Sufficient Representations for Categorical Variables

Jonathan Johannemann
jonjoh@stanford.edu

Vitor Hadad
vitorh@stanford.edu

Susan Athey
athey@stanford.edu

Stefan Wager
swager@stanford.edu

Stanford University

Abstract

Many learning algorithms require categorical data to be transformed into real vectors before it can be used as input. Often, categorical variables are encoded as *one-hot* or *dummy* vectors. However, this mode of representation can be wasteful since it adds many low-signal regressors, especially when the number of unique categories is large. In this paper, we investigate simple alternative solutions for universally consistent estimators that rely on lower-dimensional real-valued representations of categorical variables that are *sufficient* in the sense that no predictive information is lost. We then compare preexisting and proposed methods on simulated and observational datasets.

1 Introduction

Many regression problems involve data collected from a number groups that may be statistically relevant. For example, in a medical setting, we may want to model health outcomes using data on patients from several hospitals and acknowledge that different hospitals may have idiosyncratic effects on patients that are not explained by other covariates. Similar considerations arise when working with data on students from different schools, voters from different zip-codes, employees at different firms, etc.

One of the most wide-spread approaches to this problem is via fixed effect modeling, as follows. Suppose that we observe n samples (X_i, G_i, Y_i) for $i = 1, ..., n$, where $X_i \in \mathbb{R}^p$ is a set of subject-specific covariates, $G_i \in \mathcal{G}$ is a categorial variable that records group membership and $Y_i \in \mathbb{R}$ is the response of interest, and want to estimate

$$\mu(x, g) = \mathbb{E}\left[Y_i \,\middle|\, X_i = x, G_i = g\right]. \tag{1}$$

Then, the simple fixed effects approach starts by positing a model

$$\mu(x, g) = \alpha_g + x\beta, \tag{2}$$

and then estimating the coefficients β and α_g via ordinary least squares regression. More sophisticated extensions of this approach may involve considering non-linear transformations of x, interactions between group membership and the covariates x, and/or regularization [Angrist and Pischke, 2008, Diggle et al., 2002, Wooldridge, 2010].

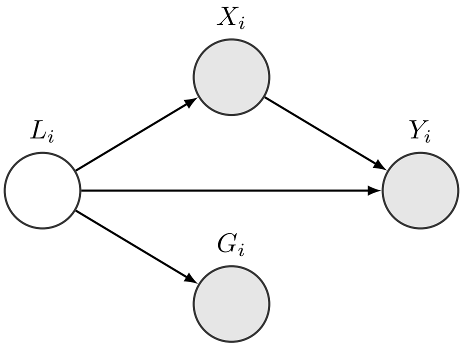


Figure 1: Causal graph depicting the key assumption that Y_i and X_i are independent of group membership G_i conditionally on latent state L_i. The grayed-out nodes are observed.

Fixed effects modeling, however, does not always perform well with complex non-linear signals or when the number of groups $|\mathcal{G}|$ is large. The model (2) is quite rigid and may not be able to represent rich signals while, at the same time, the large number of α_g parameters in the model (2) may result in problems for statistical inference [Neyman and Scott, 1948]. In other words, the model (2) may have too many parameters to be stable all while lacking the degrees of freedom to fit the signal well.

The goal of this paper is to develop a more parsimonious approach for representing group membership to avoid the above problems. Specifically, we seek a mapping ψ that embeds group membership G_i into a k-dimensional space without losing any predictive information, i.e.,

$$\psi : \mathcal{G} \to \mathbb{R}^k, \quad \mu(x, g) = f(x, \psi(g)), \tag{3}$$

such that k is small (in particular, $k \ll |\mathcal{G}|$) and the function $f(\cdot, \cdot)$ is still easy to learn. Given such a mapping, the problem (1) becomes a routine regression problem with $(p + k)$-dimensional real-valued features $(X_i, \psi(G_i))$, and we can use out-of-the-box statistical learning software on it.

In order to obtain a useful representation of group membership G_i, we of course need to assume something about the relationship between G_i and the target outcome Y_i. The core assumption made in this paper is what we call the *sufficient latent state assumption* depicted in Figure 1: group membership G_i has no direct causal effect on Y_i, but may be associated with latent variables L_i that do have a direct effect on Y_i. For example, in the case of patients spread across hospitals, we assume that hospitals themselves do not directly *cause* health outcomes Y_i; however, hospitals may still be *predictive* of Y_i through their association with latent causal variables. Patients may have unobserved characteristics, e.g., severity of disease or socioeconomic resources, that both affect Y_i and lead the patient to self-select into different hospitals. Our main result is that, under this sufficient latent state assumption, practical representations of the form (3) exist and can be learned from data.

The principle of representing high-cardinality categorical variables as real-valued vectors has played an important role in many different areas. For example, in natural language processing, it is now common to start more complex analyses with a pre-processing step that represent words as vectors that capture the way in which the words are used in context [Mikolov et al., 2013, Pennington et al., 2014]. Meanwhile, in the literature on panel data analysis, our approach is perhaps most closely related to a proposal of Bonhomme and Manresa [2015] where individual time series belong to discrete clusters and we have only one fixed effect per cluster (rather than one per time series). Bonhomme and Manresa [2015] then fit this model via a k-means like algorithm that alternates clustering and estimation with per-cluster fixed effects.[1] Finally, in the causal inference literature, Arkhangelsky and Imbens [2018] state that simply accounting for group differences through additive fixed effects may not be sufficient to adjust for all relevant differences. They propose the use of group characteristics instead of linear fixed effects modeling and offer $(\overline{X}_{C_i}, \overline{W}_{C_i})$ as a sufficient alternative where $\overline{X}_{C_i}$ is the average of the covariates per group C_i and $\overline{W}_{C_i}$ is the average treatment probability per group C_i. This better addresses the lack of functional flexibility but works with a set of different assumptions unlike those we introduce in the sufficient latent state assumption 1. The resulting framework provides a means for introducing lower dimensional, sufficient representations of categorical variables.

Our paper is structured as follows. We begin by reviewing similar problem settings in the fixed effects literature and the drawbacks of using existing methods in 1.1. In Section 2, we introduce the primary lemma which seeks to describe the true information we wish to extract from categorical variables. In Section 3, we expand on lemma 1 to develop methods that utilize this insight. In Sections 4 and 5, we run simulated and observational experiments with our proposed methods and follow up with discussion on how realized performance compared to our expectations.

1.1 Related Work

Traditionally, the discussion of how best to account for group membership G_i in a non-parametric regression has focused on different ways to encode G_i that can be given as an input to statistical software. One simple way to do so is via one-hot encoding: $\iota : \mathcal{G} \to \{0, 1\}^M$ such that the j-th entry of $\iota(g)$ is 1 if and only if g corresponds to the j-th element in $\mathcal{G}$, and where $M := |\mathcal{G}|$. Note that linear regression on one-hot encoded features $(X_i, \iota(G_i))$ exactly recovers the standard fixed effects model (2).

As discussed above, however, one-hot encoding may lead to undesirably high-dimensional problems when $M := |\mathcal{G}|$ is large.[2] In the Appendix (6.2), we present multiple encoding methods that similarly project the categories onto $\mathbb{R}^M$. These methods do not utilize information from the covariates X_i or response Y_i and suffer from the same pitfalls that come with high dimensional representation of the observed groups $\mathcal{G}$. The primary difference for these methods are the user's interpretation of the encoded variables which are commonly constructed as the comparison of the mean effect of a subset $\mathcal{G}' \in \mathcal{G}$ relative to the mean effect of the set $\mathcal{G} \backslash \mathcal{G}'$ or one of its subsets.

The problem of fixed effects is especially challenging with sparsity-seeking methods such as the lasso [Hastie et al., 2015] or decision trees [Breiman et al., 1984], and related ensemble methods such as random forests [Breiman, 2001] or gradient-boosted trees [Friedman, 2001]. Sparsity-seeking methods will set the contribution of features to zero unless there is strong evidence that those features matters for prediction, and it is difficult for rare levels of G_i

[1] Our approach is not directly comparable to either of these methods, as we do not focus on textual data, and do not assume that the latent state L_i can be consistently estimated (in contrast, Bonhomme and Manresa [2015] assume that they have access to long enough time series that their clustering step is consistent which, in our setting, would be equivalent to assuming that L_i can be recovered).

[2] Another slightly more subtle difficulty is that when the categorical variable has many levels, the individual features $\iota(G_i)_j$ become very sparse (i.e., they are usually 0 and only very rarely 1). Many approaches to statistical learning work better with features whose variance roughly captures their range than with such spiky features.

to produce sufficient evidence to get a non-zero contribution to the model via their one-hot features. The end result is that sparsity seeking methods may largely ignore high-cardinality one-hot encoded factors.

Another prevalent way of working with categorical variables and decision trees is to consider full factorial splits that allow for arbitrary grouping of the levels of the categorical variable. For a variable with $M := |\mathcal{G}|$ levels, this allows for $2^{M-1} - 1$ potential splits. Breiman et al. [1984] showed that we can optimize over this exponential set of potential splits in time that scales linearly in M; however, from a statistical point of view, such factorial splits are prone to very strong overfitting when the number of levels is large.

Recently, Cerda et al. [2018] consider a related problem of representing "dirty" categorical variables that might arise if, e.g., several categorical levels are just misspellings of each other, and propose using a low-dimensional embedding that exploits lexicographic similarity (i.e., factors with similar spellings are arranged close to each other). In this paper, we use information in the X_i, rather than lexicographic information, to construct an embedding; however, the high-level conclusion that we can achieve meaningful gains by using auxiliary information to embed categorical variables in a low-rank space remains.

We also note, it is sometimes possible to achieve strong results by randomly projecting a one-hot representation of the categorical variables into $\mathbb{R}^k$ for relatively small k [Rahimi and Recht, 2008]. We also consider this approach but find that, at least in our experiments, we can achieve better performance using carefully crafted representations that leverage continuous covariates.

2 Representing Groups with Sufficient Latent State

Our *sufficient latent state* assumption presented in the introduction and depicted in the causal graph 1 implies that the distribution of the outcome Y_i only depends on the observable factor G_i through some unobservable latent variable L_i. In other words, if we knew the value of L_i, then also knowing G_i would give us no additional information about the outcome. For a simple example, one may posit that a patient's underlying health status ($L_i \in \{\text{good}, \text{poor}\}$) may simultaneously determine to which hospital they are admitted (G_i), what symptoms (X_i) they exhibit, and what health outcomes outcomes (Y_i) they attain. Conditioned on the underlying health status, the hospital cannot provide any additional information about any of the other variables. Conversely, learning their hospital is only helpful inasmuch it allows us to infer something about their health status.

The following lemma states that we can characterize *how* the information about the categorical variable G_i enters the model: the conditional expectation function of the outcome depends only on the *conditional probabilities of the latent variable given the observable category*. This fact will be crucial when deriving the representation methods in future sections.

Lemma 1. *Suppose that the latent state L_i is discrete with k possible levels, and that the probabilistic structure required by the sufficient latent state assumption (Figure 1) holds. Then,*

$$\psi : \mathcal{G} \to \mathbb{R}^k, \quad \psi_l(g) = \mathbb{P}\left[L_i = l \,\middle|\, G_i = g\right] \tag{4}$$

provides a sufficient representation of G_i in the sense of (3):

$$\mu(x,\, g) = \frac{\sum_{l=1}^{k} \mathbb{E}\left[Y_i \,\middle|\, X_i = x,\, L_i = l\right] \mathbb{P}\left[X_i = x \,\middle|\, L_i = l\right] \psi_l(g)}{\sum_{l=1}^{k} \mathbb{P}\left[X_i = x \,\middle|\, L_i = l\right] \psi_l(g)}. \tag{5}$$

Expression (5) formalizes the intution laid out in the previous paragraph. The information associated with the category only enters the conditional expectation via the set of probabilities $\mathbb{P}\left[L_i = \ell \,\middle|\, G_i = g\right]$. If there are only k latent groups, then each category can be represented in a lossless manner by a k dimensional vector of probabilities. An immediate consequence of this result is that if we knew ψ and gave training examples $((X_i,\, \psi(G_i)),\, Y_i)$ to any universally consistent learner, the learner would eventually recover the optimal prediction function $\mu(\cdot)$. To continue the example at the top of this section, the identity of the hospital enters the model through the probability that a patient is in good or poor health given the hospital.

The dependence of the conditional expectation function μ on the latent variable probabilities ψ via (5) is non-linear; however, we will retain consistency if we use an expressive enough method for learning on $((X_i,\, \psi(G_i)),\, Y_i)$. Methods known to be universally consistent include k-nearest neighbors [Stone, 1977], various tree-based ensembles [Biau et al., 2008], and neural networks [Faragó and Lugosi, 1993].

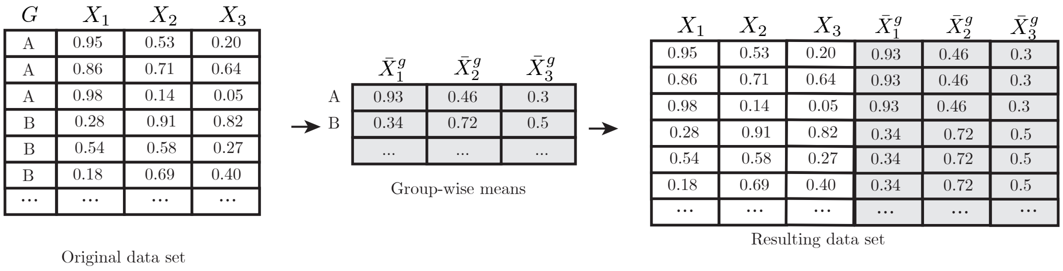
The discussion above seems to imply that we need to estimate $\psi(g) = \mathbb{P}\left[L_i | G_i = g\right]$ directly. However, because this quantity depends on the unobservable variable L_i, its identification is impossible without further assumptions and a more sophisticated approach. Instead we pursue a simpler approach by seeking different functions $f(g)$ that depend only on observables (such as $f(g) = E[X_i | G_i = g]$), and then proving that they are also sufficient representations because they can be written as invertible functions of $\psi(g)$.

3 Categorical variable encoding methods

Our methods proposed below take the form of removing the categorical column and replacing it with a set of columns that can be proven to encode all the categorical information. Each method exploiting the structure mentioned in the previous section. For an overview of other categorical encoding methods already in use, please see section 6.2 in the Appendix.

3.1 Means encoding

For our first method, we drop the categorical variables G_i and substitute in the average value of the continuous regressors X_i given the categorical variable. Figure 2 shows an illustration.



G	X_1	X_2	X_3
A	0.95	0.53	0.20
A	0.86	0.71	0.64
A	0.98	0.14	0.05
B	0.28	0.91	0.82
B	0.54	0.58	0.27
B	0.18	0.69	0.40
···	···	···	···

Original data set

	$\bar{X}_1^g$	$\bar{X}_2^g$	$\bar{X}_3^g$
A	0.93	0.46	0.3
B	0.34	0.72	0.5
	...	...	...

Group-wise means

X_1	X_2	X_3	$\bar{X}_1^g$	$\bar{X}_2^g$	$\bar{X}_3^g$
0.95	0.53	0.20	0.93	0.46	0.3
0.86	0.71	0.64	0.93	0.46	0.3
0.98	0.14	0.05	0.93	0.46	0.3
0.28	0.91	0.82	0.34	0.72	0.5
0.54	0.58	0.27	0.34	0.72	0.5
0.18	0.69	0.40	0.34	0.72	0.5
···	···	···	···	···	···

Resulting data set

Figure 2: Implementation example of the *means* encoding.[3]

[3] In Figure 2 and subsequent Figures 4 and 5, for easy interpretability, we show the Mxp matrix $\widehat{\Omega}^T$. The reason being that the ultimate implementation of $\psi(g)$ in statistical software requires appending the group encoding to the row the original categorical variable corresponded to.

This representation is easily interpretable, and it is simple to implement efficiently. This method may be particularly applicable in instances where the number of regressors p is small relative to the number of categories $p \ll |\mathcal{G}|$, since then the dimensionality reduction is more dramatic as compared to traditional encoding methods such as one-hot encoding. Figure 3 provides an intuitive explanation for why we should expect this to work: the group-wise averages of the continuous variables (X_1, X_2) may reveal the dominant latent group in each category.

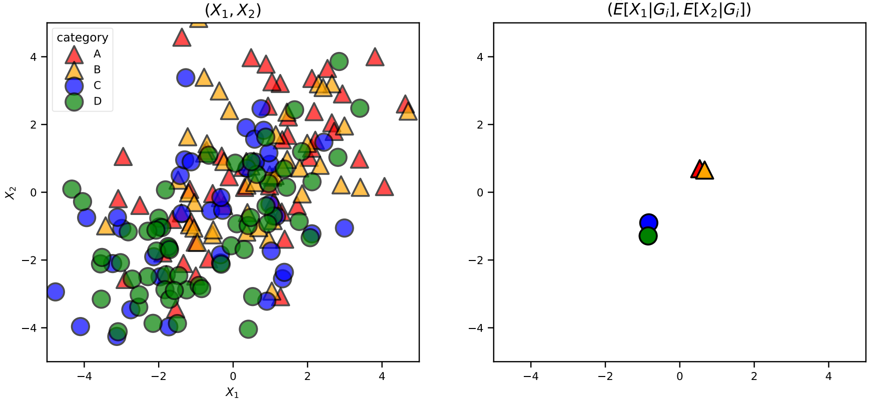


Figure 3: Intution for the *means* encoding on illustrative data. Here, categories (A, B) and (C, D) are associated with separate latent groups.

The next lemma presents the conditions in which this representation provides a sufficient representation. All proofs are in the appendix.

Lemma 2. *Under the conditions of Lemma 1, suppose in addition that the matrix A defined by $(A)_{tj} := \mathbb{E}\left[X_{it} \,\middle|\, L_i = l\right]$ is left-invertible. Then, the p-dimensional vectors $\omega(g) := \mathbb{E}\left[X_i \,\middle|\, G_i = g\right]$ are sufficient representations of each category in the sense of (3):*

$$\mu(x,\, g) = \frac{\sum_{l=1}^{k} \mathbb{E}\left[Y_i \,\middle|\, X_i = x,\, L_i = l\right] \mathbb{P}\left[X_i = x \,\middle|\, L_i = l\right] (A^{\dagger}\omega(g))_l}{\sum_{l=1}^{k} \mathbb{P}\left[X_i = x \,\middle|\, L_i = l\right] (A^{\dagger}\omega(g))_l} \tag{6}$$

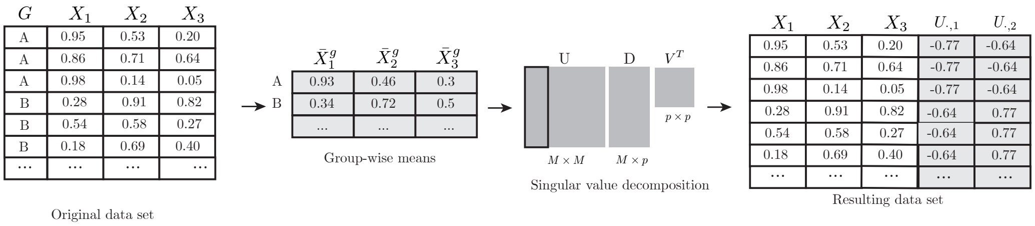
3.2 Low-rank encodings

The *means* encoding method may efficiently summarize the effect of the categorical variables if the continuous covariates are reasonably low-dimensional so that $p \ll M$. When p is large, it might be beneficial to use a lower-dimensional representation of the conditional means. We suggest two *low-rank encoding methods*, both involving matrix factorization of the transpose of our group-wise means matrix Ω of the continuous covariates where $(\Omega)_{jg} = E[X_{ij}|G = g]$.

Algorithm 1 Means Encoding Method

```
procedure GroupAverages($X, G$)
    $\widehat{\Omega} \leftarrow 0_{p \times M}$        ▷ Compute group-wise averages of continuous covariates
    for $g$ in 1:$M$ do
        $\widehat{\Omega}_{\cdot,g} \leftarrow \frac{1}{|\{i:G_i=g\}|} \sum_{i:G_i=g} X_i$
    return $\widehat{\Omega}$

procedure MeansEncoding($X, G$)
    $\widehat{\Omega} \leftarrow$ GroupAverages(X, G)
    $S \leftarrow 0_{n \times p}$
    for $i$ in 1:$n$ do                                  ▷ Populate with group averages
        $S_{i,\cdot} \leftarrow \widehat{\Omega}_{\cdot,G_i}$
    return $S$
```



G	X_1	X_2	X_3
A	0.95	0.53	0.20
A	0.86	0.71	0.64
A	0.98	0.14	0.05
B	0.28	0.91	0.82
B	0.54	0.58	0.27
B	0.18	0.69	0.40
···	···	···	···

Original data set

	$\bar{X}_1^g$	$\bar{X}_2^g$	$\bar{X}_3^g$
A	0.93	0.46	0.3
B	0.34	0.72	0.5
	...	...	...

Group-wise means

Singular value decomposition

X_1	X_2	X_3	$U_{\cdot,1}$	$U_{\cdot,2}$
0.95	0.53	0.20	-0.77	-0.64
0.86	0.71	0.64	-0.77	-0.64
0.98	0.14	0.05	-0.77	-0.64
0.28	0.91	0.82	-0.64	0.77
0.54	0.58	0.27	-0.64	0.77
0.18	0.69	0.40	-0.64	0.77
···	···	···	···	···

Resulting data set

Figure 4: Implementation example of the *low-rank* encoding with singular value decomposition. Alternatively, we could also have used sparse PCA in place of SVD.

One alternative is to consider the factorization of the transpose of our $p \times M$ group-wise means matrix Ω using singular value decomposition $\Omega^T = UDV^T$. Then we can use the first k columns of the g^{th} row of the left-singular vector matrix U as the representation for the g^{th} category. Note that in practice, we will be working with the empirical counterpart $\widehat{\Omega}$ and k is in general unknown, so we recommend using cross-validation. Figure 4 provides an illustration.

A second low-rank alternative is to use sparse principal component analysis (SPCA) method Zou et al. [2006] instead of SVD. As the name suggests, this method extends the original PCA algorithm by applying an elastic-net-style penalty on the coefficients of the loadings matrix. The result is that the matrix Ω^T is approximated by a sparse linear combination of vectors.[4] This sort of sparsity can be advantageous for two reasons. First, sparse PCA creates principal component sparse vectors which can be potentially more interpretable. Second, if our universal estimator is a tree-based model such as random forest Breiman [2001] or xgboost Chen and Guestrin [2016], which fit to data by considering sin-

[4] Formally, for a given matrix M, the sparse PCA method solves the problem [Zou et al., 2006, eq. 3.12],

$$(\hat{A}, \hat{B}) = \arg\min_{A,B} \sum_{i=1}^{n} ||M_i - AB^T M_i||^2 + \lambda \sum_{j=1}^{k} ||B_{\cdot,j}||_2^2 + \sum_{j=1}^{k} \lambda_{1,j} ||B_{\cdot,j}||_1 \tag{7}$$

$$\text{s.t.} \quad A^T A = I_{k \times k} \tag{8}$$

gular covariates at any point in the model, it may have difficulty taking advantage of dense principal components due to the rotation of the original covariate space. That is, if a tree would have a been able to produce a good split by using each variable separately, it may not be able to do the same if the variables are combined linearly. On the other hand, sparse PCA requires additional tuning of its regularization parameter λ which can be done via cross-validation.

Lemma 3 shows that if indeed there are only k latent groups, then a representation that uses only the first k columns of the left singular matrix or the first k sparse principal components is indeed sufficient. As with the *means* method, we rely on the universal consistency property of our estimator to learn a nonlinear mapping.

Lemma 3. *Under the conditions of Lemma 1, suppose in addition that the matrix A defined by $(A)_{tj} := \mathbb{E}\left[X_{it} \,\middle|\, L_i = g\right]$ is left-invertible. Then, the k-dimensional vectors $u(g) := U_{g,1:k}$ for are sufficient representations of each category in the sense of (3):*

$$\mu(x,\, g) = \frac{\sum_{l=1}^{k} \mathbb{E}\left[Y_i \,\middle|\, X_i = x,\, L_i = l\right] \mathbb{P}\left[X_i = x \,\middle|\, L_i = l\right] (A^{\dagger} V D u(g)^T)_l}{\sum_{l=1}^{k} \mathbb{P}\left[X_i = x \,\middle|\, L_i = l\right] (A^{\dagger} V D u(g)^T)_l} \tag{9}$$

Algorithm 2 Low Rank Encoding Method

```
procedure LowRankEncoding($X, G, k$)
    $\widehat{\Omega} \leftarrow$ GroupAverages(X, G)
    $U, D, V^T \leftarrow SVD(\widehat{\Omega}^T)$                 ▷ Singular value decomposition
    $S \leftarrow 0_{n \times k}$
    for $i$ in 1:$n$ do          ▷ Populate with left singular matrix truncated rows
        $S_{i,\cdot} \leftarrow U_{G_i,1:k}$
    return $S$
```

Algorithm 3 Sparse Low Rank Encoding Method

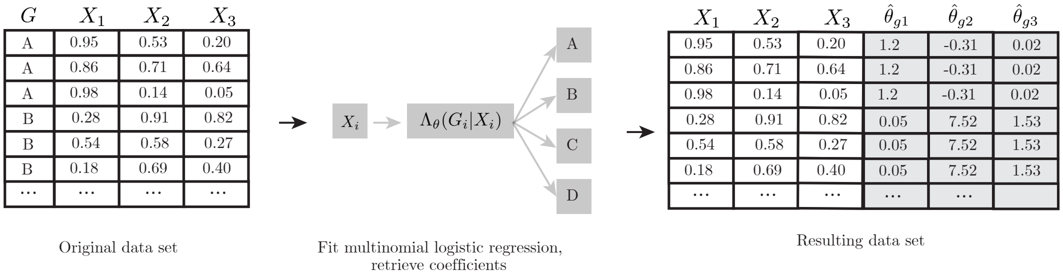
```
procedure SparseLowRankEncoding($X, G, k$)
    $\widehat{\Omega} \leftarrow$ GroupAverages(X, G)
    $A, B \leftarrow SPCA(\widehat{\Omega}^T)$            ▷ Sparse principal component analysis
    $Z \leftarrow \widehat{\Omega}^T \cdot B_{\cdot,1:k}$          ▷ Projection on truncated principal components
    $S \leftarrow 0_{n \times k}$
    for $i$ in 1:$n$ do           ▷ Populate with sparse principal components rows
        $S_{i,\cdot} \leftarrow Z_{G_i,\cdot}$
    return $S$
```

3.3 Encoding by multinomial logistic regression coefficients

Finally, we propose estimating the conditional probability of category membership by multinomial logistic regression parametrized by coefficients $\{\theta_g\}_{g \in \mathcal{G}}$

$$P(G_i|X_i) = \Lambda_\theta(G_i = g|X_i) = \frac{\exp(X_i^T \theta_g)}{\sum_{g'} \exp(X_i^T \theta_{g'})} \tag{10}$$

and then use the p-dimensional vector of coefficients θ_g associated with the g^{th} category to represent it. The motivation for this method comes from the fact that the prediction model $\mu(x, g)$ can be rewritten so that it only depends on the category g through $P(G_i = g|X_i = x)$, and under the multinomial logistic regression assumption above this boils down to dependence on the θ_g coefficients.



G	X_1	X_2	X_3
A	0.95	0.53	0.20
A	0.86	0.71	0.64
A	0.98	0.14	0.05
B	0.28	0.91	0.82
B	0.54	0.58	0.27
B	0.18	0.69	0.40
···	···	···	···

Original data set

Fit multinomial logistic regression, retrieve coefficients

X_1	X_2	X_3	$\hat{\theta}_{g1}$	$\hat{\theta}_{g2}$	$\hat{\theta}_{g3}$
0.95	0.53	0.20	1.2	-0.31	0.02
0.86	0.71	0.64	1.2	-0.31	0.02
0.98	0.14	0.05	1.2	-0.31	0.02
0.28	0.91	0.82	0.05	7.52	1.53
0.54	0.58	0.27	0.05	7.52	1.53
0.18	0.69	0.40	0.05	7.52	1.53
···	···	···	···	···	

Resulting data set

Figure 5: Implementation example of the *mnl* encoding.

A related work that also uses regression coefficients as categorical representations is the natural language processing *word2vec* model of Mikolov et al. [2013]. The authors of *word2vec* propose two methods to represent words (categories) in a large corpus of text as relatively low-dimensional real-valued vectors. In one of these methods, each word is initially assigned two representations: as a center word v_w, and as a surrounding *context* word v_c. Then, the authors posit that the optimal representation is the one that maximizes the log-probability of the inner product of the two representations $v_w^T v_c$ for all pair of words (w, c) that co-occur near each other. Our method works in an analogous way if we let the continuous vectors X_i stand in as "contexts", and let $\theta_g \in \mathbb{R}^p$ represent each category, since then maximizing the log-probability of the inner product $X_i^T \theta_g$ is the same as maximizing the multinomial logistic regression above.

Lemma 4. *Under the conditions of Lemma 1, suppose in addition that A in (25) is left-invertible, and that $\mathbb{P}\left[G_i = g \,\middle|\, X_i\right]$ is the multinomial logit distribution with coefficients $\{\theta_g\}_{g \in \mathcal{G}}$ containing an intercept. Then, the vector $\theta_g \in \mathbb{R}^p$ is sufficient in the sense of (3):*

$$\mu(x,\, g) = \frac{\sum_{l=1}^{k} \mathbb{E}\left[Y_i \,\middle|\, X_i = x,\, L_i = l\right] \mathbb{P}\left[X_i = x \,\middle|\, L_i = l\right] (A^\dagger f(\theta_g))_l}{\sum_{l=1}^{k} \mathbb{P}\left[X_i = x \,\middle|\, L_i = l\right] (A^\dagger f(\theta_g))_l} \tag{11}$$

$$\textit{where} \quad f(\theta_g) := \frac{\mathbb{E}_X\left[X_i \Lambda_\theta(g|X_i)\right]}{\mathbb{E}_X\left[\Lambda_\theta(g|X_i)\right]} \tag{12}$$

Algorithm 4 Multinomial logistic regression method (MNL)

```
procedure MNL($X, G$)
    $\hat{\theta} \leftarrow \arg\min_\theta \sum_i \log \Lambda_\theta(G_i|X_i)$        ▷ Multinomial logistic regression
    $S \leftarrow 0_{n \times p}$
    for $i$ in 1:$n$ do        ▷ Populate with left singular matrix truncated rows
        $S_{i,\cdot} \leftarrow \hat{\theta}_{G_i}$
    return $S$
```

4 Experiments

In the following section, we explore each method's effectiveness relative to one hot encoding across simulated and real world data sets. We apply two typically used methods random forests and xgboost.[5]

4.1 Simulations

We consider two simulations designs that share the distributions of latent groups L_i, observable groups G_i and covariates X_i, but whose outcome models for Y_i differ.

Latent groups, observable groups and continuous covariates A latent group L_i is drawn uniformly from the set of available groups, which we identify with integers.

$$L_i \sim \text{Uniform}(\{1, ..., |\mathcal{L}|\}) \tag{13}$$

Next, observable groups G_i are drawn according to the following rule. First, we partition the set of possible observable groups $\mathbb{G}$ into equally-sized sets $\{\mathbb{G}_\ell\}_{\ell=1}^{|\mathcal{L}|}$. Then, we draw the observable group G_i so that observations that were assigned latent group L_1 have higher probability of falling into observable group $\mathbb{G}_1$, those in L_2 likely belong to $\mathbb{G}_2$, and so on. In symbols,

$$P(G_i = g \mid L_i) = \begin{cases} \frac{p_{L_i}}{|\mathbb{G}_{L_i}|} & \text{if} \quad g \in \mathbb{G}_{L_i} \\ \frac{1-p_{L_i}}{|\mathbb{G}^C_{L_i}|} & \text{otherwise} \end{cases} \qquad \text{where} \quad p_{L_i} > 0.5 \tag{14}$$

Covariates associated with latent group $L_i = \ell$ are normally distributed as $X_i \sim \mathcal{N}(\mu_\ell, \Sigma)$. The mean is zero except for a randomly drawn set of entries $\mathcal{J}$ that are $-1, +1$, with $|\mathcal{J}| = 3$.

$$(\mu_\ell)_j = \begin{cases} 0 & \text{if } j \in \mathcal{J} \\ \text{Uniform}(\{-1, 1\}) & \text{otherwise} \end{cases} \qquad (\Sigma)_{kj} = \left(\frac{1}{2}\right)^{|k-j|} \tag{15}$$

Outcomes For the outcome setups, we make each scenario noticeably more complex than the last. In the *global linear* setup, each latent group has its own intercept while the slope β is the same across latent groups.

$$Y_i = \alpha_\ell + X_i^T \beta + \epsilon_i \tag{16}$$

where the intercept and slopes are created as follows. The slope normalization ensures that the signal from the intercept, regressors and noise is roughly comparable.

$$\alpha_\ell \sim Laplace(1) \quad \text{for each } \ell \in \mathcal{L} \tag{17}$$

$$\tilde{\beta}_j \sim \text{Uniform}(\{0, 1, -1\}) \qquad \beta = \frac{\tilde{\beta}}{\left\|\tilde{\beta}\right\|_2} \tag{18}$$

$$\epsilon_i \sim \mathcal{N}(0, 1) \tag{19}$$

[5] Simulation code can be found at: `repo url`.

In the *latent linear* setup, we increase the dependence on the latent groups and the outcome model is linear in regressors conditional on coefficients that are specific to each latent group.

$$Y_i = \alpha_\ell + X_i^T \beta_\ell + \epsilon_i \tag{20}$$

where

$$\tilde{\beta}_{\ell j} \sim \text{Uniform}(\{0, 1, -1\}) \qquad \beta_\ell = \frac{\tilde{\beta}_\ell}{\left\| \tilde{\beta}_\ell \right\|_2} \tag{21}$$

Finally, in the *latent piecewise linear* setup, we compute a dyadic basis by partitioning each feature X_i by its median and then assigning different latent group betas (β_l^+ or β_l^-) depending on whether or not the observed X_i is above or below its feature's median.

$$Y_i = \alpha_\ell + \sum_{j=1}^{p} \mathbf{1}\{X_{ij} > \text{Med}(x_j)\} \cdot X_{ij}^T \beta_{\ell j}^+ + \mathbf{1}\{X_{ij} \leq \text{Med}(x_j)\} \cdot X_{ij}^T \beta_{\ell j}^- + \epsilon_i \tag{22}$$

where

$$\tilde{\beta}_{\ell j}^+, \tilde{\beta}_{\ell j}^- \sim \text{Uniform}(\{0, 1, -1\}) \qquad \beta_\ell^+ = \frac{\tilde{\beta}_\ell^+}{\left\| \tilde{\beta}_\ell^+ \right\|_2} \qquad \beta_\ell^- = \frac{\tilde{\beta}_\ell^-}{\left\| \tilde{\beta}_\ell^- \right\|_2} \tag{23}$$

4.2 Simulation Results

For each simulated dataset, we estimated the outcome using the various methods described in Section 3, and then evaluated the predictions by their mean squared error. We simulated each simulation setup and model for 200 randomly generated seeds.

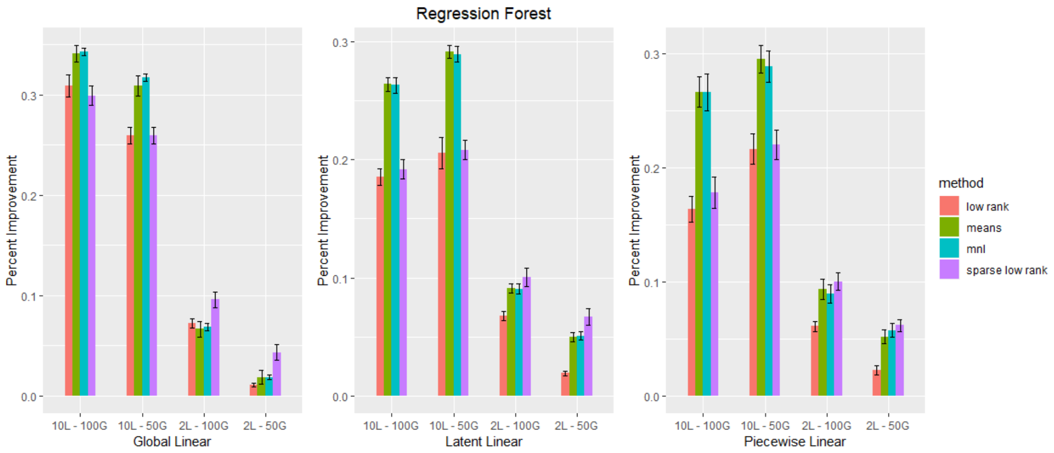


Figure 6: Percent Improvement over One Hot Encoding for Regression Forests.

Results for the simulations are provided in 6. We find that the methods described above which seek to estimate the latent groups consistently outperform methods which require adding $|\mathcal{G}|$ additional columns to our input matrix X for both the regression forest and xgboost. In particular, it appears that the sparse low rank approach tends to do well when the number of latent groups is very small. For a larger number of latent groups, we see that low rank approaches underperform and the multinomial and means encoding perform better. We also take note on how the multinomial weight approach potentially does well in this case possibly because n is large and the number of observations per group is high enough to satisfy this approach.

For the methods that do not take advantage of the low rank structure, we notice that the main improvement in performance for regression forests and xgboost occurs due to the reduction in dimensionality. We find that the permutation, fisher, and multiple permutation methods are on average much better than the methods that add $|\mathcal{G}|$ columns but still underperform relative to the methods that estimate the latent groups.

While the performance improvements over one hot encoding for 2 latent groups ranges from 1-10%, performance improvement can approach 27-33% for 10 latent groups. Intuitively, we find that this benefit is generally less prevalent for 2 latent groups for regression forests and xgboost due to the lesser complexity of the underlying relationship as defined by the conditional independence graph in 1. The improvement over one hot encoding also tends to increase as the signal becomes more dependent on latent group membership. In most cases, performance on the piecewise linear simulations maintain the same or higher percent improvement over one hot encoding. Furthermore, we find that a more complex and nonlinear method like xgboost benefits slightly less from these encoding methods.

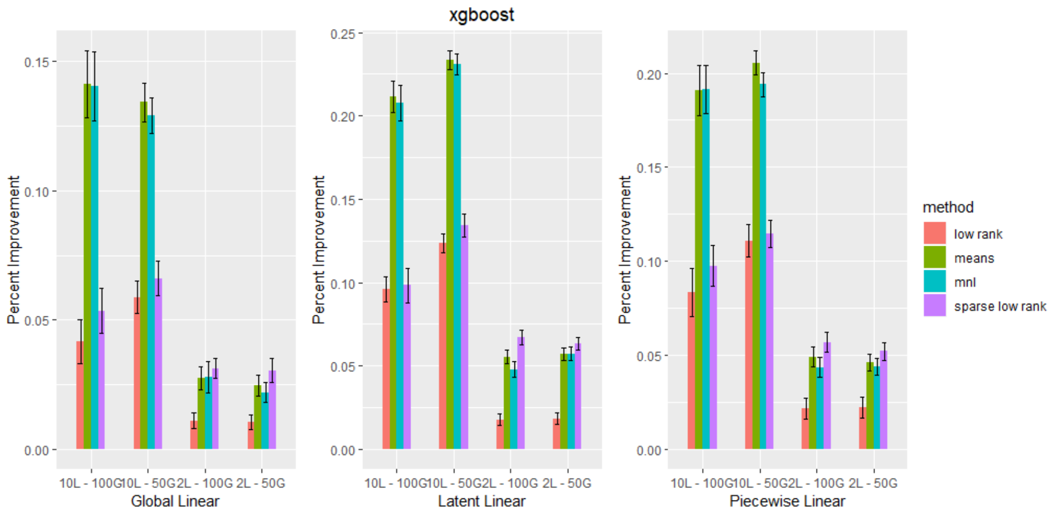


Figure 7: Percent Improvement over One Hot Encoding for xgboost.

4.3 Empirical Applications

We also evaluate these methods on publicly available datasets that are accessible on Kaggle. We run 4 fold, stratified cross validation on the datasets to avoid the case where there are categorical variables in the test set which are not contained in the training set. Since we are throwing out what could potentially be a sizable amount of information with each fold, we also conduct a paired t-test to further validate or deny results seen in the cross validation process.

Pakistan Educational Performance In Hemani [2017], Hemani consolidated a series of surveys from Alif Ailaan, a nonprofit organization in Pakistan that focuses on improving education across the country. The objective of the surveys was to provide an objective means of comparing school systems across cities and provinces in order to spark competition between local governments to spur educational reform.

The dataset used in the analysis below contains $n = 580$ and 504 after removing null valued rows which can be broken down into $|\mathcal{G}| = 127$ cities from 2013 to 2016. The number of additional covariates p is 20 and our estimation methodology is further elaborated on in Appendix 6.3.

Ames Housing The objective of the Ames Housing Dataset De Cock [2011] was to act as a more complex alternative to the Boston Housing Dataset Harrison and Rubinfeld [1978]. De Cock's aim was to use this dataset for the final project in his regression course which would allow students to more extensively showcase what they had learned.

The Ames Housing Dataset has $n = 2,930$ individual home sales in Ames, Iowa from 2006 to 2010. The dataset has 80 covariates and our categorical variable "neighborhood" has $|\mathcal{G}| = 25$.

King County House Sales The King County House Sales Dataset from harlfoxem [2016] is a data set containing the record of 21,613 home sales in King County, Washington between May 2014 to May 2015. The author does not provide much additional information aside from it being a good dataset to test regression models. The dataset is relatively popular with over 169,000 views and 28,000 downloads at the time of this paper.

The data itself came with 21 covariates including the sale price of the house. We treat the "zipcode" covariate, which has $|\mathcal{G}| = 70$, as the categorical variable.

4.4 Empirical Results

We can see that for Regression Forests on average there is an improvement over one hot encoding and xgboost stands to benefit less from using these encoding methods over one hot encoding. For regression forests, the primary case that does not benefit much from these approaches is the Ames data set which follows naturally since the number of covariates p is much larger than the number of observed groups $|\mathcal{G}|$. Therefore, methods such as means and MNL are adding 80 dimensions to the prediction problem while one hot encoding only adds 25. The Low Rank and Sparse Low Rank approaches benefit in these cases and appear to maintain potentially promising results. Contrary to the regression forest results, most of the xgboost output was not statistically significantly different than one hot encoding and the Ames data set was the closest evidence to any benefit.

Dataset	Metric	Means	Low Rank	Sparse Low Rank	MNL
Pakistan	MSE	9.963	8.228	8.868	8.656
Pakistan	p-val	0.00402	0.04333	0.00089	0.01132
Ames	MSE	1.349	1.798	3.987	-2.120
Ames	p-val	0.73221	0.00930	0.06932	0.81650
Kingcounty	MSE	8.405	8.671	7.062	8.054
Kingcounty	p-val	0.00445	0.01267	0.03102	0.00364

Table 1: Observational Dataset Results for Regression Forests.

For regression forests, on average, it looks like low rank approaches to generating encodings were most robust across data sets. This could be due to the reduction in dimensionality which may be beneficial for two reasons. First, the underlying relationships were much lower rank than the number of covariates and these methods were able to capture this information. Second, if there was no signal in the categorical variable to begin with, the low rank approaches which utilize K-fold to determine the dimensionality of the encoding are able to pick small k number of encoding vectors to reduce the potential noise covariates one would be adding.

Dataset	Metric	Means	Low Rank	Sparse Low Rank	MNL
Pakistan	MSE	2.904	0.668	2.528	-3.391
Pakistan	p-val	0.52714	0.88127	0.29955	0.23304
Ames	MSE	7.382	9.736	14.889	1.890
Ames	p-val	0.31597	0.07341	0.13348	0.59210
Kingcounty	MSE	0.773	-3.243	2.468	-0.471
Kingcounty	p-val	0.67990	0.62293	0.49389	0.87640

Table 2: Observational Dataset Results for XGBoost.

5 Conclusion

In this paper, we explore the task of mapping high-cardinality categorical variables G_i to a lower-dimensional real space without loss of information relevant to our response Y_i. To do this, we make an assumption about the relationship between G_i and Y_i which we call the *sufficient latent state assumption*. This assumption provides us with the basis for creating encoding methods which can be used by universally consistent estimators to extract sufficient representations of G_i. Among our recommendations for encoding methods, we provide encoding methods which are interpretable or focus more on reducing the size of the $\mathbb{R}^k$ representation. We find that these methods tend to outperform one hot encoding and other traditional approaches to modeling with categorical variables as the number of unique categories increases.

6 Appendix

6.1 Proofs

Definitions The following matrices that will be used below.

$$\Omega = \begin{bmatrix} \mathbb{E}\left[X_1 \,\middle|\, G = g_1\right] & \cdots & \mathbb{E}\left[X_1 \,\middle|\, G = g_M\right] \\ \vdots & \ddots & \\ \mathbb{E}\left[X_p \,\middle|\, G = g_1\right] & \cdots & \mathbb{E}\left[X_p \,\middle|\, G = g_M\right] \end{bmatrix}_{p \times M} \tag{24}$$

$$A = \begin{bmatrix} \mathbb{E}\left[X_1 \,\middle|\, L = l_1\right] & \cdots & \mathbb{E}\left[X_1 \,\middle|\, L = l_K\right] \\ \vdots & \ddots & \\ \mathbb{E}\left[X_p \,\middle|\, L = l_1\right] & \cdots & \mathbb{E}\left[X_p \,\middle|\, L = l_K\right] \end{bmatrix}_{p \times K} \tag{25}$$

$$\Psi = \begin{bmatrix} \mathbb{P}\left[L = l_1 \,\middle|\, G = g_1\right] & \cdots & \mathbb{P}\left[L = l_1 \,\middle|\, G = g_M\right] \\ \vdots & \ddots & \\ \mathbb{P}\left[L = l_K \,\middle|\, G = g_1\right] & \cdots & \mathbb{P}\left[L = l_K \,\middle|\, G = g_M\right] \end{bmatrix}_{K \times M} \tag{26}$$

We denote the columns of Ω as $\omega(g)$ and the columns of Ψ as $\psi(g)$.

Overview Proof 6.1.1 shows that categories G_i only enter the conditional expectation function $\mu(x, g)$ through the latent state probabilities $\psi(g)$. Proofs 6.1.2-6.1.4 rely on strategies for writing $\psi(g) = f(h(g))$ then showing that $h(g)$ is also a sufficient representation.

6.1.1 Proof of Lemma 1

Proof. To show the equivalence of (1) and (5), we begin by expanding (1) as

$$\mu(x,\, g) = \sum_{l=1}^{L} \mathbb{E}\left[Y_i \,\middle|\, X_i = x,\ G_i = g,\ L_i = l\right] \mathbb{P}\left[L_i = l \,\middle|\, X_i = x,\, G_i = g\right] \tag{27}$$

Now, the conditional independence assumptions encoded in our graph imply that the expectation term simplifies to

$$\mathbb{E}\left[Y_i \,\middle|\, X_i = x,\ G_i = g,\ L_i = l\right] = \mathbb{E}\left[Y_i \,\middle|\, X_i = x,\ L_i = l\right] \tag{28}$$

while the second term can be rewritten using Bayes rule as

$$\mathbb{P}\left[L_i = l \,\middle|\, X_i = x,\, G_i = g\right] = \frac{\mathbb{P}\left[X_i = x \,\middle|\, L_i = l\right] \mathbb{P}\left[L = l \,\middle|\, G_i = g\right]}{\sum_{l'=1}^{k} \mathbb{P}\left[X_i = x \,\middle|\, L_i = l'\right] \mathbb{P}\left[L = l' | G_i = g\right]} \tag{29}$$

Combining the above, we see that the mapping μ only depends on the categorical variable through the multivariable function $\psi(g) = \mathbb{P}\left[L_i | G_i = g\right]$. Therefore, $\psi(g)$ is a sufficient representation as defined in (3). □

6.1.2 Proof of Lemma 2

Proof. Begin by noting that conditioanl expectations can be computed as a linear combination of the sufficient statistics discussed in Lemma 1.

$$\mathbb{E}\left[X_i \,\middle|\, G_i = g\right] = \sum_{l=1}^{K} \mathbb{E}\left[X_i \,\middle|\, L_i = l\right] \mathbb{P}\left[L_i = l \,\middle|\, G_i = g\right] \tag{30}$$

$$= \sum_{l=1}^{K} \mathbb{E}\left[X_i \,\middle|\, L_i = l\right] \psi_l(g) \tag{31}$$

or, in matrix form,

$$\Omega = A\Psi \tag{32}$$

where these matrices are defined as in the top of this section. The sufficient representation for the category $\psi(g) = \Psi_g$ lies on the linear span of the set of columns of A. Since A has a left-inverse $A^\dagger$ such that $A^\dagger A = I$, we can retrieve the representations by matrix multiplication.

$$\psi(g) = (\Psi)_{\cdot,g} = A^\dagger(\Omega)_{\cdot,g} =: A^\dagger \omega(g) \tag{33}$$

Since $\psi(g)$ only depends on g through $\omega(g)$, it follows that $\omega(g)$ is also a sufficient representation for the category g. □

6.1.3 Proof of Lemma 3

Proof. The proof is similar to the the previous one. However, this time note that we can decompose the $\Omega^T = UDV^T$ using singular value decomposition, where the matrices have dimensions $|\mathcal{G}| \times |\mathcal{G}|$, $|G| \times p$, and $p \times p$ respectively. Letting $u(g) : g \mapsto (U)_{g,\cdot}$, we can write

$$\psi(g) = A^\dagger V D u(g)^T \tag{34}$$

where D and V do not depend on the category g and, to complete the proof, we substitute $VDu(g)^T$ with $\omega(g)$ in 33.

□

6.1.4 Proof of Lemma 4

Proof. We begin by noting that we can use Bayes' theorem to express $\omega(g) = \mathbb{E}\left[X_i \,\middle|\, G = g\right]$ as a function of $\mathbb{P}\left[G = g \,\middle|\, X_i\right]$, here is assumed to be multinomial logit.

$$\mathbb{E}_{X|G}\left[X_i \,\middle|\, G_i = g\right] = \mathbb{E}_X\left[X_i \mathbb{P}\left[X_i | G_i = g\right]\right] \tag{35}$$

$$= \frac{\mathbb{E}_X\left[X_i \mathbb{P}\left[G_i = g | X_i\right]\right]}{\mathbb{P}\left[G_i = g\right]} \tag{36}$$

$$= \frac{\mathbb{E}_X\left[X_i \mathbb{P}\left[G_i = g | X_i\right]\right]}{\mathbb{E}_X\left[\mathbb{P}\left[G_i = g \,\middle|\, X_i\right]\right]} \tag{37}$$

$$= \frac{\mathbb{E}_X\left[X_i \Lambda_\theta(g | X_i\right]}{\mathbb{E}_X\left[\Lambda_\theta(g|X_i)\right]} \tag{38}$$

However, note that expression (38) only depends on the category through the mutinomial logit coefficients θ_g that are associate with category g. Therefore, under this assumption we can write $\omega(g) = f(\theta_g) =: E[X_i|G = g]$. However, recall from (33) that if the matrix $(A)_{j\ell} := E[X_{ij}|L_i = \ell]$ has a left-inverse $A^\dagger A = I$, we can write

$$\psi(g) = A^\dagger \omega(g) = A^\dagger f(\theta_g) \tag{39}$$

Since $\psi(g)$ only depends on g through $\theta(g)$, it follows that $\theta(g)$ is also a sufficient representation for the category g. □

6.2 Additional Encoding Methods

For a more in-depth treatment, see Venables [2016]. Note that several of the methods below are simple linear transformations of each other and should yield equivalent levels of performance in theory. However, as we will see in sections 4.2 and 4.3, in practice the resulting performance can differ substantially.

One-hot or dummy This is the most common categorical encoding, and it is the method we take to be our main baseline, against which we will compare all other methods. It expands out the categorical column into $k-1$ columns where k is the number of unique elements in the set of categorical levels in the column. Each column is binary 1 or 0 depending on whether the corresponding level was observed in the original categorical column. [Murphy, 2012, sec 2.3.2]

	b	c	d	e
a	0	0	0	0
b	1	0	0	0
c	0	1	0	0
d	0	0	1	0
e	0	0	0	1

Deviation Similar to one-hot encoding except that the k^{th} unique element's row that is the reference level is now set to all values of -1. This means that categorical levels are being compared to the grand mean of all of the levels instead of the mean of a given level with respect to the reference level.

	b	c	d	e
a	1	0	0	0
b	0	1	0	0
c	0	0	1	0
d	0	0	0	1
e	-1	-1	-1	-1

Difference Compares a given level to the mean of the levels that precede it.

	b	c	d	e
a	-0.5	-0.333	-0.25	-0.2
b	0.5	-0.333	-0.25	-0.2
c	0.0	0.667	-0.25	-0.2
d	0.0	0.000	0.75	-0.2
e	0.0	0.000	0.00	0.8

Helmert Compares levels of a chosen categorical variable to the mean of the subsequent levels uniquely observed thus far.

	b	c	d	e
a	0.80	0.00	0.00	0.00
b	-0.20	0.75	0.00	0.00
c	-0.20	-0.25	0.67	0.00
d	-0.20	-0.25	-0.33	0.50
e	-0.20	-0.25	-0.33	-0.50

Repeated Effect Columns are encoded to represent a cumulative comparison of subsequent levels with previous ones.

	b	c	d	e
a	0.8	0.6	0.4	0.2
b	-0.2	0.6	0.4	0.2
c	-0.2	-0.4	0.4	0.2
d	-0.2	-0.4	-0.6	0.2
e	-0.2	-0.4	-0.6	-0.8

Permutation Assigns a unique integer to each category. Note that even when the categories do not possess an intrinsic ordering, some mappings may yield better results if they happen to be aligned with the true average effect the category has on the outcome variable.

	perm
a	5
b	3
c	4
d	1
e	2

Multi-Permutation (Multi-Perm) Following the intuition above, with a larger number of columns we might find more interesting permutations. Hence, we also experiment with four random integer mappings at once.

	perm1	perm2	perm3	perm4
a	1	5	4	2
b	2	3	5	3
c	3	1	2	4
d	4	4	1	1
e	5	2	3	5

Fisher taken from Hastie et al. [2009], we order the categories by increasing mean of the response.

For the following five methods, we use information about the continuous covariates to construct the mapping ψ.

6.3 Estimation details

Below we provide additional details to better clarify how new methods related to Means were estimated and remove basic problems in the data that complicate training models. For PCA Means, we select the first k principal components which generate 95% of the variance in X, group by the unique categorical levels, and take the means of those principal components $z_{1:k}$. For SPCA Means, we use the default hyperparameters in the "sparsepca" R package.

In the Ames dataset, we remove the features "PoolQC", "GarageQual", and "GarageYrBlt" due to almost being perfectly correlated with other features and remove the missing data rows as well. In the King County House Sales dataset, we remove the ID and date of the house sale as the covariates. Finally, in the Pakistan dataset, many of the covariates are almost perfectly correlated such as "% girls enrolled", "% boys enrolled", and "gender parity score." As a result, we removed such covariates resulting in the following list of covariates: Education score, Toilet, Province, Population, School infrastructure score, Total number of schools, Primary Schools with single teacher, Primary Schools with single classroom, Pakistan Economic Growth, Number of secondary schools, Electricity, No Facility, City, Global Terrorism Index - Pakistan, Complete Primary Schools, Building condition satisfactory, Drone attacks in Pakistan, Drinking water, Boundary wall, Bomb Blasts Occurred, % Complete Primary Schools, % Boys Enrolled.

References

Joshua D Angrist and Jörn-Steffen Pischke. *Mostly harmless econometrics: An empiricist's companion.* Princeton university press, 2008.

Dmitry Arkhangelsky and Guido Imbens. The role of the propensity score in fixed effect models. Technical report, National Bureau of Economic Research, 2018.

Gérard Biau, Luc Devroye, and Gábor Lugosi. Consistency of random forests and other averaging classifiers. *The Journal of Machine Learning Research*, 9:2015–2033, 2008.

Stéphane Bonhomme and Elena Manresa. Grouped patterns of heterogeneity in panel data. *Econometrica*, 83(3):1147–1184, 2015.

Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and Regression Trees*. CRC press, 1984.

Patricio Cerda, Gaël Varoquaux, and Balázs Kégl. Similarity encoding for learning with dirty categorical variables. *Machine Learning*, pages 1–18, 2018.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM, 2016.

Dean De Cock. Ames, iowa: Alternative to the boston housing data as an end of semester regression project. *Journal of Statistics Education*, 19(3), 2011.

Peter J Diggle, Patrick J Heagerty, Kung-Yee Liang, and Scott Zeger. *Analysis of longitudinal data*. Oxford University Press, 2002.

András Faragó and Gábor Lugosi. Strong universal consistency of neural network classifiers. *IEEE Transactions on Information Theory*, 39(4):1146–1151, 1993.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

harlfoxem. House sales in king county,usa, 2016. URL https://www.kaggle.com/harlfoxem/housesalesprediction/.

David Harrison, Jr and Daniel L Rubinfeld. Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5(1):81–102, 1978.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. New York: Springer, 2009.

Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press, 2015.

Mesum Raza Hemani. Pakistan education performance dataset, 2017. URL https://www.kaggle.com/mesumraza/pakistan-education-performance-dataset/.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations*, 2013.

Kevin Murphy. Machine learning, a probabilistic perspective, 2012.

Jerzy Neyman and Elizabeth L Scott. Consistent estimates based on partially consistent observations. *Econometrica: Journal of the Econometric Society*, pages 1–32, 1948.

Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in neural information processing systems*, pages 1177–1184, 2008.

Charles J Stone. Consistent nonparametric regression. *The Annals of Statistics*, pages 595–620, 1977.

WN Venables. codingmatrices: Alternative factor coding matrices for linear model formulae [software], 2016.

Jeffrey M Wooldridge. *Econometric analysis of cross section and panel data.* MIT press, 2010.

Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, pages 265–286, 2006.