a model, e.g., dummy variables or one-hot encoding.

Categorical variables can have an inherent hierarchical structure.

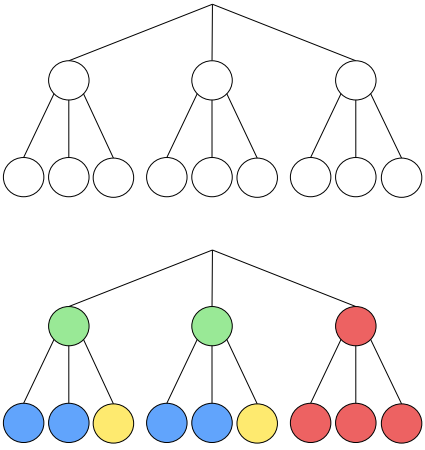


Hierarchical categorical variables often exhibit high dimensionality and high granularity, leading to overfitting and estimation issues.



Hierarchical categorical variables often exhibit high dimensionality and high granularity, leading to overfitting and estimation issues.

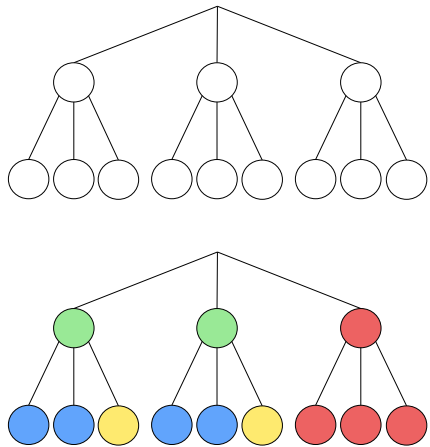
Commonly, random effects are utilised, see e.g. random effects entity embedding [Richman and Wüthrich 2024].



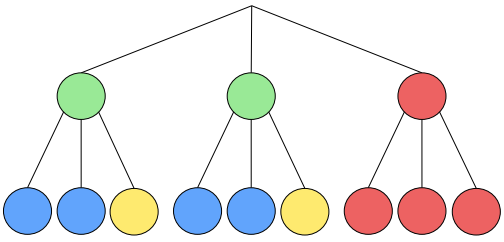
Hierarchical categorical variables often exhibit high dimensionality and high granularity, leading to overfitting and estimation issues.

Commonly, random effects are utilised, see e.g. random effects entity embedding [Richman and Wüthrich 2024].

By construction, random effects do not allow classes having the same effect on the response variable.

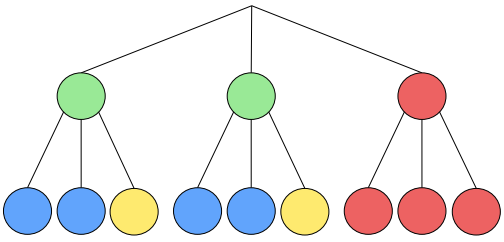


Antonio and Campo [2023] apply feature engineering to construct a risk profile for each class and merge classes within a given level based on that risk profile. They do not allow for the partial collapse of a level.

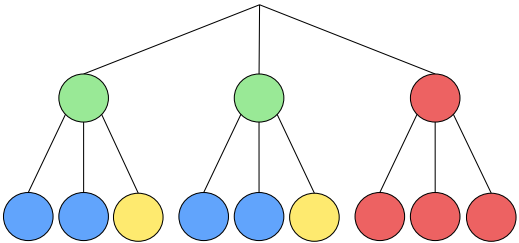


Antonio and Campo [2023] apply feature engineering to construct a risk profile for each class and merge classes within a given level based on that risk profile. They do not allow for the partial collapse of a level.

Carrizosa et al. [2022] introduce the tree based linear regression model (TLR), which allows for the collapse of descendant classes by balancing the predictive accuracy and complexity of the model.

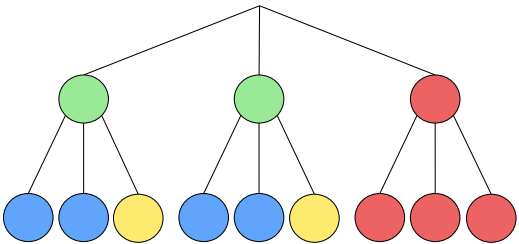


Develop a methodology to reduce both the within-level dimensionality as well as the overall granularity of a hierarchical categorical variable by:



Develop a methodology to reduce both the within-level dimensionality as well as the overall granularity of a hierarchical categorical variable by:

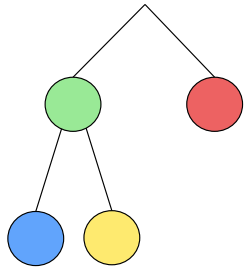
1. learning embedding vectors for every class at each level in the hierarchy, and



Develop a methodology to reduce both the within-level dimensionality as well as the overall granularity of a hierarchical categorical variable by:

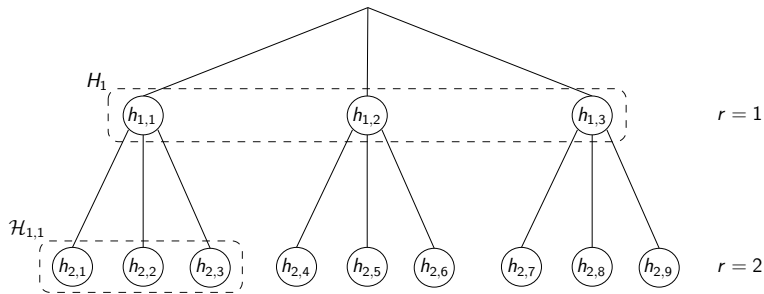
1. learning embedding vectors for every class at each level in the hierarchy, and

2. proposing a clustering algorithm that leverages the information encoded in the embeddings to reduce the hierarchy.



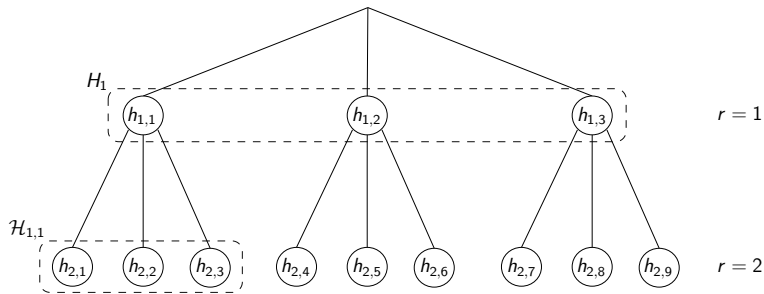
Notation & Setup

Hierarchical categorical variable $\boldsymbol{h} = (h_1, \ldots, h_R)$ with R levels, where $h_r \in H_r = \{h_{r,1}, \ldots, h_{r,n_r}\}$ is a (non-hierarchical) categorical variable.



Hierarchical categorical variable $\boldsymbol{h} = (h_1, .., h_R)$ with R levels, where $h_r \in H_r = \{h_{r,1}, .., h_{r,n_r}\}$ is a (non-hierarchical) categorical variable.

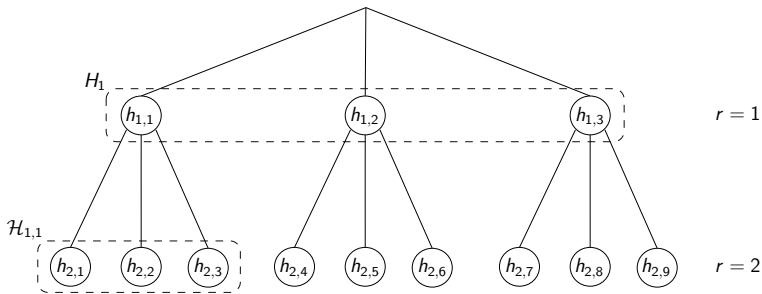
Assume a dataset $\mathcal{D} = (y_i, \boldsymbol{h}_i, \boldsymbol{x}_i)_{i=1}^n$ of n observations.



Hierarchical categorical variable $\boldsymbol{h} = (h_1, .., h_R)$ with R levels, where $h_r \in H_r = \{h_{r,1}, .., h_{r,n_r}\}$ is a (non-hierarchical) categorical variable.

Assume a dataset $\mathcal{D} = (y_i, \boldsymbol{h}_i, \boldsymbol{x}_i)_{i=1}^n$ of n observations.

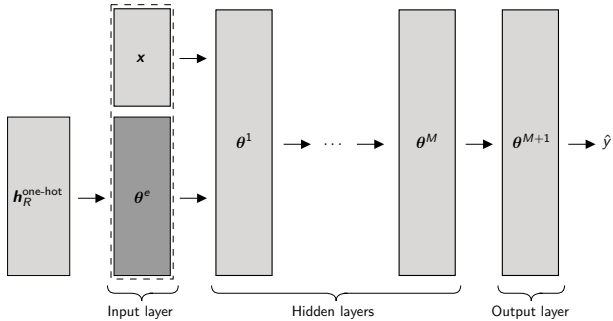
We want to learn $\widetilde{\boldsymbol{h}} = (\widetilde{h}_1, .., \widetilde{h}_{\widetilde{R}})$ with $\widetilde{R} \leq R$ levels, where we have that $\widetilde{n}_r \leq n_r \, \forall \, r = 1, .., \widetilde{R}$.



Methodology

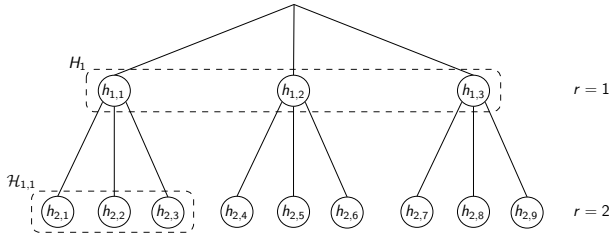
To embed the hierarchy for

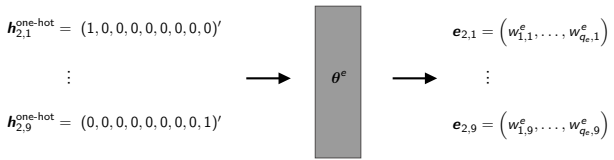
1. $r = R$: we learn a feedforward neural network and apply entity embedding [Guo and Berkhahn 2016] to h_R.



To embed the hierarchy for

1. $r = R$: we learn a feedforward neural network and apply entity embedding [Guo and Berkhahn 2016] to h_R.



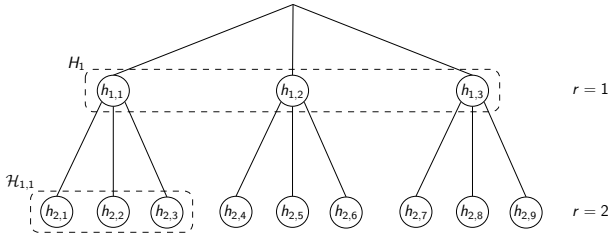


To embed the hierarchy for

1. $r = R$: we learn a feedforward neural network and apply entity embedding [Guo and Berkhahn 2016] to h_R.

2. $r = 1.., R-1$: we average the embeddings over the hierarchical structure:

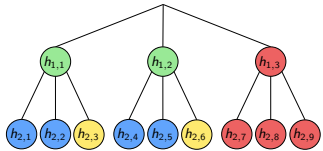
$$\boldsymbol{e}_{r,s} = \frac{1}{\dim(\mathcal{H}_{r,s})} \sum_{l | h_{r+1,l} \in \mathcal{H}_{r,s}} \boldsymbol{e}_{r+1,l}$$
$$\forall\, r = 1, .., R-1,\ \forall\, s = 1, .., n_r.$$



Methodology

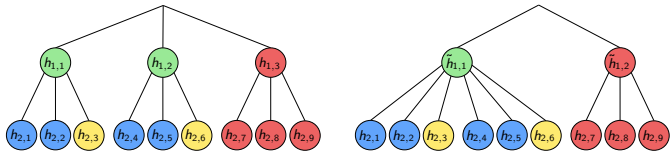
Reducing a hierarchy

We propose a top-down clustering algorithm that for a given level r



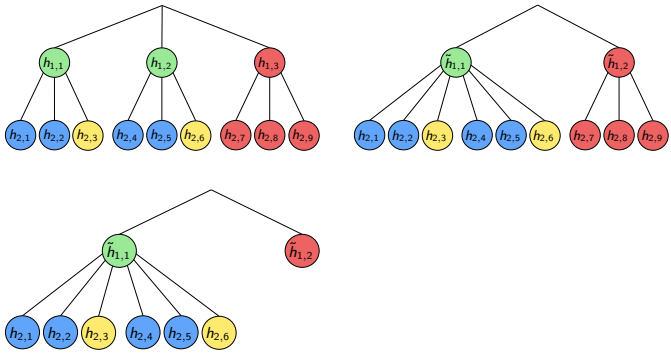
We propose a top-down clustering algorithm that for a given level r

1. merges similar classes within level r, and



We propose a top-down clustering algorithm that for a given level r

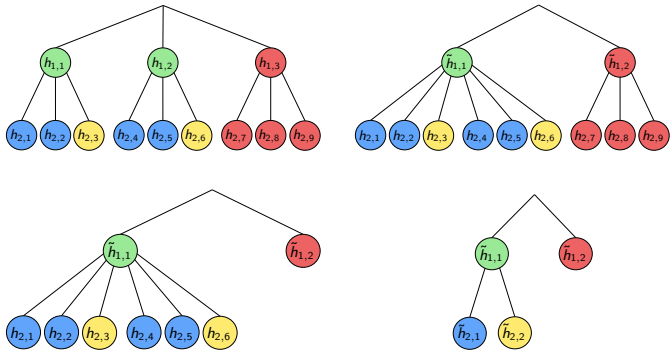
1. merges similar classes within level r, and
2. collapses descendant classes on level $r+1$ that are sufficiently close in the embedding space with their parent class on level r.



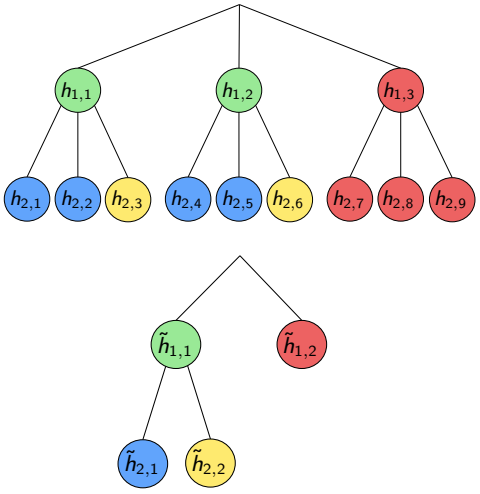
We propose a top-down clustering algorithm that for a given level r

1. merges similar classes within level r, and
2. collapses descendant classes on level $r+1$ that are sufficiently close in the embedding space with their parent class on level r.

Both steps are repeated for every level in the hierarchy, starting from $r = 1$.

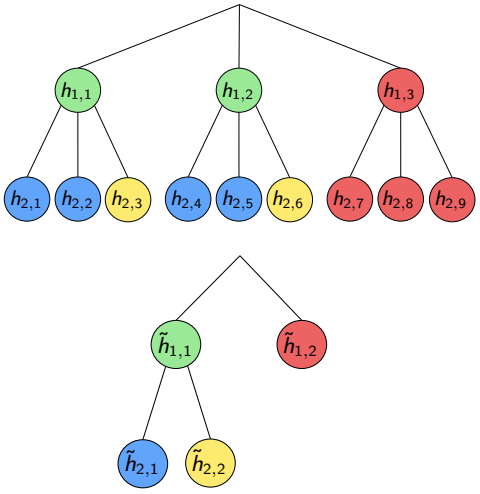


Each step consists of multiple clustering tasks. For each clustering task,



Each step consists of multiple clustering tasks. For each clustering task,

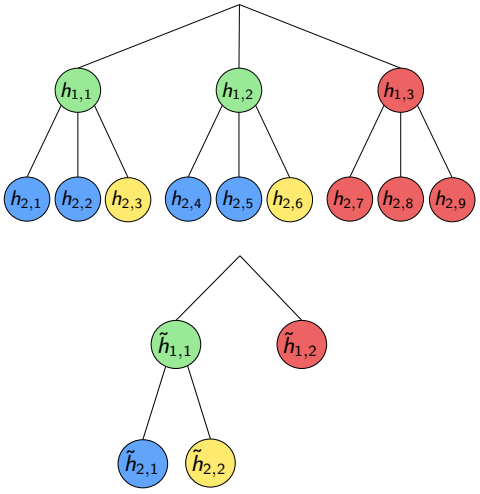
1. we apply the k-medoids algorithm [Kaufman and Rousseeuw 2009] to a set of embeddings corresponding to a subset of classes, and



Each step consists of multiple clustering tasks. For each clustering task,

1. we apply the k-medoids algorithm [Kaufman and Rousseeuw 2009] to a set of embeddings corresponding to a subset of classes, and

2. use the silhouette index [Vendramin et al. 2010] as a cluster validation metric to determine the number of clusters.



Methodology

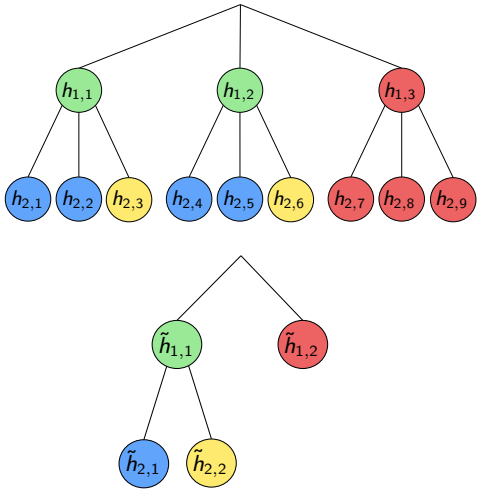
Reducing a hierarchy

Each step consists of multiple clustering tasks. For each clustering task,

1. we apply the k-medoids algorithm [Kaufman and Rousseeuw 2009] to a set of embeddings corresponding to a subset of classes, and

2. use the silhouette index [Vendramin et al. 2010] as a cluster validation metric to determine the number of clusters.

Higher value tuning parameter SI^* results in more reduced hierarchical structure.



Simulation experiments

Simulation experiments

Balanced

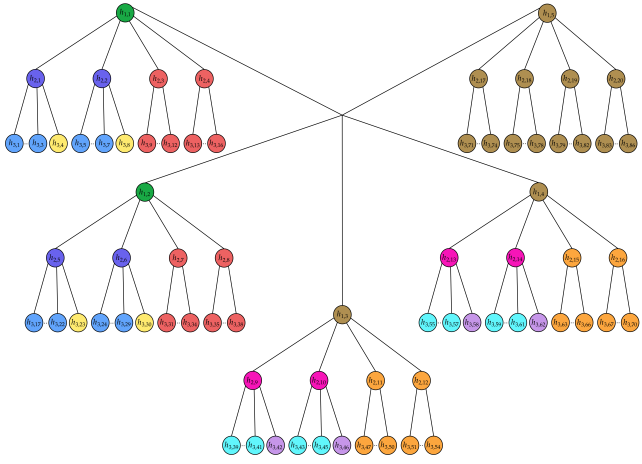
We simulate

Simulation experiments

Balanced

We simulate

1. a hierarchical categorical variable $\boldsymbol{h}$,



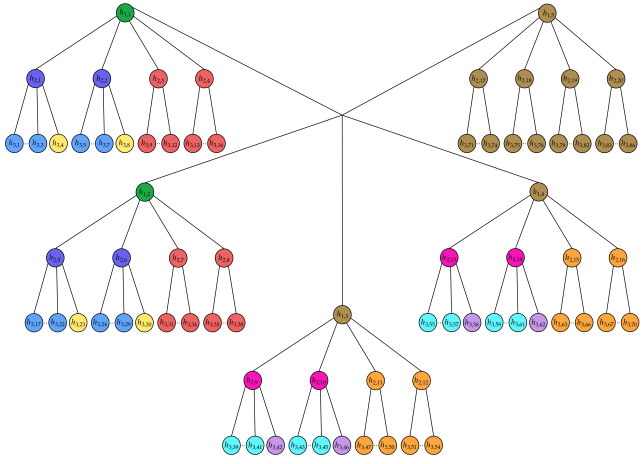
Simulation experiments

Balanced

We simulate

1. a hierarchical categorical variable $\boldsymbol{h}$,

2. additional covariate vector $\boldsymbol{x} = (x_1, x_2, x_3)$ where $x_1 = \sin(a_1)$ with $a_1 \sim U(0, 5)$, $x_2 \sim N(0, 1)$ and $x_3 = a_3^2$ with $a_3 \sim U(1, 2)$, and

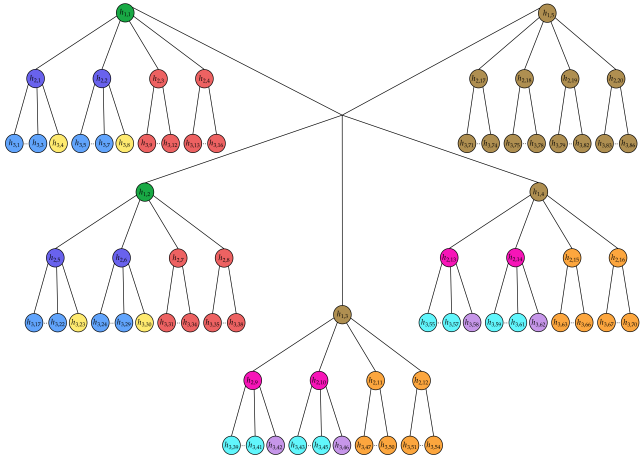


Simulation experiments

Balanced

We simulate

1. a hierarchical categorical variable $\boldsymbol{h}$,
2. additional covariate vector $\boldsymbol{x} = (x_1, x_2, x_3)$ where $x_1 = \sin(a_1)$ with $a_1 \sim U(0, 5)$, $x_2 \sim N(0, 1)$ and $x_3 = a_3^2$ with $a_3 \sim U(1, 2)$, and
3. a response variable y.



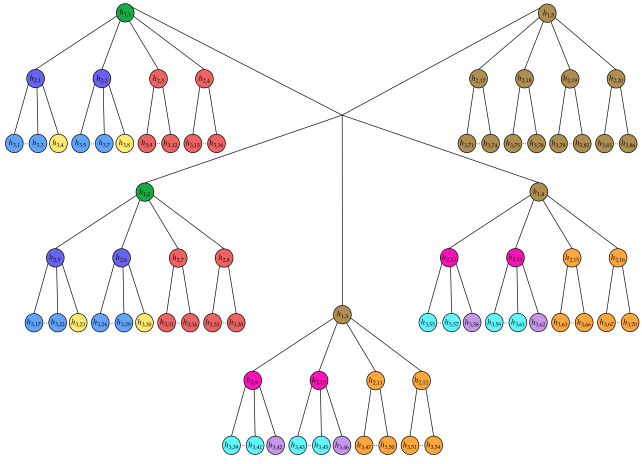
Simulation experiments

Balanced

We simulate

1. a hierarchical categorical variable $\boldsymbol{h}$,
2. additional covariate vector $\boldsymbol{x} = (x_1, x_2, x_3)$ where $x_1 = \sin(a_1)$ with $a_1 \sim U(0, 5)$, $x_2 \sim N(0, 1)$ and $x_3 = a_3^2$ with $a_3 \sim U(1, 2)$, and
3. a response variable y.

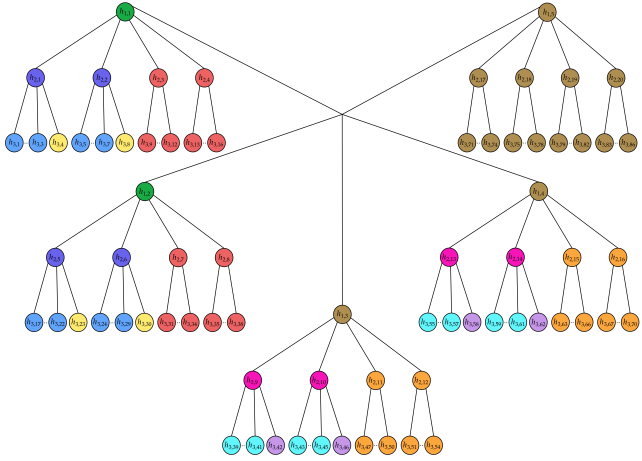
Classes represented using the same colour are simulated to have the same effect on the response.



Simulation experiments

Balanced

We consider the case where

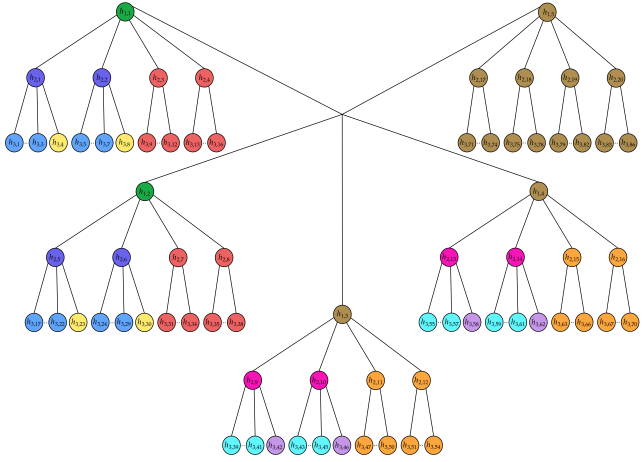


Simulation experiments

Balanced

We consider the case where

1. only $\boldsymbol{h}$ has an effect on y,

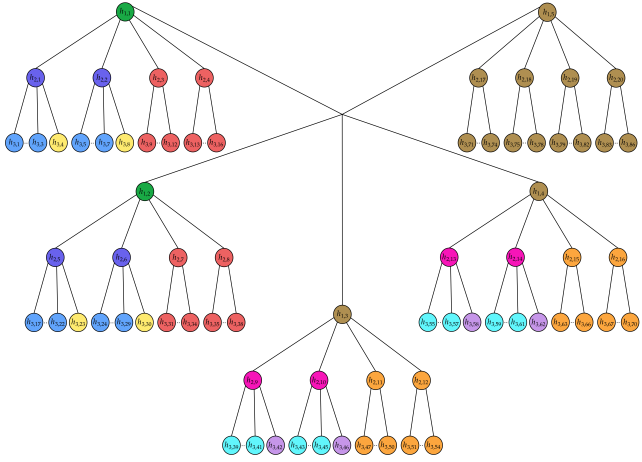


Simulation experiments

Balanced

We consider the case where

1. only $\boldsymbol{h}$ has an effect on y,
2. both $\boldsymbol{h}$ and $\boldsymbol{x}$ have an effect, and

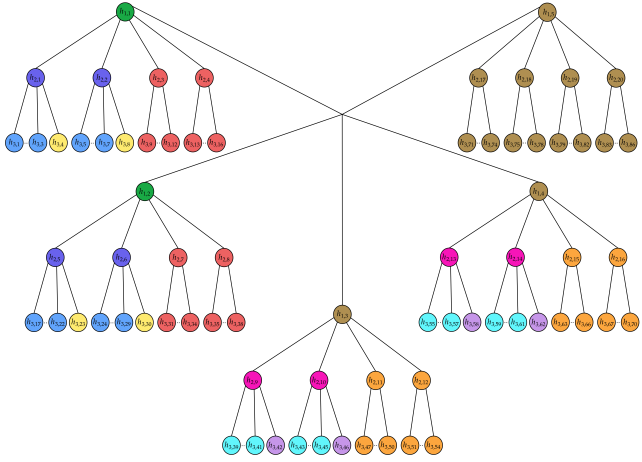


Simulation experiments

Balanced

We consider the case where

1. only $\boldsymbol{h}$ has an effect on y,
2. both $\boldsymbol{h}$ and $\boldsymbol{x}$ have an effect, and
3. the case where both $\boldsymbol{h}$ and $\boldsymbol{x}$ have no effect.



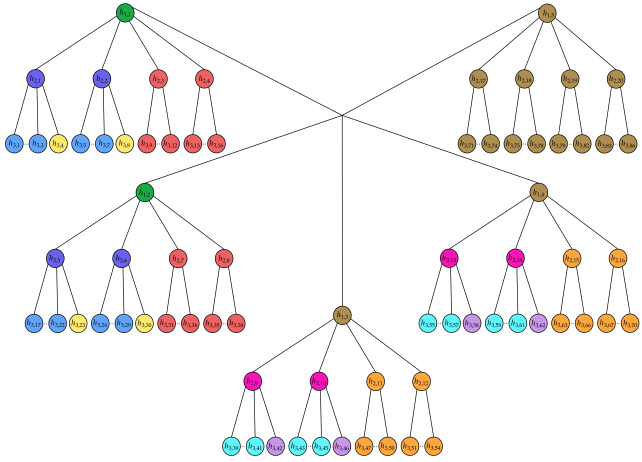
Simulation experiments

Balanced

We consider the case where

1. only $\boldsymbol{h}$ has an effect on y,
2. both $\boldsymbol{h}$ and $\boldsymbol{x}$ have an effect, and
3. the case where both $\boldsymbol{h}$ and $\boldsymbol{x}$ have no effect.

For all three cases, we simulate normally distributed data as well as Poisson data.



Simulation experiments

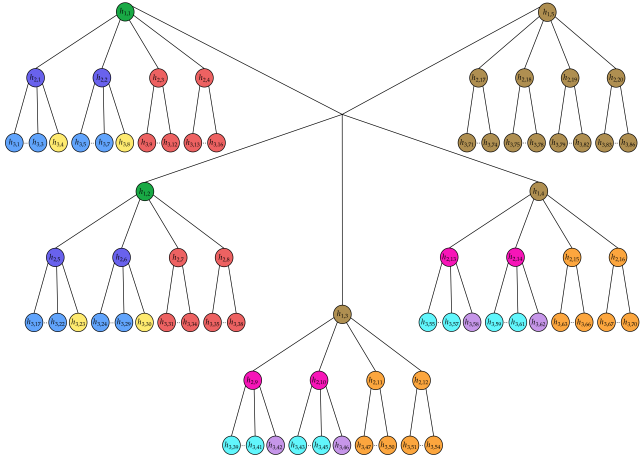
Balanced

We consider the case where

1. only $\boldsymbol{h}$ has an effect on y,
2. both $\boldsymbol{h}$ and $\boldsymbol{x}$ have an effect, and
3. the case where both $\boldsymbol{h}$ and $\boldsymbol{x}$ have no effect.

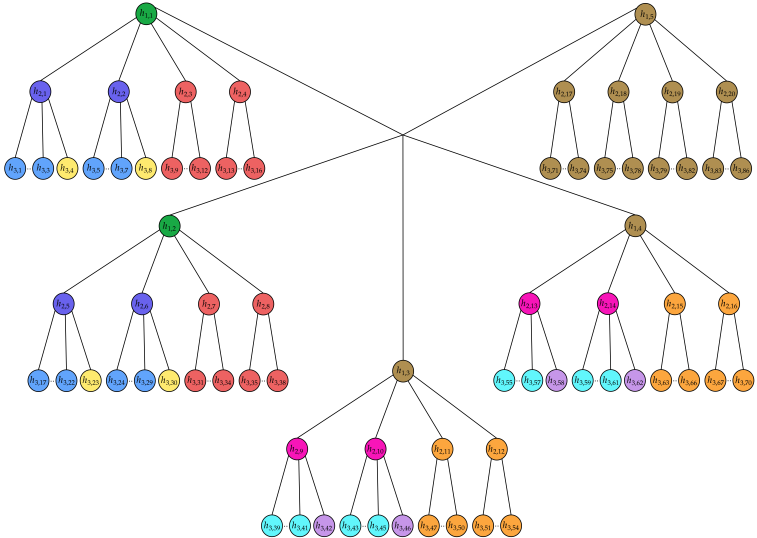
For all three cases, we simulate normally distributed data as well as Poisson data.

For each experiment, 100 datasets consisting of 1000 observations of each class at the lowest level in the hierarchy, i.e. h_R, are simulated.



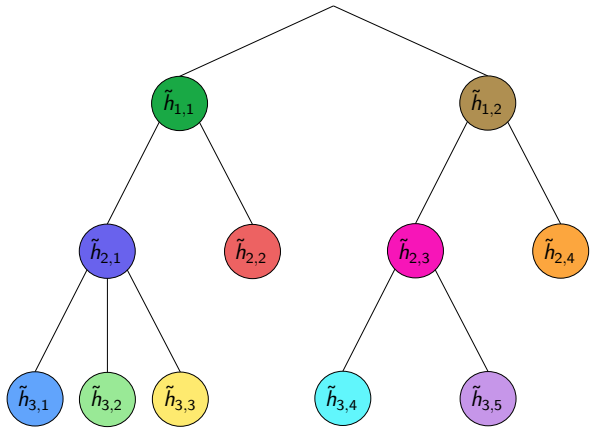
Simulation experiments

Balanced



Simulation experiments

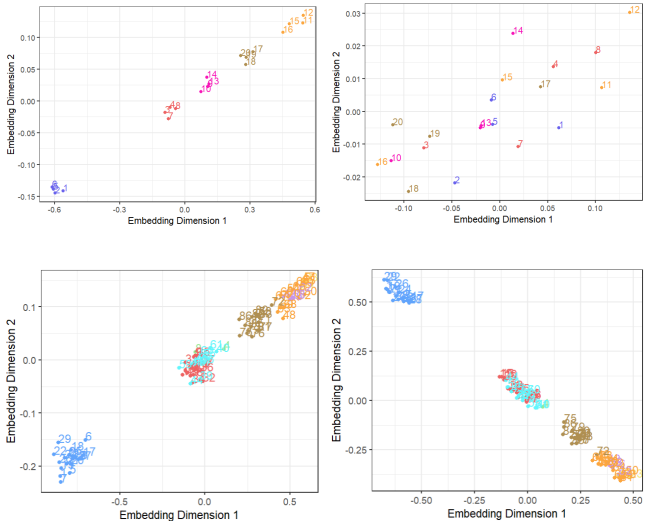
Balanced



Simulation experiments

Balanced - methodology

We set the embedding dimension $q_e = 2$ to visualise the embedding space.

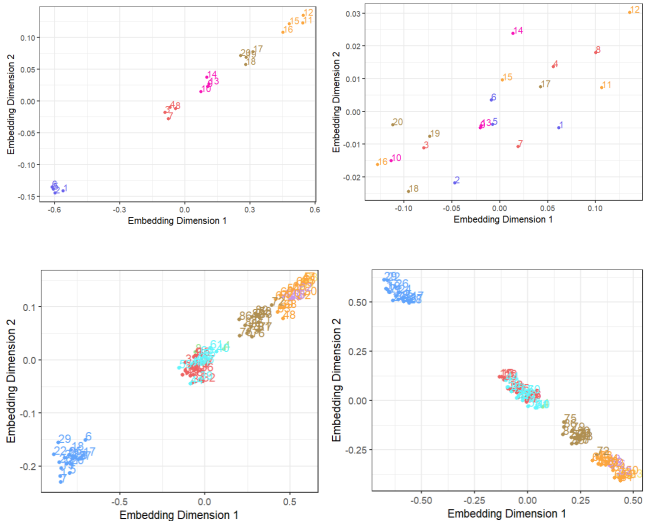


Simulation experiments

Balanced - methodology

We set the embedding dimension $q_e = 2$ to visualise the embedding space.

To learn the embedding vectors, we use a network with a single hidden layer consisting of two neurons.



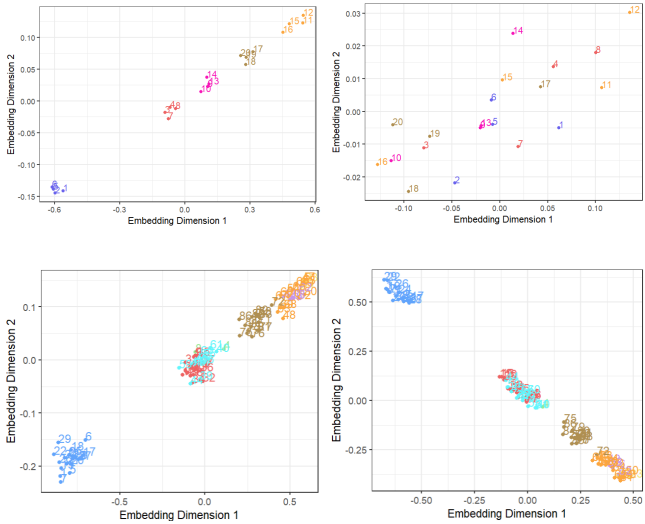
Simulation experiments

Balanced - methodology

We set the embedding dimension $q_e = 2$ to visualise the embedding space.

To learn the embedding vectors, we use a network with a single hidden layer consisting of two neurons.

The activation function in the output layer is the identity function or the exponential function for normal and Poisson data, respectively.



Simulation experiments

Balanced - results

	True structure retrieved	Different structures
Normal distribution		
$\boldsymbol{h}$ no effect	96.6%	9
$\boldsymbol{h}$ with effect	90.4%	8
$\boldsymbol{h}$ and $\boldsymbol{x}$ with effect	92.8%	8
Poisson distribution		
$\boldsymbol{h}$ no effect	92.6%	19
$\boldsymbol{h}$ with effect	97.4%	2
$\boldsymbol{h}$ and $\boldsymbol{x}$ with effect	99%	2

Simulation experiments

Balanced - results

Most of time, the true structure is retrieved. If not, the retrieved structure closely resembles the true structure.

	True structure retrieved	Different structures
Normal distribution		
$\boldsymbol{h}$ no effect	96.6%	9
$\boldsymbol{h}$ with effect	90.4%	8
$\boldsymbol{h}$ and $\boldsymbol{x}$ with effect	92.8%	8
Poisson distribution		
$\boldsymbol{h}$ no effect	92.6%	19
$\boldsymbol{h}$ with effect	97.4%	2
$\boldsymbol{h}$ and $\boldsymbol{x}$ with effect	99%	2

Simulation experiments

Balanced - results

Most of time, the true structure is retrieved. If not, the retrieved structure closely resembles the true structure.

Slightly better performance on the Poisson data.

	True structure retrieved	Different structures
Normal distribution		
$\boldsymbol{h}$ no effect	96.6%	9
$\boldsymbol{h}$ with effect	90.4%	8
$\boldsymbol{h}$ and $\boldsymbol{x}$ with effect	92.8%	8
Poisson distribution		
$\boldsymbol{h}$ no effect	92.6%	19
$\boldsymbol{h}$ with effect	97.4%	2
$\boldsymbol{h}$ and $\boldsymbol{x}$ with effect	99%	2

Simulation experiments

Balanced - results

Most of time, the true structure is retrieved. If not, the retrieved structure closely resembles the true structure.

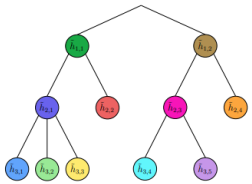
Slightly better performance on the Poisson data.

Higher number of different structures retrieved in case there is no effect of $\boldsymbol{h}$ on the response.

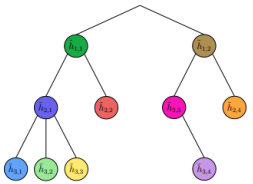
	True structure retrieved	Different structures
Normal distribution		
$\boldsymbol{h}$ no effect	96.6%	9
$\boldsymbol{h}$ with effect	90.4%	8
$\boldsymbol{h}$ and $\boldsymbol{x}$ with effect	92.8%	8
Poisson distribution		
$\boldsymbol{h}$ no effect	92.6%	19
$\boldsymbol{h}$ with effect	97.4%	2
$\boldsymbol{h}$ and $\boldsymbol{x}$ with effect	99%	2

Simulation experiments

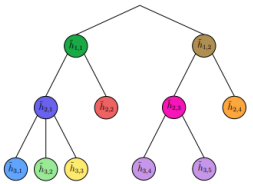
Balanced - results



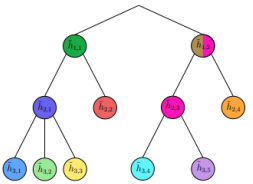
(a) 90.4% of cases



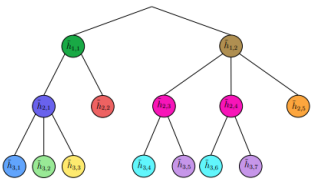
(b) 7.8% of cases



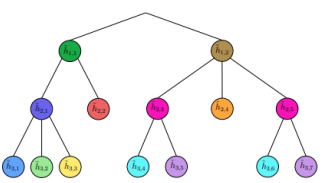
(c) 0.6% of cases



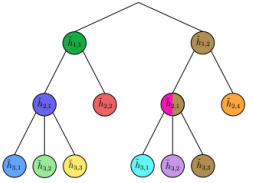
(d) 0.4% of cases



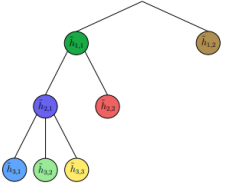
(e) 0.2% of cases



(f) 0.2% of cases



(g) 0.2% of cases



(h) 0.2% of cases

Simulation experiments

Unbalanced - results

	Number of observations in each class of h_R			
	50-100	50-150	50-200	50-250
True structure retrieved	43.2%	52.8%	60.8%	68.4%
Different structures	51	32	39	27
$AIC(\widetilde{\boldsymbol{h}}) < AIC(\boldsymbol{h})$	99.4%	100%	100%	100%
$BIC(\widetilde{\boldsymbol{h}}) < BIC(\boldsymbol{h})$	100%	100%	100%	100%

Simulation experiments

Unbalanced - results

Less observations decreases the number of times the true structure is retrieved and increases the number of different structures.

	Number of observations in each class of h_R			
	50-100	50-150	50-200	50-250
True structure retrieved	43.2%	52.8%	60.8%	68.4%
Different structures	51	32	39	27
$AIC(\widetilde{\boldsymbol{h}}) < AIC(\boldsymbol{h})$	99.4%	100%	100%	100%
$BIC(\widetilde{\boldsymbol{h}}) < BIC(\boldsymbol{h})$	100%	100%	100%	100%

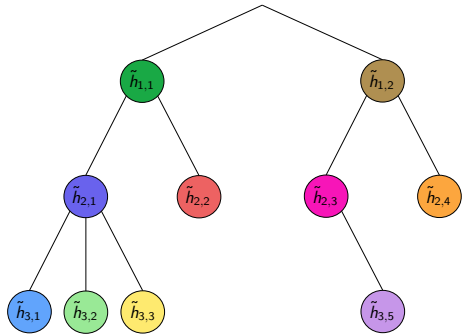
Simulation experiments

Unbalanced - results

	Number of observations in each class of h_R			
	50-100	50-150	50-200	50-250
True structure retrieved	43.2%	52.8%	60.8%	68.4%
Different structures	51	32	39	27
$AIC(\widetilde{\boldsymbol{h}}) < AIC(\boldsymbol{h})$	99.4%	100%	100%	100%
$BIC(\widetilde{\boldsymbol{h}}) < BIC(\boldsymbol{h})$	100%	100%	100%	100%

Less observations decreases the number of times the true structure is retrieved and increases the number of different structures.

Overall, even when the number of observations is decreased, the retrieved structures still closely resemble the true structure.



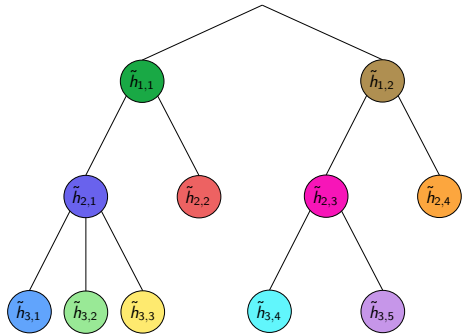
Simulation experiments

Unbalanced - results

	Number of observations in each class of h_R			
	50-100	50-150	50-200	50-250
True structure retrieved	43.2%	52.8%	60.8%	68.4%
Different structures	51	32	39	27
$AIC(\widetilde{\boldsymbol{h}}) < AIC(\boldsymbol{h})$	99.4%	100%	100%	100%
$BIC(\widetilde{\boldsymbol{h}}) < BIC(\boldsymbol{h})$	100%	100%	100%	100%

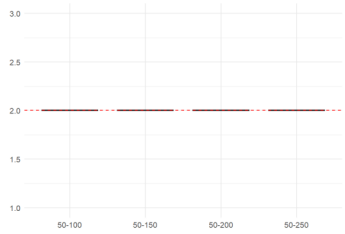
Less observations decreases the number of times the true structure is retrieved and increases the number of different structures.

Overall, even when the number of observations is decreased, the retrieved structures still closely resemble the true structure.

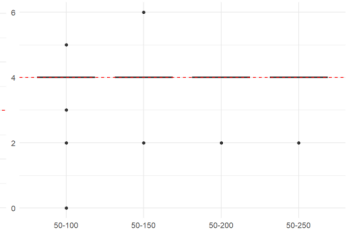


Simulation experiments

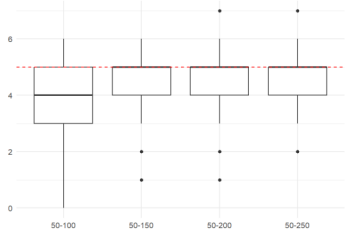
Unbalanced - results



(a) $r = 1$



(b) $r = 2$



(c) $r = 3$

Application to a real dataset

Application to a real dataset

`cancer_reg` dataset used by Carrizosa et al. [2022] consisting of 3047 observations including

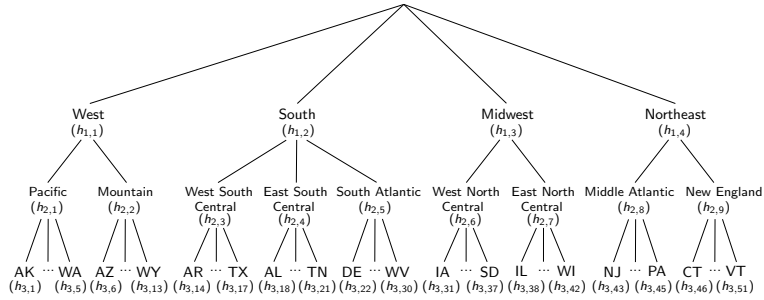
`cancer_reg` dataset used by Carrizosa et al. [2022] consisting of 3047 observations including

1. 31 non-hierarchical covariates describing socio-economic information,

Application to a real dataset

`cancer_reg` dataset used by Carrizosa et al. [2022] consisting of 3047 observations including

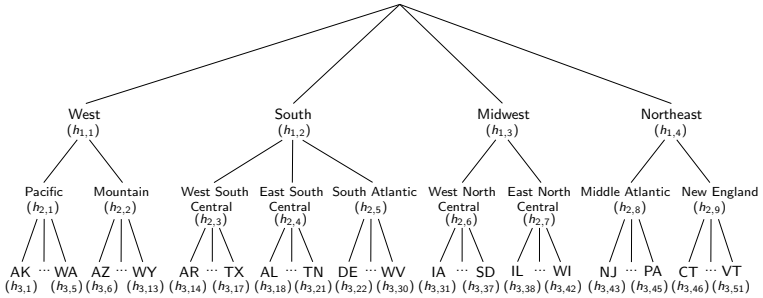
1. 31 non-hierarchical covariates describing socio-economic information,
2. hierarchical variable `geography` consisting of three levels, and



Application to a real dataset

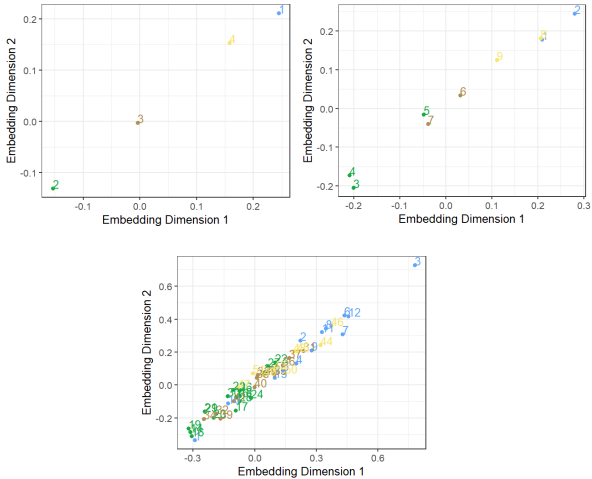
`cancer_reg` dataset used by Carrizosa et al. [2022] consisting of 3047 observations including

1. 31 non-hierarchical covariates describing socio-economic information,
2. hierarchical variable `geography` consisting of three levels, and
3. a response variable relating to cancer mortality.



Application to a real dataset

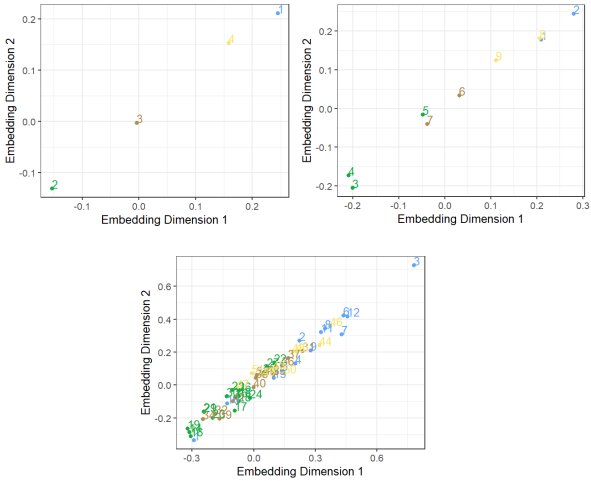
We apply the same network structure as before.



Application to a real dataset

We apply the same network structure as before.

We standardise the non-hierarchical continuous predictors.

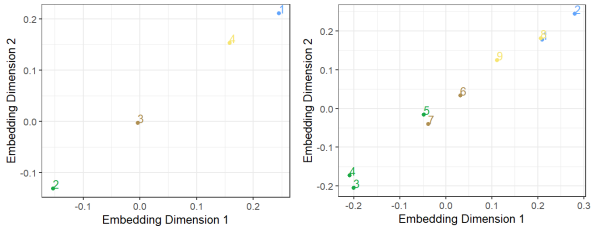


Application to a real dataset

We apply the same network structure as before.

We standardise the non-hierarchical continuous predictors.

We exclude the variables `pctsomecol18_24`, `pctemployed16_over` and `pctprivatecoveragealone` due to missing values.



Application to a real dataset

Results

We consider a grid of possible values for the tuning parameter SI^*.

		AIC	*BIC*
h		6087.80	6617.73
SI^*	0.1	**6074.261**	6363.31
	0.3	**6074.261**	6363.31
	0.5	6075.01	**6352.02**
	0.7	6449.19	6678.03
`Carrizosa_AIC`		6083.19	6570.96
`Carrizosa_BIC`		6169.04	6476.15

Application to a real dataset

Results

We consider a grid of possible values for the tuning parameter SI^*.

BIC indicates a simpler representation of the hierarchical categorical variable compared to the AIC.

		AIC	BIC
$\boldsymbol{h}$		6087.80	6617.73
SI^*	0.1	**6074.261**	6363.31
	0.3	**6074.261**	6363.31
	0.5	6075.01	**6352.02**
	0.7	6449.19	6678.03
Carrizosa_AIC		6083.19	6570.96
Carrizosa_BIC		6169.04	6476.15

We consider a grid of possible values for the tuning parameter SI^*.

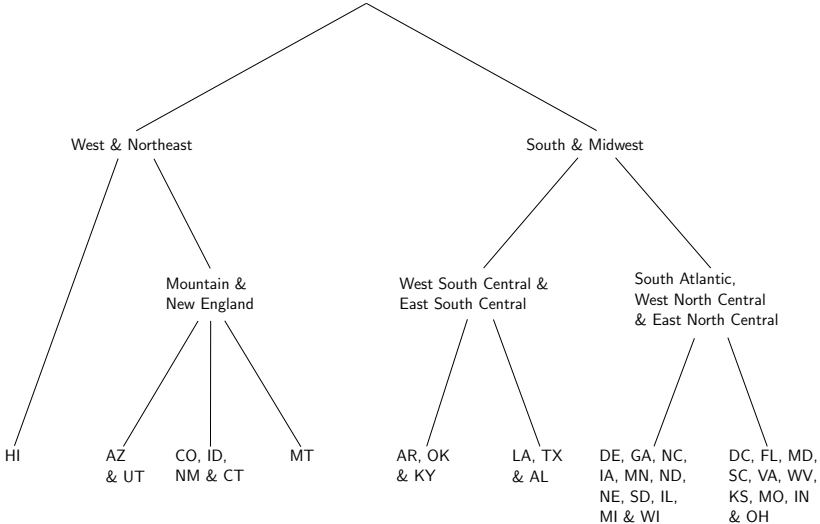
BIC indicates a simpler representation of the hierarchical categorical variable compared to the *AIC*.

Our methodology improves the model fit over the original structure and outperforms the solution provided by Carrizosa et al. [2022].

		AIC	*BIC*
h		6087.80	6617.73
SI^*	0.1	**6074.261**	6363.31
	0.3	**6074.261**	6363.31
	0.5	6075.01	**6352.02**
	0.7	6449.19	6678.03
`Carrizosa_AIC`		6083.19	6570.96
`Carrizosa_BIC`		6169.04	6476.15

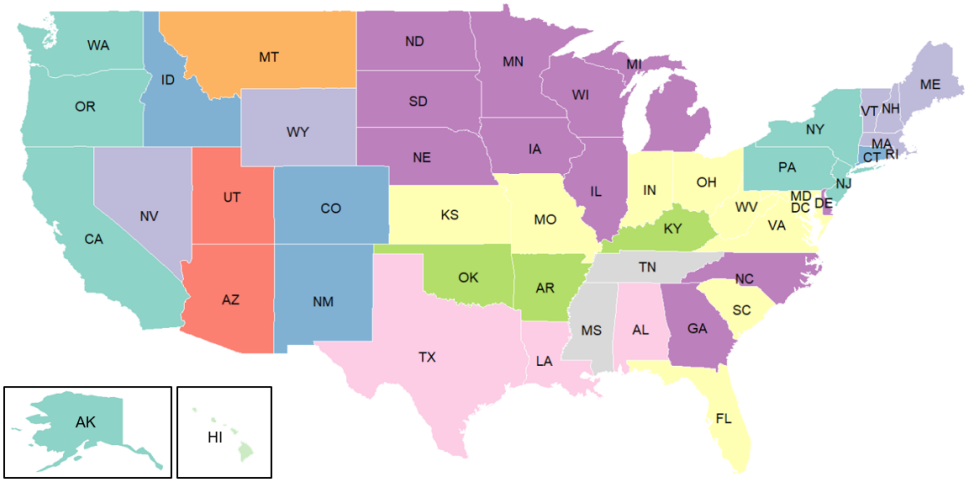
Application to a real dataset

Results

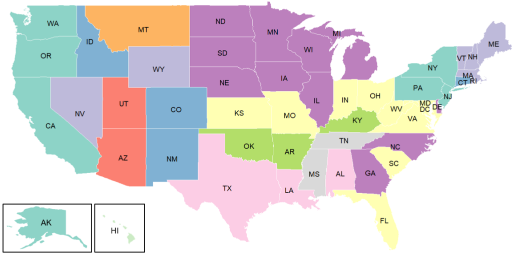


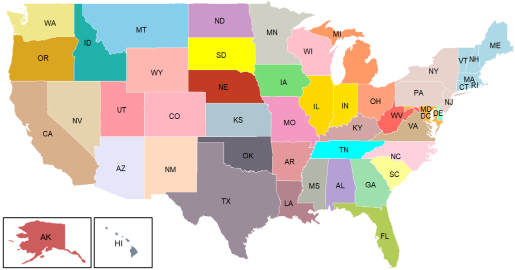
Application to a real dataset

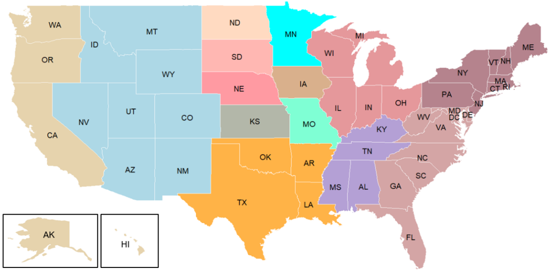
Results



Application to a real dataset







Conclusions

We propose a novel methodology relying on entity embeddings and clustering techniques to reduce the dimensionality and granularity of a hierarchical categorical variable.

The resulting reduced hierarchical structure can be incorporated in any type of predictive model.

Using simulated data, we find that our methodology can effectively approximate the true underlying structure with respect to a response variable.

We verify our approach on a real dataset and show that it outperforms existing solutions.

We propose a novel methodology relying on entity embeddings and clustering techniques to reduce the dimensionality and granularity of a hierarchical categorical variable.

The resulting reduced hierarchical structure can be incorporated in any type of predictive model.

Using simulated data, we find that our methodology can effectively approximate the true underlying structure with respect to a response variable.

We verify our approach on a real dataset and show that it outperforms existing solutions.

Conclusions

We propose a novel methodology relying on entity embeddings and clustering techniques to reduce the dimensionality and granularity of a hierarchical categorical variable.

The resulting reduced hierarchical structure can be incorporated in any type of predictive model.

Using simulated data, we find that our methodology can effectively approximate the true underlying structure with respect to a response variable.

We verify our approach on a real dataset and show that it outperforms existing solutions.

We propose a novel methodology relying on entity embeddings and clustering techniques to reduce the dimensionality and granularity of a hierarchical categorical variable.

The resulting reduced hierarchical structure can be incorporated in any type of predictive model.

Using simulated data, we find that our methodology can effectively approximate the true underlying structure with respect to a response variable.

We verify our approach on a real dataset and show that it outperforms existing solutions.

Thank you!

Katrien Antonio and Bavo DC Campo. On clustering levels of a hierarchical categorical risk factor. **Annals of Actuarial Science**, 2023.

Emilio Carrizosa, Laust Hvas Mortensen, Dolores Romero Morales, and M Remedios Sillero-Denamiel. The tree based linear regression model for hierarchical categorical variables. **Expert Systems with Applications**, 203:117423, 2022.

Cheng Guo and Felix Berkhahn. Entity embeddings of categorical variables. **arXiv preprint arXiv:1604.06737**, 2016.

Leonard Kaufman and Peter J Rousseeuw. **Finding groups in data: an introduction to cluster analysis**. John Wiley & Sons, 2009.

Ronald Richman and Mario V Wüthrich. High-cardinality categorical covariates in network regressions. **Japanese Journal of Statistics and Data Science**, pages 1–45, 2024.

Lucas Vendramin, Ricardo JGB Campello, and Eduardo R Hruschka. Relative clustering validity criteria: A comparative overview. **Statistical analysis and data mining: the ASA data science journal**, 3(4):209–235, 2010.