Data and text mining

Anomaly detection in mixed high dimensional molecular data

Lena Buck [1,*], Tobias Schmidt[1], Maren Feist [2], Philipp Schwarzfischer [3], Dieter Kube [2], Peter J. Oefner [3], Helena U. Zacharias [4], Michael Altenbuchinger [5], Katja Dettmer [3], Wolfram Gronwald [3,*], and Rainer Spang [1,*]

[1] Department of Statistical Bioinformatics, University of Regensburg, 93040, Regensburg, Germany.
[2] Department of Hematology and Medical Oncology, University Medicine Goettingen, 37075 Goettingen, Germany.
[3] Institute of Functional Genomics, University of Regensburg, 93040, Regensburg, Germany.
[4] Peter L. Reichertz Institute for Medical Informatics of TU Braunschweig and Hannover Medical School, Hannover Medical School, 30625 Hannover, Germany.
[5] Department of Medical Bioinformatics, University Medical Center Göttingen, 37075 Göttingen, Germany.

*Shared last authorship and to whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Mixed molecular data combines continuous and categorical features of the same samples, such as OMICS profiles with genotypes, diagnoses, or patient sex. Like all high dimensional molecular data it is prone to incorrect values that can stem from various sources as for example the technical limitations of the measurement devices, errors in the sample preparation or contamination. Most anomaly detection algorithms identify complete samples as outliers or anomalies. However, in most cases, not all measurements of those samples are erroneous but only a few one-dimensional features within the samples are incorrect. These one-dimensional data errors are continuous measurements that are either located outside or inside the normal ranges of their features but in both cases show atypical values given all other continuous and categorical features in the sample. Additionally, categorical anomalies can occur for example when the genotype or diagnosis was submitted wrongly.

Results: We introduce **ADMIRE** (**A**nomaly **D**etection using **MI**xed g**R**aphical mod**E**ls), a novel approach for the detection and correction of anomalies in mixed high dimensional data. Hereby, we focus on the detection of single (one-dimensional) data errors in the categorical and continuous features of a sample. For that the joint distribution of continuous and categorical features is learned by Mixed Graphical Models, anomalies are detected by the difference between measured and model-based estimations and are corrected using imputation. We evaluated ADMIRE in simulation and by screening for anomalies in one of our own metabolic data sets. In simulation experiments ADMIRE outperformed the state-of-the-art methods Local Outlier Factor, stray and Isolation Forest.

Availability: All data and code is available at https://github.com/spang-lab/adadmire. ADMIRE is implemented in a python package called adadmire which can be found at https://pypi.org/project/adadmire.

Contact: wolfram.gronwald@ukr.de

Supplemental information: Supplemental data are available at *Bioinformatics* online.

1 Introduction

Molecular data is error prone. Systematic errors in e.g. sample collection or preparation can affect large sets of features and need to be corrected using normalization methods. Additionally, technical problems can affect individual measurements. Due to the different molecular properties of the measured features, it is often the case that a sample shows only in a few of its measured features abnormalities while the rest of them are inconspicuous. Also, not all samples might be affected in the same way as each sample is usually processed separately and therefore is exposed to a different kind of error source. Consequently, molecular data sets contain individual data errors that can affect each measured feature in each sample in a different way. These one-dimensional data errors are especially hard to detect in the setting of high-dimensional molecular data sets. Furthermore, they might present themselves as univariate outliers, with measured values exceeding the range of the features by multiple orders. But they also appear as anomalies when a value fits well into the univariate distribution of its feature, but not into the joint distribution of all features. For example, if a gene shows expression values between 4-6 in men and between 8-14 in women, a value of 12 in a man is suspicious.

More formally, a given value x_{ij} of a feature j in a sample i might be a typical value for the marginal distribution of feature j, but not for its conditional distribution given all other features of sample i. These anomalies can only be detected when the information given by the categorical, phenotypic information is taken into consideration as well. But this data can also contain anomalies. Data entry errors or a mix-up during the experimental procedure can lead to artefacts in the phenotypical information of a data set. Samples are then assigned for example to the wrong treatment class, a female participant is considered as a male, etc.

The literature knows numerous methods for detecting uni-variate outliers in molecular data (Grubbs, 1969) and for detecting multi-variate anomalies in continuous (Korn, F. et al, 2001; DeCoste and Levine, 2004; Hodge and Austin, 2004; Ando, 2007) as well as in discrete data (John, 1995). A common approach to anomaly detection is using the k nearest neighbors to detect anomalies within this neighborhood as done by the Local outlier factor (LOF) (Breunig *et al.*, 2000) and the Search and TRace AnomalY (stray) algorithm (Talagala *et al.*, 2021), or to use random forests to isolate anomalous samples (see Isolation Forest (Liu *et al.*, 2008)). Unlike our method which aims at the detection of anomalies in individual entries of the data matrix, those algorithms however confine themselves to identifying suspicious samples, see supplement.

Most data sets in molecular biology are mixed. Continuous OMICS data is complemented by discrete phenodata like patient characteristics (sex, diagnosis, treatment), experimental conditions (experimental groups, controls) or technical designs (batches, repetitions). Therefore, we developed a novel approach to anomaly detection based on Mixed Graphical Models (MGMs). MGMs (Lee and Hastie, 2015, Cheng *et al.*, 2017) are well established generalisations of Gaussian Graphical Models (GGMs) (Lauritzen, 1996, Meinshausen and Bühlmann, 2006) to mixed data. Beyond anomaly detection MGMs have been succesfully used for studying the structure of metabolic, proteomic or transcriptomic networks (Chun *et al.*, 2013, Wang, T. et al, 2016, Zhao and Duan, 2019, Altenbuchinger, M. et al, 2020). We briefly review the concept of MGMs, describe how ADMIRE detects anomalies, handles missing values, validate it in simulation experiments, compare it to alternative approaches and demonstrate its power in the contexts of finding experimental artefacts in a state of the art metabolomics data set.

2 Methods

In a nutshell: ADMIRE fits for each sample in a leave one out approach a MGM to the mixed data set. From this MGM we derive the conditional distribution of a feature given all other features. We then compare an actual observation of a specific feature in a specific sample with its corresponding conditional distribution. If the value is far away from what can be expected from the model given all other features of the same sample, we flag it as anomaly and the user may choose to replace it by a model based imputation.

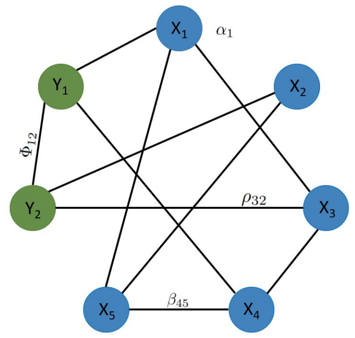


Fig. 1. A Mixed Graphical Model. The nodes include both continuous features (blue), and discrete features (green). A missing edge between two nodes denotes their conditional independence given all other variables. The node and edge weights correspond to the couplings and potentials in equation (1).

2.1 Mixed Graphical Models

Like Gaussian Graphical Models, their continuous counterpart, MGMs learn the conditional independence structure of a given set of features together with parameters that define the joint distribution of both continuous and discrete variables (Lee and Hastie, 2015). The conditional independence structure is encoded in an undirected graph where nodes represent features and edges the conditional dependencies between them. The conditional distribution of a node (feature) x_j given all other nodes (features) $x_{\setminus j}$ only depends on the values of the nodes that are directly connected to x_j. More formally, the data is modelled as a pairwise Markov random field with density

$$\begin{aligned} p(x,y;\Theta) \propto exp\Bigg(\sum_{j=1}^{p} \sum_{s=1}^{p} -\tfrac{1}{2}\beta_{js}x_j x_s + \sum_{j=1}^{p} \alpha_j x_j \\ + \sum_{j=1}^{p} \sum_{s=1}^{q} \rho_{js}(y_s)x_j + \sum_{j=1}^{q} \sum_{s=1}^{q} \phi_{js}(y_j,y_s) \Bigg), \end{aligned} \tag{1}$$

where $x_1, \ldots, x_p$ are continuous features. $y_1, \ldots, y_q$ discrete features where y_j has L_j distinct states. Together the x_j and the y_j form the nodes of the networks. The remaining parameters are node and edge weights (couplings) that jointly define how the distribution of a node depends on the values of its direct neighbors. β_{js} are couplings between two continuous nodes, α_j are continuous node potentials, $\rho_{sj}(y_j)$ are continuous-discrete couplings and $\phi_{sj}(y_s, y_j)$ are discrete-discrete couplings. We denote the complete parameter set by $\Theta = \{\{\beta_{js}\}, \{\alpha_j\}, \{\rho_{jt}\}, \{\phi_{rt}\}, j, s \in \{1 \ldots p\}, r, t \in \{1 \ldots q\}\}$. Figure 1 visualizes the roles of individual parameters.

To simplify notations we will omit the index i of the sample whenever the focus is on the features x_j in the continuous and y_j in the discrete case. Single data points in our data matrix are realizations of the random variables x_j or y_j and are denoted by x_{ij} or y_{ij} respectively.

Equation (1) defines the full joint distribution of both discrete and continuous features. To judge whether a specific continuous x_{ij} or discrete y_{ij} data point fits to all other observed data points in the same sample, we need to calculate the conditional distribution of a node given all its direct neighbors. Following (Lee and Hastie, 2015) the conditional distribution of a continuous variable x_j given all other continuous variables $x_{\setminus j}$ and discrete variables y is Gaussian with

$$x_j | (x_{\setminus j}, y; \Theta) \sim \mathcal{N}(\hat{x}_j, \beta_{jj}^{-1}) \tag{2}$$

where the linear regression

$$\hat{x}_j = \alpha_j + \sum_s \rho_{js}(y_s) - \sum_{s \neq j} \beta_{js} x_s \tag{3}$$

yields the mean and the variance is given by β_{jj}^{-1}.

The conditional distribution of a discrete variable y_j with L_j states has the probability mass function

$$p(y_j|y_{\setminus j},x;\Theta) = \frac{exp\big(\sum_s \rho_{sj}(y_j)x_s+\Phi_{jj}(y_j,y_j)+\sum_{s\neq j} \Phi_{js}(y_j,y_s)\big)}{\sum_{l=1}^{L_j} exp\big(\sum_s \rho_{sj}(l)x_s+\Phi_{jj}(l,l)+\sum_{s\neq j} \Phi_{js}(l,y_s)\big)} \tag{4}$$

which corresponds to a multiclass logistic regression. Together, the conditional distributions (2) and (4) describe the conditional independence structure via the regression coefficient of a variable on all others. We denote the conditional distribution (2) of a continuous feature x_j in a sample i by Q_{ij} and the conditional distribution of a discrete feature y_j in sample i by p_{ij}.

2.2 Detection of data anomalies in continuous features

ADMIRE builds on the discrepancies between the original observations x_{ij} from their model based conditional distributions and the resulting linear predictions $\hat{x}_{ij}$. The estimated means $\hat{x}_{ij}$ from the conditional distribution (2) serve as a regression based re-estimation of a continuous feature based on all other features see (Altenbuchinger, M. et al, 2019). Furthermore, the conditional distribution describes how well an observed data point fits to the rest of the data. More specifically, it tells us the probability of observing a specific feature value given all other continuous and categorical features for the same sample. Let x_{ij} be the observed, measured value, $\hat{x}_{ij}$ the estimated mean and $\epsilon = |x_{ij} - \hat{x}_{ij}|$ the deviation of the observed value from the estimated mean. Then the probability p of observing a deviation greater or equal ϵ is given by

$$p = \mathbb{P}(x \leq \hat{x} - \epsilon) + \mathbb{P}(x \geq \hat{x} + \epsilon) = 2 * F(\hat{x} - \epsilon), \tag{5}$$

where F is the cummulative distribution function of $x \sim \mathcal{N}(\hat{x}_{ij}, \beta_{jj}^{-1})$. We apply (5) to all entries x_{ij} in the data matrix and rank them according to their probability. Entries at the top of this list have a low probability and are most likely anomalies. Mind that the same ranking is achieved, when instead of the probabilities the scores $s_{ij}^o = \frac{|x_{ij} - \hat{x}_{ij}|}{\sqrt{\beta_{jj}^{-1}}}$ are used for ranking. Data entries with a high deviation from the estimated mean rank at the top of the list.

We threshold this list by comparing the observed scores with anomaly-free scores simulated from the estimated distribution (2). For every observed data point x_{ij}, let Q_{ij} be its model based conditional distribution given all other features $k \neq j$ of sample i defined in (2). We generate random data by drawing one random value r_{ij} from each Q_{ij}, resulting in as many random data points as original continuous observations. Note that this data does not contain anomalies, since every simulated data point was drawn from its proper conditional distribution. Let $s_{ij}^r = \frac{|r_{ij} - \hat{x}_{ij}|}{\sqrt{\beta_{jj}^{-1}}}$ be the score of r_{ij}. The joint distribution of the s_{ij}^r represents a score distribution for data in which no anomalies exist. Next, we sort the lists of observed scores s_{ij}^o and random scores s_{ij}^r and compare them rank by rank. If the real data contains anomalies, the scores of top ranking data points are higher than rank matching random scores. This results in different score distributions for highly ranking scores. To stabilize the distribution of random scores, we draw repeatedly from the distributions Q_{ij} and compute s_{ij}^r by averaging the resulting scores rank by rank. The first random score that exceeds its matched observed score is chosen for thresholding the lists and we flag all data points with an observed score higher than this threshold value as anomalies.

2.3 Detection of discrete anomalies

Similar to the continuous case, we can calculate for each discrete data entry y_{ij} a score depending on the conditional distribution (4) and compare the resulting ranked list to anomaly-free scores generated from the estimated distribution.

Let $y_{ij} = k$ be the j-th discrete feature in sample i with observed state k. Then the discrete observed score is defined as $s_{ij}^o = -\log\big(p_{ij}(k)\big)$ where $p_{ij}(k)$ is the conditional probability (4) of observing state k in feature y_j for sample i given all other features (discrete and continuous). If the probability of observing $y_{ij} = k$ is low, the score s_{ij}^o is high and the discrete feature is most likely erroneous. For thresholding we draw for each observed discrete value y_{ij} a random value r_{ij} from the conditional distribution p_{ij}. If the observation $y_{ij} = k$ is an anomaly, the probability $p_{ij}(k)$ of observing state k should be low, resulting in a realization $r_{ij} \neq k$ with a different state. We define random scores by $s_{ij}^r = -\log(p_{ij}(r_{ij}))$. The random scores contain no anomalies. Again, we draw multiple times from the distribution and average over the repeated scores rank by rank. In line with the continuous case, we match observed and random scores rank by rank and set the threshold as the first random score that is higher as its observed counterpart.

2.4 Imputation of missing values

ADMIRE imputes missing values by a two step procedure. If the value of feature j is missing in sample i, ADMIRE pre-imputes it in step 1 by the value of j in the sample i', which has the smallest euclidean distance to i among all samples where the value of j is not missing. After the pre-imputation, feature j is re-scaled in the entire data set. In step 2, an MGM is fitted on the pre-imputed data set including calibration of the regularization parameter. Finally, all pre-imputed missing values are re-estimated, as described in section 2.2 and 2.3.

2.5 Implementation and model training

ADMIRE estimates the parameter set $\Theta = \{\{\beta_{js}\}, \{\alpha_j\}, \{\rho_{jt}\}, \{\phi_{rt}\}, j, s \in \{1 \dots p\}, r, t \in \{1 \dots q\}\}$ which defines the node and edge weights and hence specifies the joint probability distribution (1) together with the conditional distributions (2) and (4). Let $\{x_j\}_{j=1,\dots p}$ be the standardized continuous features with mean 0 and variance 1 across samples and $\{y_j\}_{j=1,\dots q}$ the discrete features. Then, following (Altenbuchinger, M. et al, 2019, Lee and Hastie, 2015) we minimize the negative pseudo log-likelihood

$$\tilde{l}(\Theta|x,y) = -\sum_{j=1}^{p} \log(Q(x_j|x_{\setminus j}, y; \Theta)) - \sum_{j=1}^{q} \log(p(y_j|x, y_{\setminus j}; \Theta)) + \lambda \left\|\Theta\right\|_1 \tag{6}$$

to estimate Θ. The pseudo-likelihood (6) consists of the product of all conditional distributions where $Q(x_s|x_{\setminus s}, y; \Theta)$ is the conditional distribution of a continuous variable given all other variables (2) and $p(y_r|x, y_{\setminus r}; \Theta)$ is the distribution of a discrete variable conditioned on all other variables (4). The term $\lambda \left\|\Theta\right\|_1$ corresponds to the lasso penalty with an additional weighting scheme to adjust for group sizes and variances of the features, see (Altenbuchinger, M. et al, 2019). Following (Altenbuchinger, M. et al, 2019) the minimization is done using a proximal gradient descent algorithm (O'Donoghue and Candès, 2013).

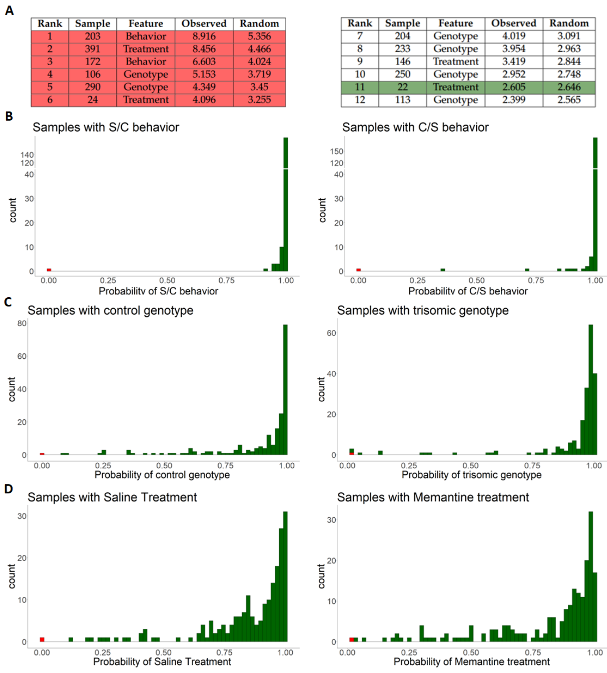
The sparseness parameter λ is calibrated by leave-one-out cross-validation. More precisely, let $\lambda = (\lambda_1, \dots, \lambda_m)$ be a sequence of values and $i \in \{1, \dots n\}$. For every λ_k and every i we fit a MGM leaving out the i-th sample. The resulting parameters $\Theta_i(\lambda_k)$ are used to re-estimate the continuous features x_{ij} via equation (3). For every λ_k we get a matrix

$\hat{x}_{ij}$ with the same dimension as the continuous input data. We choose the λ_k with smallest mean squared error between original and re-estimated data as the optimal sparseness parameter. The corresponding parameters $\Theta_i(\lambda_k)$ and the cross-validated estimators $\hat{x}_{ij}$ are finally used for anomaly detection.

Note that $\hat{x}_{ij}$ and $\hat{y}_{ij}$ are estimated given all other features in the sample and thus can be affected by other anomalies in the same sample. To compensate this effect, we check for each estimated data point $\hat{x}_{ij}$ in the continuous case or $\hat{y}_{ij}$ in the discrete case, if its regressors x_{ik} and y_{ik}, $k \neq j$, are potential anomalies (probability (5) of less than 5%). If a continuous estimator x_{ik} is flagged as a potential anomaly, we replace it by the group mean $\bar{x}_{lk}$ where l corresponds to the samples with the same discrete states as sample i. If a discrete estimator y_{ik} is flagged as an anomaly we replace its state by the state with highest estimated probability. The resulting adjusted estimators then are used in (3) and (4) to predict $\hat{x}_{ij}$ and $\hat{y}_{ij}$.

ADMIRE is implemented in a easy-to-use python package called adadmire which is listed in the python package index PyPi.

3 Simulations



Rank	Sample	Feature	Observed	Random
1	203	Behavior	8.916	5.356
2	391	Treatment	8.456	4.466
3	172	Behavior	6.603	4.024
4	106	Genotype	5.153	3.719
5	290	Genotype	4.349	3.45
6	24	Treatment	4.096	3.255

Rank	Sample	Feature	Observed	Random
7	204	Genotype	4.019	3.091
8	233	Genotype	3.954	2.963
9	146	Treatment	3.419	2.844
10	250	Genotype	2.952	2.748
11	22	Treatment	2.605	2.646
12	113	Genotype	2.399	2.565

A
B
Samples with S/C behavior
count
Probability of S/C behavior
Samples with C/S behavior
count
Probability of C/S behavior
C
Samples with control genotype
count
Probability of control genotype
Samples with trisomic genotype
count
Probability of trisomic genotype
D
Samples with Saline Treatment
count
Probability of Saline Treatment
Samples with Memantine treatment
count
Probability of Memantine treatment
0.00
0.25
0.50
0.75
1.00

Fig. 2. Observed and random scores for the data set containing artificial discrete anomalies and estimated probabilities for the categorical variables split in the respective binary states across the according samples.
A) Highest ranking observed and random scores, artificial anomalies are marked in red, the threshold is marked in green.
B) Estimated probabilities for behavior (C/S or S/C)
C) Estimated probabilities for genotype (control/trisomic)
D) Estimated probabilities for treatment with treatment either Memantine or Saline

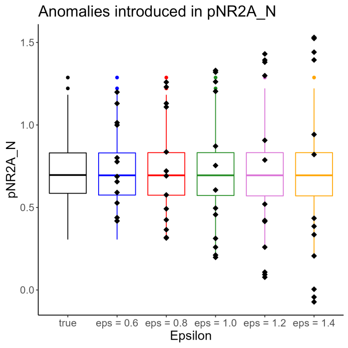


Fig. 3. Influence of the parameter ϵ on the strength of the anomalies in protein pNR2A_N. Black dots indicate introduced anomalies.

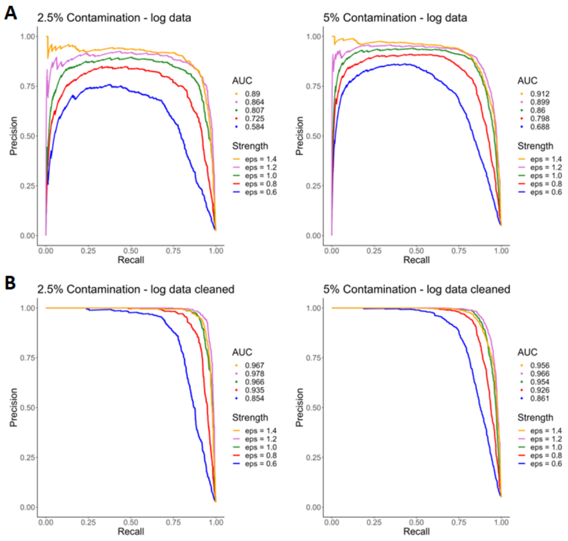


Fig. 4. Precision-Recall curves for the simulations with 2.5% and 5% contamination:
A) PR curves of ADMIRE on log-transformed data without correcting for intrinsic outliers
B) PR curves of ADMIRE on log-transformed simulations corrected for intrinsic outliers

We studied the performance of ADMIRE by simulating artificial anomalies in a proteomics data set (Higuera *et al.*, 2015). The data set consists of protein expression levels from the brains of mice with and without Down syndrome. In total 77 proteins (continuous features) were measured using reverse phase protein arrays (RPPA) in several groups of mice that can be characterized by three discrete features: genotype (normal/trisomic), treatment (saline/memantine) and behavior: a protocol used to stimulate learning (shock-context/context-shock). In total, 72 mice were analyzed with 3 replicates in a five-point dilution series resulting in 1,080 measurements per protein. Each measurement can be considered as an independent sample. Since the focus of this study is the evaluation of ADMIRE's anomaly detection and correction we excluded 12 proteins because they contained missing values. Extensive performance evaluation of ADMIRE's imputation routine can be found in the supplement. Furthermore, we sub-sampled 400 samples such that each of the 8 different groups of mice was represented by 50 samples. This resulted in a data set of 400 samples, 68 continuous features, 3 discrete features and 400*68 = 27,200 continuous and 3*400 = 1200 discrete data points. In the following analyses we used the log-transformed protein measurements. Further information on the data set can be found in the supplement.

3.1 Anomaly Detection

To validate the detection of discrete anomalies, we introduced artificial anomalies by changing the original states of the discrete features. For each feature we chose two samples and swapped the according states, e.g. a sample with original treatment "Saline" was assigned to the other treatment state "Memantine". Thereby, we introduced six artificial anomalies in the data set.

ADMIRE detects among the 1200 discrete data points 10 anomalies. Figure 2A) reports the 12 discrete data points with highest ranking observed scores. Additionally, we reported for each rank the corresponding calculated random score. In green we marked the threshold for anomaly detection, where the random score exceeds the equally ranking observed score. The rows marked in red correspond to the artificially introduced anomalies. As can be seen, all six artificially introduced anomalies are detected by ADMIRE. The other detected anomalies cannot be verified since the data set was not generated by us. Figure 2B-D additionally show the estimated probabilities for the three features split in their corresponding states. Overall, high probabilities (low scores) were computed for all data points, except for the samples where the state was swapped (marked in red).

To study anomaly detection in continuous data points we introduced artificial anomalies similar as in (Steinbuss and Böhm, 2017). We randomly choose n_a data points and perturb them by adding random shifts. The size of the shifts is relative to the normal range of the feature and can be calibrated by a parameter ϵ. For $\epsilon < 1$ the perturbed data does not exceed the range of the feature and thus does not present an outlier. For larger values of ϵ, the perturbations can introduce outlier values as well. In addition, our simulation ensures that every chosen data point is perturbed by at least 15%. Details on the simulation can be found in the supplement. For illustration, Figure 3 shows the distribution of artificial anomalies introduced in the data of the protein pNR2A_N for different values of ϵ. We run 10 simulation scenarios varying the number of introduced anomalies and their strengths ϵ. We either introduced 2.5% anomalies (corresponding to 680 perturbed data points) or 5% (corresponding to 1360 perturbed data points) and also varied the strength ϵ of the introduced anomalies. In Supplemental Table 2 we summarized the 10 simulations.

The algorithm shows good performance in the detection of anomalies with an area under the curve of 0.890 for a contamination level of 2.5% and of 0.912 for 5% contamination and ϵ set in both cases to 1.4. With decreasing ϵ (1.2 - 0.6) the magnitude of the anomalies decreases and the number of hidden anomalies increases. Therefore, the anomalies are harder to detect, which is reflected in lower AUCs. Nevertheless the detection of anomalies remains good with AUCs ranging from 0.864 to 0.584 for 2.5% contamination and 0.899 to 0.688 for 5% of contamination (see Figure 4A). Note, that we did not adjust the proteomics data for intrinsic anomalies that might exist in addition to the simulated ones. If we did identify these anomalies using ADMIRE and adjust the PR curves for them (see Figure 4B), the performance increases further, with AUCs now ranging from 0.978 to 0.854 for 2.5 % of contamination and 0.966 to 0.861 for 5 % contamination. Further information on the detection of intrinsic anomalies can be found in the supplement.

Finally, we compared ADMIRE to three competing outlier detection algorithms: Isolation Forest (Liu *et al.*, 2008), LOF (Breunig *et al.*, 2000) and stray (Talagala *et al.*, 2021) in the context of the 10 simulations described above. Since these methods aim at finding anomalous instances in a data set, we applied them feature-wise. Our algorithm outperforms all methods, which reached only maximal AUCs of 0.63 and 0.747 for 2.5% and 5% contamination (stray) and 0.701 and 0.789 (LOF) on the log-transformed simulations. Isolation Forest performed best on the scaled raw data with AUCs up to 0.828 for 2.5% and 0.888 for 5% contamination. Further information on how Isolation Forest, LOF and stray were applied can be found in the supplement, together with the precision recall curves after correcting for the intrinsic anomalies.

3.2 Anomaly Correction

Here we study how ADMIRE performs in correcting detected anomalies. For the 10 simulations described above, we calculated anomaly thresholds and corrected all data points by replacing them with their re-estimated values (3). We next compared both the uncorrected (perturbed) and corrected data to the original data (ground truth) and calculate mean absolute percentage errors for both (Table 1). Anomaly correction reduced theses errors strongly, showing that the algorithm automatically can improve the quality of data sets significantly. Note that correction was applied to all detected anomalies including the falsely detected ones, suggesting that in case of false positive detections the corrections do not compromise the data very much.

Data set	ϵ	**# Introduced**	**# Detected**	**TP**	$\mathbf{MAPE_i}$	$\mathbf{MAPE_c}$
S_1	$\epsilon = 0.6$	680	856	504	1.047%	0.823%
S_2	$\epsilon = 0.6$	1360	1244	927	2.073%	1.22%
S_3	$\epsilon = 0.8$	680	812	566	1.396%	0.747%
S_4	$\epsilon = 0.8$	1360	1244	1040	2.763%	1.139%
S_5	$\epsilon = 1.0$	680	763	596	1.746%	0.676%
S_6	$\epsilon = 1.0$	1360	1241	1103	3.454%	1.119%
S_7	$\epsilon = 1.2$	680	729	606	2.095%	0.65%
S_8	$\epsilon = 1.2$	1360	1198	1093	4.145%	1.23%
S_9	$\epsilon = 1.4$	680	666	578	2.444%	0.73%
S_{10}	$\epsilon = 1.4$	1360	1072	1008	4.836%	1.473%

Table 1. Summary of the corrected data sets. The table shows the strength of the simulation (ϵ), the number of introduced anomalies (column "# Introduced"), the number of detected continuous anomalies (column "# Detected") and the number of true positive anomalies among the detected ones (TP), the by the anomaly simulation introduced mean average percentage error (MAPE$_i$) and the mean average percentage error after correcting the data sets with ADMIRE (MAPE$_c$).

4 Anomaly detection in metabolomics data

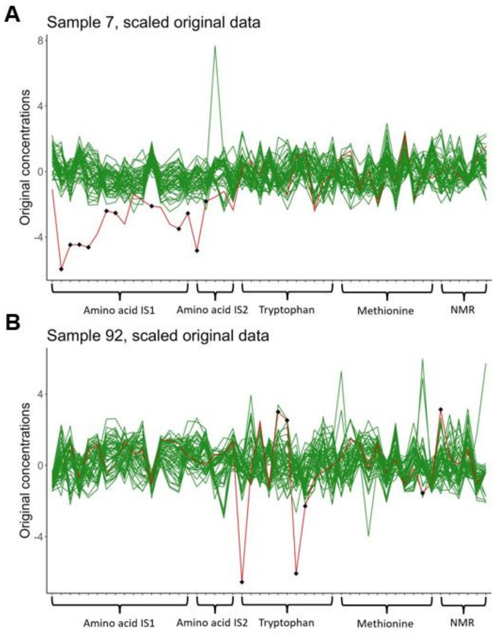


Fig. 5.

A) Scaled, originally measured concentrations of sample 7 (red) with all other samples in the same MYC group (green), detected anomalies are marked as black diamonds. The features (metabolites) on the x-axis are ordered according to the different quantification methods.

B) Scaled, originally measured concentrations of sample 92 (red) with all other samples in the same MYC group (green), detected anomalies are marked as black diamonds.. The features (metabolites) on the x-axis are ordered according to the different quantification methods.

We used ADMIRE to investigate anomalies in one of our own metabolomics data sets (Feist, M. et al., 2018). This data was generated to study the metabolism of B-cells in response to stimuli from a tumor micro-environment. In particular, we were interested how the responses changed when the oncogene MYC was activated. MYC activation is a hallmark of many B-cell lymphomas. We used human P493/6 B-cells that contain an inducible MYC-construct and stimulated them with different cocktails of micro-environmental factors. Their metabolism responded to these stimuli and we profiled these changes using both nuclear magnetic resonance (NMR) spectroscopy and mass spectrometry (MS) applied to the cell cultures' supernatants and cell pellets, which were both independently measured. Note that in the previous paper by Feist et al., 2018 only cell pellet data were evaluated, while the present contribution focuses on the data obtained from the corresponding supernatants.

Continuous features consist of 49 metabolites that were quantified in a total of 100 samples. 11 features were measured using NMR and 38 using MS. The discrete features are the MYC status (high/low) of the B-cells and the 10 batches in which the samples were processed.

We run ADMIRE on the full data set including both continuous and discrete variables. First, we checked for discrete anomalies. These could be manual data entry mistakes such as misassignments of either the MYC-status or one of the batches. Supplemental Figure 8A shows observed scores next to rank matching random scores for the 10 top scoring discrete data points. No observed score exceeds the random score and we conclude that all discrete features are correct. Artificially introduced errors, similar to section 3, were detected correctly, see supplement.

Next, we studied potential anomalies in the continues metabolite measurements. Our algorithm flaged 46 out of 4,900 continuous data points as anomalies (0.94%). The flagged anomalies are distributed uniformly across the 49 features with mostly only one anomaly per feature, indicating that there are no globally conspicuous features. However, if we mapped anomalies to samples, a different distribution was observed. Supplemental Figure 9 shows that while most samples contain only a small number of anomalies (75% of the sample do not even have an anomaly at all), two samples show significantly more. In sample 7, ADMIRE flagged 11 out of the 49 continuous features as anomalies and in sample 92 a total number of 7 features were flagged.

Figure 5A shows sample 7 (red) together with all samples of the same MYC state (green lines). The black diamonds are the anomalies detected by ADMIRE. All anomalies are in the first two blocks, which correspond to the metabolites that where quantified by mass spectrometry (MS). All of them were amino acids. To verify that the detected anomalies are genuine errors, we quantified them again using NMR, a completely independent method. This was possible for 10 out of 11 flagged features. Only for cystine NMR signals were too low and highly overlapping such that no NMR measurement was possible. For the remaining 10 metabolites NMR confirmed that the MS based measurements were in fact incorrect, deviating by more than 15% from the corresponding NMR measurements. We suspect that a pipetting mistake in the probe preparation for amino acid mass spectrometry is responsible for the anomalies ADMIRE found. Metabolites were quantified relative to added internal standards with different separate standard mixes for amino acids and tryptophan and therefore, any pipetting error in the standard will falsify results for this specific measurement type. Further note that for each measurement method such as the amino acid method or the tryptophan method a separate internal standard mix was used. As a consequence, a pipetting error can be detected using NMR as a validation method since it uses a different internal standard and is, therefore, not affected. This shows nicely the potential of the MGM for detecting true anomalies and also patterns of anomalies within a sample. For the validation of the anomaly correction, we calculated the mean absolute percentage error for the 11 anomalies of sample 7 with clear NMR signals. Hereby we used for cystine, that could not be validated by NMR, the originally measured concentration. The MAPE between the originally measured and validated values is reduced from 76.63 to 12.27 when the originally measured values were replaced by the corrections proposed by ADMIRE.

Sample	Metabolite	Score	Corrected	Original	Validated
31	Hippuric acid	21.05	5.52e-05	3.20e-03	4.31e-04
29	Spermidine	13.81	1.08e-05	9.30e-05	9.30e-05
7	Aspartate	12.178	-1.14e-02	-7.30e-02	8.77e-05
92	Anthranilic acid	11.27	8.076e-07	-2.12e-05	-2.12e-05*
7	Glutamate	10.19	0.128	-0.151	0.142
90	Acetone	10.16	-9.37e-04	4.73e-03	3.91e-03
93	Succinic acid	8.73	-3.88e-03	2.05e-02	-1.47e-02
7	Glycine	8.70	-1.91e-02	-8.25e-02	-6.25e-02
92	Kynurenic acid	8.39	-2.41e-06	-7.35e-06	-7.35e-06*
7	Proline	8.22	0.103	-2.95e-02	0.137
7	Asparagine	8.09	-5.29e-02	-1.43e-01	-2.31e-02
99	Succinic acid	6.25	-3.88e-03	1.57e-02	-1.49e-02
89	4-Hydroxyproline	5.70	-8.32e-03	-2.79e-02	1.48e-04
92	Indole-lactic acid	5.69	-7.28e-06	2.93e-08	2.93e-08*
7	Cystine	5.55	-2.44e-02	-4.26e-02	-4.26e-02*
7	Tyrosine	5.17	-7.50e-02	-0.102	-6.45e-02
7	Leucine	5.10	-0.190	-0.261	-0.211
92	Kynurenine	4.77	-3.15e-04	-7.93e-04	-7.93e-04*
80	3-Hydroxyanthranilic acid	4.56	2.11e-05	3.84e-05	3.84e-05*
89	Acetone	4.47	-7.967e-04	2.46e-03	1.71e-03
80	Adenosine	4.22	2.37e-05	4.80e-05	4.80e-05*
92	Indole-3-acetic acid	4.21	6.01e-06	4.03e-06	4.03e-06*
29	S-Adenosylmethionine	4.14	1.51e-05	3.89e-05	3.89e-05*
15	Ornithine	3.93	-2.31e-02	2.56e-02	2.56e-02
3	Kynurenic acid	3.90	1.12e-07	2.28e-06	2.28e-06*
64	Ornithine	3.83	-5.03e-03	3.00e-02	-1.58e-02
7	Glutamine	3.75	-0.762	-1.16	-0.87
66	Formic acid	3.74	6.33e-02	3.80e-02	6.98e-02
7	Isoleucine	3.72	-0.183	-0.237	-0.17
64	Tryptophan	3.66	-2.83e-03	6.14e-04	-1.84e-03
92	Succinic acid	3.65	-3.60e-04	-1.25e-02	-1.25e-02
2	Putrescine	3.56	2.52e-04	3.99e-04	3.99e-04*
74	Ornithine	3.35	-5.84e-03	2.51e-02	-6.45e-03
98	Putrescine	3.25	2.09e-04	3.44e-04	2.90e-04
92	Ornithine	3.21	6.69e-03	3.70e-02	1.52e-02
84	S-Adenosylmethionine	3.16	2.48e-05	4.21e-05	4.21e-05*
18	Phenylalanine	3.12	1.15e-02	6.0e-02	2.89e-02
56	5-Methylthioadenosine	3.09	5.51e-07	2.12e-06	2.12e-06*
87	5-Hydroxy-indole-acetic acid	3.07	7.35e-06	-3.20e-06	-3.20e-06*
52	Adenosine	2.98	2.72e-05	4.48e-05	4.48e-05*
73	3-Hydroxyanthranilic acid	2.98	1.27e-05	-4.19e-07	-4.19e-07*
7	Valine	2.93	-0.117	-0.145	-0.112
32	Adenosine	2.78	3.14e-05	4.81e-05	4.81e-05*
3	Lysine	2.72	-1.70e-02	1.24e-02	-1.350e-02
13	Ornithine	2.69	1.22e-02	1.33e-02	1.33e-02
79	5-Methylthioadenosine	2.63	9.264e-07	2.27e-06	2.27e-06*

Table 2. Detected anomalies ordered according to their scores. Rows marked in green are anomalies that could be validated as true anomalies. Red corresponds to measurements that either show no conspicuous MS spectra but could not be validated by an independent method (marked with *) or false positives where the original measurement is correct. Yellow corresponds to the anomalies in sample 92 that could not be validated by an independent method and purple to anomalies where the original data contained a large amount of imputed values. All concentration values are given in mM.

The sample with the second highest amount of anomalies is sample 92. In this sample ADMIRE detected seven anomalies. Only two of these could be quantified by NMR (one false positive and one true anomaly). For the other metabolites NMR signals were too low and overlapping for accurate quantification. Figure 5B shows sample 92 together with all other samples of the same MYC state. The anomalies are mostly located in the tryptophan group of measurements, which was independently measured employing a dedicated MS method (see supplement for details). Again, this points to a possible pipetting error during sample preparation. Most probably, the sample volume used for the tryptophan method was incorrect.

For the remaining flagged anomalies we inspected the raw spectra and searched for deviations or errors in the integration of the single spectra. Whenever possible, we validated MS measurements by reanalyzing the correspondent NMR spectra. This is only possible for metabolites with concentrations up to a lower limit of micromolecular range. For smaller concentrations the sensitivity of the NMR is not sufficient enough to quantify reliably. Table 2 reports all 46 anomalies sorted by their anomaly score. The last three columns show the corrections proposed by ADMIRE, the originally measured value (original) and the validated, true measurement (validated), respectively. All anomalies that could be unambiguously validated as anomalies are highlighted in green. For them, the difference between the original and the verified value was at least 15%. False positives, where ADMIRE detected an anomaly but the verification showed no erroneous measurement or other peculiarity are marked in red. Note that we treated metabolites that couldn't be verified by an independent method and whose spectra showed no abnormalties also as false positives. These anomalies are marked with an asterisk. The rows highlighted in yellow correspond to the anomalies of sample 92 which all belong to the tryptophan measurement group. Here, we couldn't verify an error in the measurement, but a mishap during sample generation similar to sample 7 is likely. Two anomalies belonging to the features Spermidine and 3-Hydroxyanthranilic acid are marked in purple. We included these two features although both contained a large number of imputed values and measurements below the lower limit of quantification. Note that these values were not imputed by ADMIRE but preprocessed using the laboratory's own pipeline.

We calculated for the 46 validated data points in Table 2 the MAPE between the original measured concentrations and the validated concentrations and compared it to the MAPE between the corrections proposed by ADMIRE and the validated ones. Using the corrected concentrations the MAPE decreased from 23.015 to 10.802, which is an almost 2.5 fold improvement. Again, the false positive anomalies were included in the calculation of the MAPE. This shows once more that even if ADMIRE detects a false positive anomaly, its correction is still close to the original, true value.

5 Discussion

Incorrect data points make data analysis invalid, even if they are infrequent. In large data sets they are hard to detect manually, but easier to detect automatically because they are inconsistent with the inherent structure of the rest of the data. Here we describe ADMIRE, an algorithm that combines Mixed Graphical Models and cross validated re-estimation of data points to detect data anomalies in large mixed molecular data sets. The MGM learns inherent data structure, the CV based re-estimation checks whether individual data points are consistent with this data structure.

Outliers are a special instance of anomalies. An outlier is a value of a feature that is suspiciously higher or lower than all other values of the same feature. In general, they are more easily detected. Although we can in principal detect them feature by feature independently from all other features, the use of conditional distributions can nevertheless support the process. Importantly, anomalies do not need to present as univariate outliers and in fact many of the anomalies we detected did not.

ADMIRE was primarily designed for molecular data sets that combine continuous features such as abundance of certain molecules (OMICS data) with discrete features that for example describe experimental designs or patient characteristics. Here, incorrect data in continuous features can result from experimental artifacts, while incorrect discrete data can be caused by incorrect manual data entry. However, ADMIRE can be used for any large data set continuous, discrete or mixed.

ADMIRE does not only detect anomalies, it also has routines to correct them thus generating more consistent data sets. In this way it can be used as a pre-processing or data normalization routine as well. Additionally, the adadmire package offers a testing routine that allows the user to test ADMIRE in simulations with their own data. Finally, anomalies do not need to be incorrect data points. They can also be observations that are rare, unusual but correct. Such oddities can be scientifically interesting and ADMIRE can be used to spot them for further investigation. In this way it can be used as a data mining tool as well.

6 Funding

This study was funded by the Bavarian Ministry of Science and the Arts within the Bavarian Research Network bayresq.net (project Metabodefense) and the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), project number 509149993, TRR 374. HUZ and MA are supported by the Federal Ministry of Education and Research (Bundesministerium für Bildung und Forschung; BMBF) within the framework of the e:Med research and funding concept (grants 01ZX1912A and 01ZX1912C).

References

Altenbuchinger, M. et al (2019). A multi-source data integration approach reveals novel associations between metabolites and renal outcomes in the german chronic kidney disease study. *Scientific Reports*, **9**(1), 13954.

Altenbuchinger, M. et al (2020). Gaussian and mixed graphical models as (multi-)omics data analysis tools. *Biochimica et Biophysica Acta (BBA) - Gene Regulatory Mechanisms*, **1863**(6), 194418.

Ando, S. (2007). Clustering needles in a haystack: An information theoretic analysis of minority and outlier detection. *Seventh IEEE International Conference on Data Mining (ICDM 2007)*.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., and Sander, J. (2000). Lof. *ACM SIGMOD Record*, **29**(2), 93–104.

Cheng, J., Li, T., Levina, E., and Zhu, J. (2017). High-dimensional mixed graphical models. *Journal of Computational and Graphical Statistics*, **26**(2), 367–378.

Chun, H., Chen, M., Li, B., and Zhao, H. (2013). Joint conditional gaussian graphical models with multiple sources of genomic data. *Frontiers in Genetics*, **4**, 294.

DeCoste, D. and Levine, M. B. (2004). Automated event detection in space instruments: A case study using ipex-2 data and support vector machines. *Artif Intell Rev*, **22**, 85–126.

Feist, M. et al. (2018). Cooperative stat/nf-kb signaling regulates lymphoma metabolic reprogramming and aberrant got2 expression. *Nature Communications*, **9**(1514).

Grubbs, F. E. (1969). Procedures for detecting outlying observations in samples. *Technometrics*, **11**(1), 1–21.

Higuera, C., Gardiner, K. J., and Cios, K. J. (2015). Self-organizing feature maps identify proteins critical to learning in a mouse model of down syndrome. *PLOS ONE*, **10**(6), e0129126, 1–28.

Hodge, V. and Austin, J. (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review*, **22**(2), 85–126.

John, G. H. (1995). Robust decision trees: Removing outliers from databases. In *In Knowledge Discovery and Data Mining*, pages 174–179. AAAI Press.

Korn, F. et al (2001). Quantifiable data mining using principal component analysis. *Technical report CS-TR-3754, Institute for Systems Research, University of Maryland, College Park, MD*.

Lauritzen, S. L. (1996). *Graphical Models*. Clarendon Press, Oxford.

Lee, J. D. and Hastie, T. J. (2015). Learning the structure of mixed graphical models. *Journal of Computational and Graphical Statistics*, **24**(1), 230–253.

Liu, F. T., Ting, K. M., and Zhou, Z.-H. (2008). Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the Lasso. *The Annals of Statistics*, **34**(3), 1436 – 1462.

O'Donoghue, B. and Candès, E. (2013). Adaptive restart for accelerated gradient schemes. *Foundations of Computational Mathematics*, **15**(3), 715–732.

Steinbuss, G. and Böhm, K. (2017). Hiding outliers in high-dimensional data spaces. *International Journal of Data Science and Analytics*, **4**(3), 173–189.

Talagala, P. D., Hyndman, R. J., and Smith-Miles, K. (2021). Anomaly detection in high-dimensional data. *Journal of Computational and Graphical Statistics*, **30**(2), 360–374.

Wang, T. et al (2016). FastGGM: An Efficient Algorithm for the Inference of Gaussian Graphical Model in Biological Networks. *PLOS Computational Biology*, **12**(2), e1004755,1–16.

Zhao, H. and Duan, Z.-H. (2019). Cancer Genetic Network Inference Using Gaussian Graphical Models. *Bioinformatics and biology insights*, **13**, 1177932219839402.