Robust Variational Autoencoders for Outlier Detection in Mixed-Type Data

Citation for published version:

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.
Robust Variational Autoencoders for Outlier Detection and Repair of Mixed-Type Data

Simão Eduardo*1 Alfredo Nazábal*2 Christopher K. I. Williams12 Charles Sutton123
1School of Informatics, University of Edinburgh, UK
2The Alan Turing Institute, UK; 3Google Research

Abstract

We focus on the problem of unsupervised cell outlier detection and repair in mixed-type tabular data. Traditional methods are concerned only on detecting which rows in the dataset are outliers. However, identifying which cells corrupt a specific row is an important problem in practice, and the very first step towards repairing them. We introduce the Robust Variational Autoencoder (RVAE), a deep generative model that learns the joint distribution of the clean data while identifying the outlier cells, allowing their imputation (repair). RVAE explicitly learns the probability of each cell being an outlier, balancing different likelihood models in the row outlier score, making the method suitable for OD in mixed-type datasets. We show experimentally that not only RVAE performs better than several state-of-the-art methods in cell OD and repair for tabular data, but also that is robust against the initial hyper-parameter selection.

1 Introduction

The existence of outliers in real world data is a problem data scientists face daily, so outlier detection (OD) has been extensively studied in the literature (Chandola et al., 2009; Emmott et al., 2015; Hodge and Austin, 2004). The task is often unsupervised, meaning that we do not have annotations indicating whether individual cells in the data table are clean or anomalous. Although supervised OD algorithms have been proposed (Lee et al., 2018; An and Cho, 2015; Schlegl et al., 2017), annotations of anomalous cells are often not readily available in practice. Instead, unsupervised OD attempts to infer the underlying clean distribution, and explains outliers as instances that deviate from that distribution. It is important to focus on the joint distribution over features, because although some outliers can be easily identified as anomalous by considering only the marginal distribution of the feature, many others are only detectable within the context of the other features (section 2.2 of (Chandola et al., 2009)). Recently deep models outperformed traditional ones for tabular data tasks (Klambauer et al., 2017), capturing their underlying structure better. They are an attractive choice for OD, since they have the flexibility to model a wide variety of clean distributions. However OD work has mostly focused on image datasets, repairing dirty pixels instead of cells in tabular data, e.g. (Wang et al., 2017b; Zhou and Paffenroth, 2017; Akrami et al., 2019).

Outliers present unique challenges to deep generative models. First, most work focuses on detecting anomalous data rows, without detecting which specific cells in a row are problematic (Redyuk et al., 2019; Schelter et al., 2018). Work on cell-level detection and repair often focuses on real-valued features, e.g. images (Zhou and Paffenroth, 2017; Wang et al., 2017b; Schlegl et al., 2017), or does not provide a principled way to detect anomalous cells (Nguyen and Vien, 2018). Since focus is on row OD, not enough care is given to cell granularity, which means it is often difficult to properly repair the dirty cells, e.g. large number of columns exist or when the data scientist is not a domain expert. Second, tabular data is often mixed-type, including both continuous and categorical columns. Although modelling mixed-type data has been explored (Nazabal et al., 2018; Vergari et al., 2019), the difficulty arises when handling outliers. Standard outlier scores are based on the probability that the model assigns to a cell, but these values are not comparable between likelihood models, performing poorly for mixed-type data. Finally, the effect of outliers in unsupervised learning can be insidious. Since deep generative models are highly flexible,
they are not always robust against outliers [Hendrycks and Dietterich 2019], overfitting to anomalous cells. When the model overfits, it cannot identify these cells as outliers, because it has modelled them as part of the clean distribution, and consequently, most repair proposals are skewed towards the dirty values, and not the underlying clean ones.

Our main contributions are: (i) Our Robust Variational Autoencoder (RVAE), a novel fully unsupervised deep generative model for cell-level OD and repair for mixed-type tabular data. It uses a two-component mixture model for each feature, with one component for clean data, and the other component that robustifies the model by isolating outliers. (ii) RVAE models the underlying clean data distribution by down-weighting the impact of anomalous cells, providing a competitive outlier score for cells and a superior estimate of cell repairs. (iii) We present a hybrid inference scheme for optimizing the model parameters, combining amortized and exact variational updates, which proves superior standard amortized inference. (iv) RVAE allows us to present an outlier score that is commensurate across mixed-type data. (v) RVAE is robust against the selection of its hyper-parameters, while other OD methods suffer from fine tuning of their parameters to each specific dataset.

2 Variational Autoencoders

We consider a tabular dataset $X$ with $n \in \{1, \cdots, N\}$ instances and $d \in \{1, \cdots, D\}$ features, where each cell $x_{nd}$ in the dataset can be real (continuous), $x_{nd} \in \mathbb{R}$, or categorical, $x_{nd} \in \{1, \cdots, C_d\}$ with $C_d$ the number of unique categories of feature $d$. Cells in the dataset are potentially corrupted with an unknown noising process appropriate for the feature type. The objective in this work is not only detecting the anomalous instances in the dataset, termed row outliers, but also determining the specific subset of cells that are anomalous, termed cell outliers, proposing potential repair values for them.

A common approach to unsupervised OD is to build a generative model $p(X)$ that models the distribution of clean data. A powerful class of deep generative models are variational autoencoders (VAEs) [Kingma and Welling 2014], which model $p(X)$ as

$$p(X) = \prod_{n=1}^{N} \int d\mathbf{z}_n p(\mathbf{z}_n)p(\mathbf{x}_n|\mathbf{z}_n)$$  (1)

where $p(\mathbf{x}_n|\mathbf{z}_n) = \prod_{d=1}^{D} p(\mathbf{x}_{nd}|\mathbf{z}_n)$ and $p(\mathbf{x}_{nd}|\mathbf{z}_n)$ is the conditional likelihood of feature $d$, $\mathbf{z}_n \in \mathbb{R}^K$ is the latent representation of instance $\mathbf{x}_n$, and $p(\mathbf{z}_n) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ is an isotropic multivariate Gaussian prior. To handle mixed-type data, we choose the conditional like-

hood $p_{\theta}(x_{nd}|\mathbf{z}_n)$ differently for each feature type. For real features $p_{\theta}(x_{nd}|\mathbf{z}_n) = \mathcal{N}(x_{nd}|m_d(\mathbf{z}_n), \sigma_d)$, where $\sigma_d$ is a global parameter. For categorical features $p_{\theta}(x_{nd}|\mathbf{z}_n) = f(\mathbf{a}_d(\mathbf{z}_n))$, where $\mathbf{a}_d(\mathbf{z}_n)$ is an unnormalized vector of probabilities for each category and $f$ is the softmax function. All $m_d(\mathbf{z}_n)$ and $\mathbf{a}_d(\mathbf{z}_n)$ are parameterized by feed-forward networks.

As exact inference for $p_{\theta}(\mathbf{z}_n|\mathbf{x}_n)$ is generally intractable, a variational posterior $q_{\phi}(\mathbf{z}_n|\mathbf{x}_n)$ is used; in VAEs this is also known as the encoder. It is modelled by a Gaussian distribution with parameters $\mu(\mathbf{x}_n)$ and $\Sigma(\mathbf{x}_n)$

$$q_{\phi}(\mathbf{z}_n|\mathbf{x}_n) = \mathcal{N}(\mathbf{z}_n|\mu(\mathbf{x}_n), \Sigma(\mathbf{x}_n))$$  (2)

where $\phi = (\mu(\mathbf{x}_n), \Sigma(\mathbf{x}_n))$ are feed-forward neural networks, and $\Sigma(\mathbf{x}_n)$ is a diagonal covariance matrix. VAEs are trained by maximizing the lower bound on the marginal log-likelihood called the evidence lower bound (ELBO), given by

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} \sum_{d=1}^{D} \mathbb{E}_{q_{\phi}(\mathbf{z}_n|\mathbf{x}_n)}[\log p_{\theta}(x_{nd}|\mathbf{z}_n)] - D_{KL}(q_{\phi}(\mathbf{z}_n|\mathbf{x}_n)||p(\mathbf{z}_n)))$$  (3)

where the neural network parameters of the decoder $\theta$ and encoder $\phi$ are learnt with a gradient-based optimizer. When VAEs are used for OD, typically an instance in a tabular dataset is an outlier if the expected likelihood $\mathbb{E}_{q_{\phi}(\mathbf{z}_n|\mathbf{x}_n)}[\log p_{\theta}(\mathbf{x}_n|\mathbf{z}_n)]$ is small [An and Cho 2015; Wang et al. 2017].

3 Robust Variational Autoencoder (RVAE)

To improve VAEs for OD and repair, we want to make them more robust, by automatically identifying potential outliers during training, so they are downweighted when training the generative model. We also want a cell-level outlier score which is comparable across continuous and categorical attributes. We can achieve both goals by modifying the generative model.

We define here our robust variational autoencoder (RVAE), a deep generative model based on a two-component mixture model likelihood (decoder) per feature, which isolates the outliers during training. RVAE is composed of a clean component $p_{\theta}(x_{nd}|\mathbf{z}_n)$ for each dimension $d$, explaining the clean cells, and an outlier component $p_{\omega}(x_{nd})$, explaining the outlier cells. The mixing variable $w_{nd} \in \{0, 1\}$ acts as a gate to determine whether cell $x_{nd}$ should be modelled by the clean component ($w_{nd} = 1$) or the outlier component ($w_{nd} = 0$). We define the marginal likelihood of the mixture model
We see that with mixing variables have their gradient contribution down-weighted. Conversely, cell outliers with low values of highly, and have more impact on the model parameters. of the robustness property of the RVAE. The gradient of Examining the gradients of helps to understand given assumption that Bernoulli( is a parameter that reflects our belief about the cleanliness of the model under dataset . Finally, the ELBO for the RVAE model can be written as

\[ \mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} \sum_{d=1}^{D} \mathbb{E}_{q_{\phi}(z_n|x_n)} \left[ \pi_{nd}(x_n) \log p_\theta(x_{nd}|z_n) \right] + \frac{1}{N} \sum_{n=1}^{N} \sum_{d=1}^{D} \mathbb{E}_{q_{\phi}(z_n|x_n)} \left[ (1 - \pi_{nd}(x_n)) \log p_\theta(x_{nd}) \right] - \frac{1}{N} \sum_{n=1}^{N} D_{KL}(q_{\phi}(z_n|x_n)||p(z_n)) - \frac{1}{N} \sum_{n=1}^{N} D_{KL}(q_{\pi}(w_{nd}|x_n)||p(w_{nd})). \]  

(7)

Examining the gradients of helps to understand the robustness property of the RVAE. The gradient of with respect to the model parameters is given by

\[ \nabla_\theta \mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} \sum_{d=1}^{D} \pi_{nd}(x_n) \mathbb{E}_{q_{\phi}(z_n|x_n)} \left[ \nabla_\theta \log p_\theta(x_{nd}|z_n) \right]. \]

(8)

We see that \( \pi_{nd}(x_n) \) acts as a weight on the gradient. Cells that are predicted as clean will have higher values of \( \pi_{nd}(x_n) \), and so their gradients are weighted more highly, and have more impact on the model parameters. Conversely, cell outliers with low values of \( \pi_{nd}(x_n) \) will have their gradient contribution down-weighted. A similar formulation can be obtained for the encoder parameters \( \phi \).

\[ \text{3.1 Outlier Model} \]

The purpose of the outlier distribution \( p_0(x_{nd}) \) is to explain the outlier cells in the dataset, completely removing their effect in the optimization of the parameters of clean component \( p_\theta \). For categorical features, we propose using the uniform distribution \( p_0(x_{nd}) = \mathcal{C}_d^{-1} \). Such a uniform distribution assumption has been used in multiple object modelling (Williams and Titsias, 2003) as a way to factor in pixel occlusion. In (Chemudugunta et al., 2006) a similar approach for background words is proposed. For real features, we standardize the features to have mean 0 and standard deviation 1. We use an outlier model based on a broad Gaussian distribution\(^2\) \( p_0(x_{nd}) = \mathcal{N}(x_{nd}|0, \sigma) \), with \( \sigma > 1 \). Anomalous cells modelled by the outlier component will be further apart from \( m_d(z_n) \) relative to clean ones.

Although more complex distributions can be used for \( p_0(x_{nd}) \), we show empirically that these simple distributions are enough to detect outliers from a range of noise levels (Section 3). Furthermore, RVAE can easily be extended to handle other types of features (Nazabal et al., 2018), e.g.: for count features we can use a Poisson likelihood, where the outlier component \( p_0 \) would be a Poisson distribution with a large rate; for ordinal features we could have an ordinal logit likelihood, where \( p_0 \) can be a uniform categorical distribution.

\[ \text{3.2 Inference} \]

We use a hybrid procedure to train the parameters of RVAE that alternates amortized variational inference using stochastic gradient for \( \phi \) and \( \theta \), and coordinate ascent over \( \pi \). When we do not amortize \( \pi \), but rather treat each \( \pi_{nd}(x_n) \) as an independent parameter of the optimization problem, then an exact solution for \( \pi_{nd}(x_n) \) is possible when \( \phi \) and \( \theta \) are fixed. Optimizing the ELBO (7) w.r.t. \( \pi_{nd}(x_n) \), we obtain an exact expression for the optimum \( \hat{\pi}_{nd}(x_n) \).

\[ \begin{align*}
    \hat{\pi}_{nd}(x_n) &= g \left( r + \log \frac{\alpha}{1 - \alpha} \right), \\
    r &= \mathbb{E}_{q_{\phi}(z_n|x_n)} \left[ \log \frac{p_\theta(x_{nd}|z_n)}{p_0(x_{nd})} \right]
\end{align*} \]

where \( g \) is the sigmoid function. The first term in (9) represents the density ratio \( r \) between the clean component \( p_\theta(x_{nd}|z_n) \) and the outlier component \( p_0(x_{nd}) \). When \( r > 1 \) it will bias the decision towards assuming the cell being clean, conversely \( r < 1 \) it will bias the decision towards the cell being dirty. Such a ratio \( r \) has

---

1Recall from (Bishop, 2006) (Section 9, page 431) mixture models can also be written in product form using mixing variables \( w_{nd} \), as we adopt here.

2This is standard, see (Quinn et al., 2009; Gales and Olsen, 1999).

3The derivation of equation (6) is provided in the Supplementary Material (Section 2).
arisen in the literature [Hido et al., 2011; Yamada et al., 2017]. The second term in (9) represents our prior belief about cell cleanliness, denoted by α ∈ [0, 1]. Higher values of α will skew the decision boundary towards a higher ˆπ_{nd}(x_n), and vice-versa. This coordinate ascent strategy is common in variational inference for conjugate exponential family distributions (Jordan et al., 1999). We term this model RVAE-CVI (Coordinate ascent Variational Inference) below.

Alternatively π_{nd}(x_n) can be obtained using amortized variational inference. However two problems arise in the process. First an inference gap is introduced by amortization, leading to slower convergence to the optimal solution. Secondly, there might not be enough outliers in the data to properly train a neural network to recognize the decision boundary between clean and dirty cells. We term this model RVAE-AVI (Amortized Variational Inference). RVAE inference is summarized in Algorithm 1 for both the coordinate ascent version (RVAE-CVI) and the amortized version (RVAE-AVI).

Algorithm 1 RVAE Inference

1: procedure RVAE(η learning rate, M batch size, T number epochs, α prior value)
2: if RVAE-AVI = True then
3: Define NN parameters: Ψ = {φ, θ, τ}; τ is NN params of π_r(x_n) encoder (AVI)
4: else if RVAE-CVI = True then
5: Define NN parameters: Ψ = {φ, θ};
6: initialize Ψ;
7: for 1,...,T do
8: Sample mini-batches \( \{X_m\}_{m=1}^{M} \sim p(X) \);
9: Evaluate \( p_\eta(x_{nd}|x_m) \) and \( p_\theta(x_{md}) \forall m, d \); Forward-Pass of Deep AE
10: if RVAE-CVI = True then
11: Infer ˆπ_{nd}, ∀m, d using eq. (9);
12: else if RVAE-AVI = True then
13: Evaluate encoder π_r(x_n);
14: Forward-Pass of NN
15: \( g_\psi \leftarrow \nabla_\psi L(\Psi, π_r(x_n), α) \) using eq. (7);
16: \( \Psi \leftarrow \Psi + η \cdot \text{Adam}(\Psi, g_\psi) \);

3.3 Outlier Scores

A natural approach to determine which cells are outliers in the data is computing the likelihood of the cells under the trained model. In a VAE, the scores for row and cell outliers would be

Cell: \(-E_{q_\phi(z_n|x_n)}[\log p_\theta(x_{nd}|z_n)]\)

Row: \(-\sum_{d=1}^{D} E_{q_\phi(z_n|x_n)}[\log p_\theta(x_{nd}|z_n)]\),

where a higher score means a higher outlier probability. However, likelihood-based outlier scores present several problems, specifically for row scores. In mixed type datasets categorical features and real features are modelled by probability and densities distributions respectively, which have different ranges. Often this leads to continuous features dominating over categorical ones. With the RVAE we propose an alternative outlier score based on the mixture probabilities ˆπ_{nd}(x_n)

Cell: \(-\log ˆπ_{nd}(x_n)\)
Row: \(-\sum_{d=1}^{D} \log ˆπ_{nd}(x_n)\),

where again a higher score means a higher outlier probability. Notice that the row score is just the negative log-probability of the row being clean, given by ˆπ_n = ∏_{d=1}^{D} ˆπ_{nd}(x_n). These mixture-based scores are more robust against some features or likelihood models dominating the row outlier score, making them more suitable for mixed type datasets.

3.4 Repairing Dirty Cells

Cell repair is related to missing data imputation, however this is a much harder task, since positions of anomalous cells are not given, and need to be inferred. After the anomalous cells are identified, a robust generative model allows to impute them given the dirty row directly. In general, repair under VAE-like models can be obtained via maximum a posteriori (MAP)

\( \hat{x}^{i}_{nd} = \arg \max_{x_{nd}} p_\theta(x_{nd}|z_n) \ x_n \sim q_\phi(z_n|x_n^o) \)

where superscript i denotes imputed or clean cells (depending on context), and o corresponds to observed or dirty cells. In the case of RVAE, \( p_\theta(x_{nd}|z_n) \) is the clean component responsible for modelling the underlying clean data, see [1]. This reconstruction is akin to robust PCA’s clean component. In practice, for real features \( \hat{x}^{i}_{nd} = m_d(z_n) \), the mean of the Gaussian likelihood, and for categorical features \( \hat{x}^{i}_{nd} = \arg \max_c f(a_{dc}(z_n)) \), the highest probability category. Other repair strategies are discussed in the Supplementary Material (Section 10).

4 Experiments

We showcase the performance of RVAE [1] and baseline methods, for both the task of identifying row and cell outliers and repairing the corrupted cells in the data. Four different datasets from the UCI repository [Lichman, 2013], with a mix of real and categorical features, were selected for the evaluation (see Supplementary Material, Section 1). We compare RVAE with

https://github.com/sfme/RVAE_MixedTypes/
ABDA (Vergari et al., 2019) in a different OD task in the Supplementary material (Section 9).

4.1 Corruption Process

All datasets were artificially corrupted in both training and validation sets. This is a standard practice in OD (Futami et al., 2018; Redyuk et al., 2019; Krishnan et al., 2016; Natarajan et al., 2013), and a necessity in our setting, due to the scarcity of available datasets with labelled cell outliers. No previous knowledge about corrupted cell position, or dataset corruption proportion is assumed. For each dataset, a subset of cells are randomly selected for corruption, following a two step procedure: a) a percentage of rows in the data are selected at random to be corrupted; b) for each of those selected rows, 20% of features are corrupted at random, with different sets of features being corrupted in each select row. For instance, a 5%-20% scenario means that 5% of the rows in the data are randomly selected to contain outliers, and for each of these rows, 20% of the features are randomly corrupted, leading to 1% of cells corrupted overall in the dataset. We will consider for the experiments five different levels of row corruption, \{1%, 5%, 10%, 20%, 50%\}, leading to five different levels of cells corrupted across the data, \{0.2%, 1%, 2%, 4%, 10%\}. This corruption process is repeated 5 times (instances). Here, we show results for the aggregate of all datasets for one of those instances, leaving the full disclosure of the results to the Supplementary Material (Section 8).

Real features: Additive noise is used as a noising process, with dirty cell values obtained as \(x'_{nd} \sim x_{nd} + \zeta\), with \(\zeta \sim p_{\text{noise}}(\mu, \eta)\). Four different noise distributions \(p_{\text{noise}}\) are explored: Gaussian noise \((\mu = 0, \eta = 5\sigma_d)\), with \(\sigma_d\) the standard deviation of feature \(d\); Laplace noise \((\mu = 0, \eta = \{4\sigma_d, 8\sigma_d\}\); Log-Normal noise \((\mu = 0, \eta = 0.75\sigma_d)\); and a Mixture of two Gaussian noise components \((\mu_1 = -0.5, \eta_1 = 3\sigma_d, \text{with probability 0.6 and } \mu_2 = 0.5, \eta_2 = 3\sigma_d \text{ with probability 0.4})\).

Categorical features: The noising process is based on the underlying marginal (discrete) distribution. We replace the cell value by a dirty one by sampling from a tempered categorical distribution\(^5\) (and excluding the current clean category):

\[
x'_{ndc} \sim \frac{p_c(x'_{nd})^\beta}{\sum_{i=1}^{C} p_c(x'_{ndi})^\beta},
\]

with the range \(\beta = [0, 0.5, 0.8]\). Notice that, when \(\beta = 0\), the noise process reduces to the uniform distribution, while when \(\beta = 1\), the noising process follows the marginal distribution.

4.2 Evaluation metrics

In the OD experiments, we use Average Precision (AVPR) (Everingham et al., 2014; Salton and McGill 1986), computed according to the outlier scores of each method. AVPR is a measure of area under the precision-recall curve, so higher is better. For cell outliers we report the macro average of the AVPR for each feature in the dataset\(^6\). In the repair experiments, different metrics are necessary depending on the feature types. For real features, we compute the Standardized Mean Square Error (SMSE) between the estimated values \(\hat{x}'_{nd}\) and the original ground truth in the dirty cells \(x'_{nd}\), normalized by the empirical variance of the ground truth values: \(\text{SMSE}_d = \frac{\sum_{i=1}^{N_c} (x'_{ndi} - \hat{x}'_{ndi})^2}{\sum_{i=1}^{N_c} (x'_{ndi} - \mu_d)^2}\), where \(\mu_d\) is the statistical mean of feature \(d\) and \(N_c\) is the number of corrupted cells for that feature\(^7\). For categorical features, we compute the Brier Score for each category in the feature: \(\text{Brier}_d = \frac{1}{2N_c} \sum_{n=1}^{N_c} \sum_{c=1}^{C} (x'_{ndc} - p_c(x'_{nd}))^2\), where \(p_c(x'_{nd})\) is the probability of category \(c\) for feature \(d\), \(x'_{ndc}\) the one-hot true value for category \(c\), and \(C\) the number of unique categories in the feature. We used the coefficient \(\frac{1}{2}\) in the Brier score to limit the range to \([0, 1]\). We name both metrics in the rest of the paper as SMSE for simplicity, but the correct metric is always used for each type.

4.3 Competing Methods

We compare to several standard OD algorithms. Most methods are only concerned about row OD, whilst only a few can be used for cell OD. For more details on parameter selection and network settings for RVAE and competitor methods, see the Supplementary Material (Section 3).

**Exclusively row outlier detection.** We consider Isolation Forest (IF) (Liu et al., 2008), an OD algorithm based on decision trees, which performed quite well in the extensive comparison of (Emmott et al., 2015); and One Class Support Vector Machines (OC-SVM) (Chen et al., 2001) using a radial basis function kernel.

**Row and cell outlier detection.** We compare to (i) estimating the Marginal Distribution for each feature and using the negative log-likelihood as the outlier score. For real features we fit a Gaussian mixture model with the number of components chosen via Bayesian Information Criterion. The maximum number of components is limited to $5$. The AVPR macro average is defined as the average of the AVPR for all the features in a dataset.

\(^5\)Also known as power heuristic in importance sampling.

\(^6\)The AVPR macro average is defined as the average of the AVPR for all the features in a dataset.

\(^7\)In our experiments \(\mu_d = 0\) in practice, since the data has been standardized before using any method.
ments is set at 40. For categorical features, the discrete distribution is given by the normalized category frequency; (ii) a combination of OC-SVM and Marginal Distribution for each feature. We use Platt scaling to transform the outlier score of OC-SVM for each row (to obtain log-probability), and then combine it with marginal log-likelihood of each feature. This score, a combined log-likelihood, is then used for cell OD; (iii) VAEs with $\ell_2$ regularization and outlier scores given by (10); (iv) DeepRPCA (Zhou and Paffenroth, 2017), an unsupervised model inspired by robust PCA. The data $X$ is divided in two parts $X = R + S$, where $R$ is a deep autoencoder reconstruction of the clean data, and $S$ is a sparse matrix containing the estimated outlier values (see Supplementary Material, Section 3, for further description). Outlier scores for rows are given by the Euclidean norm $\sqrt{\sum_{d=1}^{D}(s_{nd})^2}$, whilst cell scores are given by $|s_{nd}|^2$, where $s_{nd} \in S$. (v) A set of Conditional Predictors (CondPred), where a neural network parametrizing $p_\theta(x_n)$ is employed for each feature in the data given the rest. However, $\ell_2$ regularization is necessary to prevent overfitting, and the model is overall much slower to train than VAE.

**Repair:** We compare to VAE, DeepRPCA, Marginal Distribution method and Conditional Predictor (CondPred) method for repairing dirty cells (same model parameters as in OD). We use (12) for all VAE-based methods. For DeepRPCA we use $X^i = R$. For CondPred the estimate is $\hat{x}_{nd} = \text{arg max}_{x_{nd}} p_\theta(x_{nd}|x_n^d \backslash_d)$, with $x_n^d \backslash_d$ meaning all features in $x_n$ except $x_{nd}$. The Marginal Distribution method takes $x_{nd}$ and uses as estimate the mean of the closest GMM component in the real line. For RVAE, results using a different inference strategy are provided in Supplementary Material, using pseudo-Gibbs sampling (Rezende et al., 2014).

### 4.4 Hyperparameter selection for competing methods

In order to tune the hyperparameters for the competing methods, we reserved a validation set with known inlier/outlier labels and ground truth values. This validation set was not used by the RVAE method. Thus the performance obtained by the competitor methods is an optimistic estimate of their performance in practice. Note also that RVAE-CVI is robust to the selection of its parameter $\alpha$ in (9), as we will show in Section 4.8.

In Figure 1 we compare the performance of the conditional predictor method and VAE, with respect to RVAE-CVI when $\ell_2$ regularization is not used, and when the best $\ell_2$ regularization value is used for each dataset. We can observe clearly that a significant gap exists in the performance of these competitor methods when not fine-tuned, making it explicit the dependence of these methods from a labelled validation set. In the rest of the experiments we will use the best possible version of each competitor method.

### 4.5 Outlier detection

We compare the performance of the difference methods in OD, both at row and cell levels. We focus on Gaussian noise ($\mu = 0, \eta = 5\sigma_d$) for real features and uniform categorical noise, i.e. $\beta = 0$ in (13), relegating results on other noise processes scenarios to Section 4.7. In Figure 2 we show the average OD performance across all datasets for all OD models in terms of both row (left figure) and cell OD (right figure). We term RVAE-CVI-nll our model with outlier score as defined in (10) and RVAE-CVI-pi our model with outlier score as defined in (11). We relegate RVAE-AVI results to the Supplementary Material (Section 6), since RVAE-AVI is worse than RVAE-CVI in general. Additional results on the OD for each dataset are also available in the Supplementary Material (section 4). In the right figure, we can observe that RVAE-CVI is performing similar to the conditional predictor method on cell OD while being consistently better than the other methods. Additionally, it performs comparatively well in row OD, being similar to the conditional predictor at higher noise levels. We would like to remind that RVAE-CVI does not need a validation set to select its parameters. This means that RVAE-CVI is directly applicable for datasets where no ground truth is available, providing a comparable performance to other methods where pa-
Parameter tuning for each dataset is necessary. Figure 2 (left figure) also confirms our hypothesis (Section 3.3) on the proper score to compute row outliers. We can see in the upper figure that RVAE-CVI using scores based on estimate \( \hat{\pi}_{nd}(x_n) \), as per score (11), are better for row OD compared to averaging different feature log-likelihoods (10). A further analysis of the OD performance of each model for the different feature types is shown in Figure 3. While the model based on estimating the marginal distribution works well for real features (left figure), it performs poorly on categorical features. Similarly, for the method combining OCSVM and the marginal estimator, it detects outliers better than the other methods in real features and low noise levels, but performs poorly for categorical features. In contrast, RVAE performs comparatively better across different types than the other models, with comparable performance to the conditional predictor.

### 4.6 Repair

In this section, we compare the ability of the different models to repair the corrupted values in the data. We use the same noise injection process as Section 4.5. Figure 4 shows the average SMSE repair performance across datasets for all models when repairing the dirty cells in the data (more details in the Supplementary Material, Section 5). We can observe that RVAE-CVI outperforms the other models for all the different cell corruption scenarios, being of particularly significance in lower cell corruption regimes. This is significantly important since all the comparator methods required hyperparameter selection and still performed worse than RVAE-CVI. Also, in Figure 5, we can see the repair performance of different models according to the types of features in the data. Notice that RVAE-CVI is consistently better than the other models across real features while being slightly worse on categorical features.

### 4.7 Robustness to Noising Processes

Figure 6 shows the performance of the different models across there different combinations of noise processes for all datasets and noise corruption levels (three other noise processes are covered in the Supplementary Material, Section 7). We notice that all the models perform consistently across different types of noise. RVAE-CVI is performing better in repair for low level noise corruption, while providing competitive performance in OD. Also, our choice of outlier models on Section 3.1 does not have a negative effect on the ability of RVAE to detect outliers and repair them. Different noise processes define what is feasible to detect and repair.

### 4.8 Robustness to hyperparameter values

In this section, we examine the robustness of RVAE inference to the choice \( \alpha \), and study its effect in both
OD and repair of dirty cells. We have analyzed values of $\alpha$ in the set $\{0.2, 0.5, 0.8, 0.9, 0.99\}$ and evaluated RVAE-CVI in all datasets under all levels of cell corruption and the noising process of Sections 4.5 and 4.6. Figure 7 shows the performance of RVAE-CVI in both OD (left figure) and repair (right figure) across different values of $\alpha$. Larger values of $\alpha$ lead in general to a better OD performance, with a slight degradation when we approach $\alpha = 1$. Repair performance is consistent across different choices of $\alpha$, but values closer to 0 or 1 lead to a degradation when repairing dirty cells.

5 Related Work

There is relevant prior work in the field of OD and robust inference in the presence of outliers, a good meta-analysis study presented in (Emmott et al., 2015). Different deep models have been applied to this task, including autoencoders (Zong et al., 2018; Nguyen and Vien, 2018; Zhou and Paffenroth, 2017), VAEs (An and Cho, 2015; Wang et al., 2017b), and generative adversarial networks (GANs) (Schlegl et al., 2017; Lee et al., 2018). In (Nalisnick et al., 2018) the authors show that deep models trained on a dataset assign high likelihood to instances of a quite different dataset, which is problematic in OD. We identify outliers during training rather than from a fully-trained model, down-weighting their effect on parameter learning. Earlier in training, the model had less chance to overfit, so it should be easier to detect outliers.

Most closely related to our model are methods based on robust PCA (RPCA) and autoencoders. They focus on unsupervised learning in the presence of outliers, even though most methods need labelled data for hyper-parameter tuning (Candes et al., 2011; Zhou and Paffenroth, 2017; Zong et al., 2018; Nguyen and Vien, 2018; Xu et al., 2018; Akrami et al., 2019). RPCA-based alternatives, often assume that the features are real valued, and model the noise as additive with a Laplacian prior. A problem in RPCA-type models is that often the hyper-parameter that controls the outlier mechanism is dataset dependent and difficult to interpret and tune. In (Wang et al., 2017b) the authors proposed using a VAE as a recurrent unit, iteratively denoising the images. This iterative approach is reminiscent of the solvers used for RPCA. However their work is not easily extended to mixed likelihood models and suffers from the same problems as VAEs when computing row scores (Section 3.3).

Robust Variational Inference. Several methods explore robust divergences for variational learning, under the presence of outliers (Regli and Silva, 2018; Futami et al., 2018), applied to supervised tasks. These divergences have hyper-parameters which are dataset dependent, and can be difficult to tune in unsupervised OD; in contrast, the $\alpha$ hyperparameter used in RVAE is arguably more interpretable, and experimentally robust to misspecification. Recently a VAE model using one of these divergences at the decoder was proposed for down-weighting outliers (Akrami et al., 2019). However, in contrast to our model, they focused on image datasets and are not concerned with cell outliers. The same hyperparameter tuning problem arises, and it is not clear out to properly extend to categorical features.

Bayesian Data Reweighting. Wang et al (Wang et al., 2017a) proposes an approach that raises the likelihood of each observation by some weights and then infer both the latent variables and the weights from corrupted data. Unlike RVAE, these weights are only defined for each instance, so the method cannot detect cell-level outliers. Additionally, the parameters of the model are trained via MCMC instead of variational inference, making them more difficult to apply in the context of deep generative models.

Classifier Confidence. Several methods explore adding regularization to improve neural network classifier robustness to outliers (Lee et al., 2018; Hendrycks et al., 2019). However, the regularization hyper-parameters are not interpretable and often require a validation dataset to tune them. Other works like (Hendrycks and Gimpel, 2017), use the confidence of the predicted distribution as a measure of OD.

6 Conclusions

We have presented RVAE, a deep unsupervised model for cell OD and repair in mixed type tabular data. RVAE allows robust identification of outliers during training, reducing their contribution to parameter learning. Furthermore, a novel row outlier score for mixed type features was provided. RVAE performs similarly in OD and repair to other methods that heavily rely on fine tuning hyper-parameters with a trusted labelled set. Still, RVAE outperforms or matches the other models in dirty cell repair.
Acknowledgements

This work was supported by The Alan Turing Institute under the EPSRC grant EP/N510129/1. This work was supported in part by the EPSRC Centre for Doctoral Training in Data Science, funded by the UK Engineering and Physical Sciences Research Council (grant EP/L016427/1) and the University of Edinburgh. We also thank reviewers for fruitful comments and corrections. The author Simão would like to thank helpful discussions with: Afonso Eduardo, Kai Xu and CUP group members in general.

References


Robust Variational Autoencoders for Outlier Detection and Repair of Mixed-Type Data


Haowen Xu, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, Dan Pei, Yang Peng, Jie Chen, Zhaogang Wang, and Honglin Qiao. Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications. In *WWW*, 2018.

