Freie Universität Berlin

Habilitationsschrift

Robust analysis of high-dimensional omics data using computer simulation and graphical visualization

zur Erlangung der Lehrbefähigung
für das Fach Bioinformatik

vorgelegt dem Fachbereichs Mathematik und Informatik
Freie Universität Berlin

von

Dr. rer. hum. biol. Jochen Kruppa-Scheetz

Eingereicht:	11/2019
Dekan:	Prof. Dr. Günther Rothe
1. Gutachter/in:	PD Dr. Tim Conrad
2. Gutachter/in:	Prof. Dr. Knut Reinert
Tag des mündlichen Vortrags:	13. Mai 2024

Contents

Index of abbreviations

AUC Area under the curve i.e under the receiver operating curve (ROC)

AMINO Amino acid

bp Base pairs

CLR C-type lectin receptors

cov Covariance

CpG Sites of DNA where a cytosine is followed by a guanine

EPE Expected prediction error

Feature Covariate or risk factor in machine learning

FN False negative or type II error or β error

FP False positive or type I error or α error

GWAS Genome-wide association study

k-mer Substrings of length k of a string

n Sample size i.e. a group of patient of size n

NGS Next generation sequencing

MSE Mean square error

p Covariate or risk factor of a regression model

ppv Positive predictive value

Pr Probability

read Short DNA string of 50bp to 300bp

RMSE Root mean square error

ROC Receiver operating curve

Σ Covariance matrix

sens Sensitivity

SNP Single nucleotide polymorphism

spec Specificity

tidyverse R package

TN True negative

TP True positive

var Variance

> [...] which robust/resistant methods you use is not important-what is important is that you use some. It is perfectly proper to use both classical and robust/resistant methods routinely, and only worry when they differ enough to matter. But when they differ, you should think hard.
> Tukey, J. W. Huber (2002) taken from Tukey (1979)

1 Introduction

1.1 Robust statistical methods

In the recent years science has faced major struggles. One the one hand the discussion of "fake news" in the social media and the answer of science to the search of the "truth" has emerged. This discussion stands in contrast to the overall falsification principle of hypothesis testing. We do not prove a hypothesis, we reject a hypothesis. On the other hand the crisis of reproducibility in science is a major topic in many journals today. The journal of The American Statistician (Vol. 73, 2019) has come up with the issue topic "Statistical Inference in the 21st Century: A World Beyond p $<$ 0.05" and Wasserstein et al. (2016) discussed beforehand to abandon the p-value entirely. All three topics, fake news, reproducibility, and the discussion of p-values, face the demand of robust statistical analysis, which makes scientific findings reliable for the public. Therefore, the question emerged, how can statistical methods help to translate basic research into the clinical routine? In this work, I will discuss different approaches to validate new statistical methods and make these algorithms more robust in real world data settings.

The translation form basic research findings to the patient, from bench to bedside, is a important aim in clinical research (Woolf, 2008). Four steps, T1 to T4, must be taken in the actual 4T model for clinical translation for the full translation of a basic research finding to the clinical application: discovery from basic research or basic knowledge on potential clinical application (T1) to evidence based guidelines or efficacy knowledge (T2) to clinical care or intervention and applied knowledge (T3) to finally the health of a community or population into the health knowledge (T4). Biometry and statistical bioinfomatics can take part in these processes by presenting and using robust methods and tools for quality assessment of clinical research studies. A statistical method should be robust and therefore help to reproduce findings of experiments (Maronna et al., 2018). A well known example of a robust method is the median $\tilde{x}$ for the calculation of the "middle" of a set of numbers. In difference to the arithmetic mean $\bar{x}$, the median is robust against outliers. Therefore, if a very high number is included, the mean will be biased while the median changes only slightly.

More data, denoted as $\boldsymbol{X}$, is produced nowadays than ever before. This data processing is also covered by the naming "big data". The storage of big data is even called "data lakes" to describe the enormous amount of data. The problematics of data handling and visualization has given birth to a new discipline, the data scientist or data analyst, which works mainly on the data processing. In classical biometry big data was first introduced by the micro array technology in genetics. At the time of the late 90's the problem was named high dimensional data or $p \gg n$

problem ("p larger n problem"). The first time in biometry more parameters (p) where available than sample size n. The rule of thumb demanded 10 samples per factor to use a regression model, hence with genetic data including over 500,000 genetic variants and only five samples, in the early days, and up to 10,000 samples nowadays, the standard model approaches did not work. Figure 1 shows the results of a genome wide association study (GWAS) on rheumatoid arthritis. The data as been used in Kruppa et al. (2012) for the prediction and classification of rheumatoid arthritis patients by machine learning algorithms. The data consist of 506,665 SNP but only $\sim$ 2000 patients.

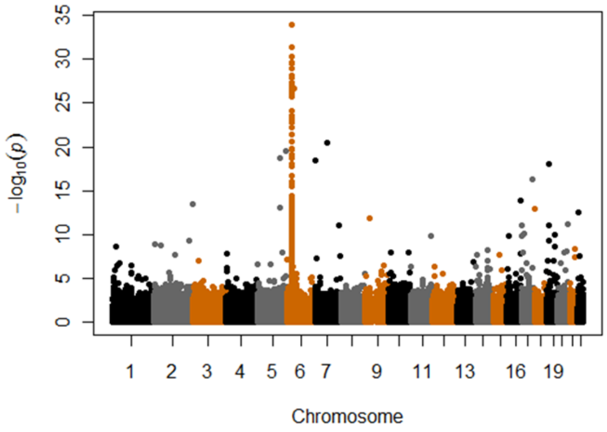

Figure 1 – Manhattan plot of a GWAS on rheumatoid arthritis with 506,665 SNPs plotted after quality control. The data set consists of 868 cases and 1,194 controls. The x-axis shows the position of the single SNP on the genome; the y-axis shows the $-\log_{10}$ transformed p-value of the SNP. Taken from Kruppa, J. *et al.* (2012). Risk estimation and risk prediction using machine-learning methods. Human genetics, 131(10), 1639-1654.

1.2 The inflation of false positive findings

Ioannidis (2005) stated, that most scientific findings are false. What is the core of this statement and how is a false positive finding defined in classical hypothesis testing? The inflation of the type I error (α error; false positive findings) becomes the first time really challenging and problematic in genetics with the large amount of markers to test. We assume stochastically independent tests with k null and alternative hypotheses, where in fact all null hypotheses are valid. Then, we test all null hypotheses to a local level $\alpha = 0.05$. The probability that at least one false null hypothesis will be rejected and we will find one false positive is $1 - (1 - \alpha)^k$. Hence, if 50 hypotheses are tested, $1 - (1 - 0.05)^{50} = 0.92 \approx 100\%$, the probability of making at least one wrong test decision is almost 100%. In the above shown GWAS on rheumatoid arthritis (Figure 1) with 500,000 single nucleotide polymorphisms (SNP) we will find roughly 25,000 false positive SNPs with a significant difference though the null hypothesis is true and there is no association. Hence, we will assume the finding of significant associations between the genotype and the disease even though there is no dependency in the real world. How can we control the type I error, also known as a "false positive" or the type II error, also known as a "false negative"? In hypotheses testing the type I error will be controlled. Normally, the Bonferroni adjustment and the Benjamini Hochberg (FDR) are suggested (Thissen et al., 2002). Most of the the time the Bonferroni adjustment of the type I error is to conservative, hence less than 5% of the Null hypothesis are rejected. The principle idea of FDR is to sort the p-values and reject the null hypothesis in the sense of a hierarchical testing principle. The used threshold Q has not to be 5% and allows a more flexible and liberal results.

To overcome these multiple testing problem and hold an overall α-level of 5% for all comparisons Hothorn et al. (2008) introduced simultaneous contrast test and simultaneous confidence intervals. The analysis is possible in R package `multcomp` and is based on the multivariate t distribution. The multivariate t distribution allows generate critical t values for the decision against the null hypothesis by taking into account the correlation between t statistics (Mi et al., 2009). In a pairwise comparison different treatment groups will be compared frequently and therefore the test statistic is not independent anymore. In my master's thesis I applied these approach on linear mixed models to control the false positive findings (Kruppa, 2009). To model the data in a correct way and adjust for confounder and after wards for possible inflation of the false positive findings, we must understand the structure of the data and the statistical summary statistics. The understanding of the correlation structure between the t test statistics was the key to solve the multiple comparison problem.

1.3 The structure of omics data

Due the large amount of data, the analysis pattern in genetics has also overall changed. The new idea of a analysis pipeline was born (Leipzig, 2017). A bioinformatic pipeline consists of many sequential algorithms, which can be single R, Perl, or C++ functions or even whole stand alone statistical programs. Each step of the pipeline needs some input data, processed the data and gives the output to the next step of the pipeline. Often the data is stored into different files to keep redundancies low and speed the analysis (Köster and Rahmann, 2012). Hence, each software has its own file needs, like the standard genome wide association study (GWAS) analysis tool kit PLINK (Purcell et al., 2007), which uses a special type of input data with defined column names and structure. Often these data types are stored in binary data format, which saves hard disk space. In addition, binary files can processed very fast but are not human readable anymore.

The equation 1 shows the typical transposed genetic data structure $\mathbf{X}^T$. In contrast to the typical data structure of biometrical data, where each row is a patient, the data matrix is transposed for the analysis of high dimensional data (Aulchenko et al., 2007). The main reason is computational power. It is easier to add a row than to add a column. This different data structure causes many adjustments in the analysis pipelines. Standard statistical methods can not process such data and therefore wrapper must be written to transform the data for each analysis algorithm. The most important consequence of the transposed data is the usage of two data sets: one data set containing only the genetic information per variant and sample. Further, an additional phenotype data set, where the information on each single sample is saved. A typical transposed genetic data set $\boldsymbol{X}^T$ with n patients and p SNPs for a GWAS looks like the following.

$$\mathbf{X}^T = \begin{array}{c} \\ \text{SNP}_1 \\ \text{SNP}_2 \\ \vdots \\ \text{SNP}_p \end{array} \begin{array}{c} \begin{array}{cccc} \text{Patient}_1 & \text{Patient}_2 & \cdots & \text{Patient}_n \end{array} \\ \begin{pmatrix} x_{11} & x_{12} & & x_{1n} \\ x_{21} & x_{22} & & x_{2n} \\ & & \ddots & \\ x_{p1} & x_{p2} & & x_{pn} \end{pmatrix} \end{array} \tag{1}$$

It is important to remember the transposed structure of the data. Nevertheless, all summary statistics are calculated on the original data structure $\boldsymbol{X}$ but the data is stored in the transposed state $\boldsymbol{X}^T$. This is especially important for the calculation of the variance/covariance matrix, which describes the dependencies or correlation between the columns of the data set.

Correlation and covariance are measure both the relationship or dependency between two variables x_1 and x_2 or in our example SNP$_1$ and SNP$_2$. The covariance describes the direction of a linear relationship of the two variables, while the correlation measures the strength and the direction of the linear relationship of x_1 and x_2. The correlation is a function of the covariance. The covariance is calculated, like the variance on the quadratic scale of the variables, while the correlation is the standardized covariance matrix. We can derive the correlation matrix by dividing the covariance matrix by the product of corresponding standard deviations. The correlation runs threfore from -1 to 1 while the covariance from $-\infty$ to $+\infty$. Both, the correlation matrix and the covariance matrix, have a size of $p \times p$, determined by the columns of the data set. The covariance matrix is symmetric, therefore $cov(x_1, x_2)$ is the same as $cov(x_2, x_1)$. The covariance of the diagonal is the variance of the variable.

$$\boldsymbol{\Sigma} = \begin{array}{c} \\ \text{SNP}_1 \\ \text{SNP}_2 \\ \vdots \\ \text{SNP}_p \end{array} \begin{array}{c} \begin{array}{cccc} \text{SNP}_1 & \text{SNP}_2 & \cdots & \text{SNP}_p \end{array} \\ \begin{pmatrix} var(\text{SNP}_1, \text{SNP}_1) & cov(\text{SNP}_1, \text{SNP}_2) & & cov(\text{SNP}_1, \text{SNP}_p) \\ cov(\text{SNP}_2, \text{SNP}_1) & var(\text{SNP}_2, \text{SNP}_2) & & cov(\text{SNP}_2, \text{SNP}_p) \\ & & \ddots & \\ cov(\text{SNP}_p, \text{SNP}_1) & cov(\text{SNP}_p, \text{SNP}_2) & & var(\text{SNP}_p, \text{SNP}_p) \end{pmatrix} \end{array} \tag{2}$$

The covariance matrix $\boldsymbol{\Sigma}$ is important for the modeling of the data. In more simple words, statistics does not model mean differences but the dependencies between the covariates described by the covariance matrix. If we want to run a simulation to validate new statistical algorithms, we want to generate artificial data. If we generate data in the most simple case and ignore correlation between the samples and covariates our algorithm will most like not produce reproducible outcomes on real world data. Real world data is very often correlated and our simulation should somehow model correlated data.

1.4 Assessing in silico simulation studies

Simulation studies in biometry consist of many steps proposed by Burton et al. (2006). A more detailed description of setting up a simulation study delivers Kleijnen (2017) and Sanchez (2005). In the following a brief summary of the essentials is described. First, data must be generated fulfilling a given set of properties. Therefore, a outcome must be chosen and the distribution of the outcome must be determined. In genetics the decision is often made between a normal distributed outcome, like expression intensities of genes, a Poisson distributed endpoint, like mapped read counts or binomial distribution, like the presence of absence of a methylation region in epigenetics. After the outcome distribution is determined by the scientific background, the simulation model must be selected. In the most simple case, we can chose a normal distributed endpoint Y depending on one independent variable x_1 shown in equation 3.

$$Y_i = \beta_0 + \beta_1 x_i + \epsilon_i \tag{3}$$

Depending on the wanted effect of β_1 we can draw the outcome Y_i for each patient from a normal, Poisson or binomial distribution. The statistical software R supports different functions among others like `rnorm`, `rpois`, `rnbinom`, and `rbinom`, for the generation of data from the normal, Poisson, negative binomial, or binomial distribution. Therefore, we can easily generate data for a group comparison of 20 patients between placebo and treatment with an mean difference of $\beta_1 = 4$ by using $x_1 = (x_{placebo}, x_{treat})$ with $x_{placebo} = \mathcal{N}(10, \sigma_{placebo})$ and $x_{treat} = \mathcal{N}(14, \sigma_{treatment})$ assuming homogeneous variances between the groups ($\sigma_{placebo} = \sigma_{treatment}$) or heterogeneous variances ($\sigma_{placebo} \neq \sigma_{treatment}$). The model can be easily extended by adding further independent variables $x_2, ..., x_p$ resulting into $\boldsymbol{X} = x_1, ..., x_p$ and a vector of corresponding effects $\boldsymbol{\beta}$ shown in equation 4.

$$\boldsymbol{Y} = \boldsymbol{\beta X} + \boldsymbol{\epsilon} \tag{4}$$

If we want to model dependent variables $x_1, ..., x_p$, we use a multivariate distribution. This changes the data generation for $\boldsymbol{X}$. Hence, the data is now drawn from a multivariate distribution taking into account the correlation between the dependent variables by the covariance matrix $\boldsymbol{\Sigma}$ and predefined effects $\boldsymbol{\mu} = \mu_1, ..., \mu_p$ for each covariate.

$$\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \tag{5}$$

The multivariate normal distribution is technical available (Mi et al., 2009). If all $x_1, ..., x_p$ are independent the covariance matrix is a uniform matrix including 1's on the diagonal. In this special case, the $n \times p$-dimensional data generated by the multivariate normal distribution is the same as from single normal distributions for each p covariate and then combined to one data set.

In biology genetic factors like SNP's or alleles are not independent from each other. The genes are ordered on the chromosomes in a specific order and can not be seen as random. The genetic recombination in the meioses changes blocks of chromosomes by chromosomal crossover and therefore SNP's are organized in blocks of high correlation. Between the blocks, the correlation might be different. Hence, genetic factors are not inherited independently, they are in linkage disequilibrium. The multivariate distribution allows to model such dependencies between the genes $(x_1, ..., x_p)$ by the off-diagonal elements including the covariance of the genes.

If the outcome is somehow normal distributed, we can use the multivariate normal distribution and generate correlated data sets, like genes combined in a pathway have a higher correlation then genes in another pathway. Very often this is not the case. Next generation sequencing is based on the count of reads to a given part of a reference genome and epigenetic uses CpG sites with the percentage of methylation of the given CpG. For both cases, the Poisson distribution and the binomial distribution, no multivariate distribution to model dependent $\boldsymbol{X}$ is available. Kruppa et al. (2016) solves the generation of correlated count data of dependent gene sets included in biological pathways. In Kruppa et al. (2018b), we were able to use a genetic algorithm, as a Monte Carlo simulation, to generate binary correlated data holding the marginal correlation. Both methods mimic the data generation of a multivariate normal distribution for other distributions. The simulation can be even more complex like a circular frequency distribution shown in figure 2 by the Mease model for three categorical outcomes used in Kruppa et al. (2014a).

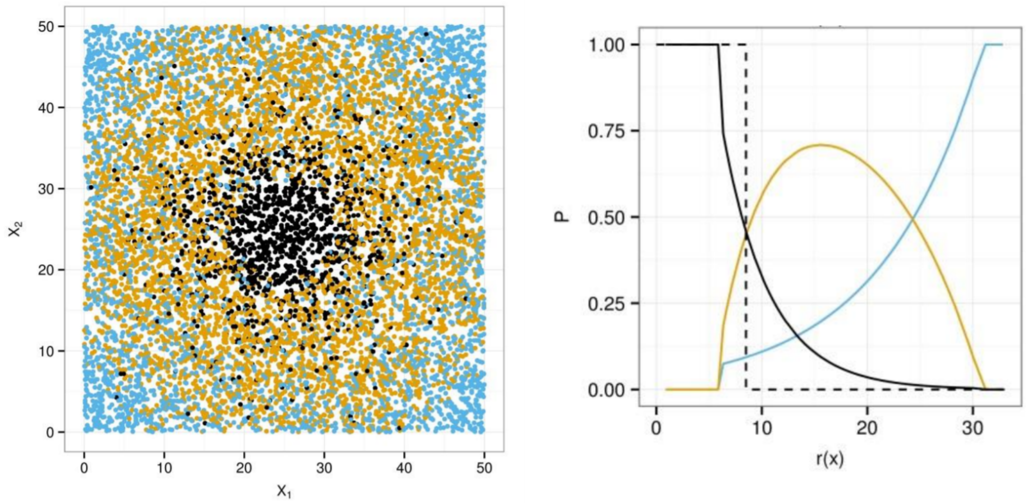

Figure 2 – On the left, the frequency distribution of the independent variables for the Mease model with three catagories. The frequencies follow a circular distribution. On the right the probabilities for the Mease model with decision line for the "black" dot class. Individuals left to the dashed lines are classified as "black" dots. Taken from Kruppa, J. *et al.* (2014). Probability estimation with machine learning methods for dichotomous and multicategory outcome: Theory. Biometrical Journal, 56(4), 534-563.

1.5 Dimension reduction of high dimensional data

Different methods have been developed to reduce the complexity of a data set. How many variables, $x_1, ..., x_p$, must be included into a model to achieve a good variance/bias trade off? If too many variables are included, the model will be to specific for the training data and will perform poorly on new validation data. Further, multivariate methods, like multiple regression and machine learning, have methodical limitation by including to much high correlated variables, i.e. variables with a high covariance, into the model (Libbrecht and Noble, 2015). We can use different approaches for the dimension reduction. One way is the classical approach using principle component analysis (PCA) or machine learning like random forests (Díaz-Uriarte and De Andres, 2006; Saeys et al., 2008). Especially, random forests can be used for generating smaller sets of important genes (Kruppa et al., 2012). Using these reduced list, we were able to predict the status of rheumatoid arthritis with a higher precision than the standard logistic regression approach.

The principle component analysis uses the $p \times p$-dimensional variance/covariance matrix $\mathbf{\Sigma}$ of the data and generates p principle components based on the data set. It is important to run the PCA on the z-transformed data space (Cheadle et al., 2003). The z-transformation is also called standardization, because the denominator includes the standard deviation. The PCA models the variance and assigns to each the first component as much variability as possible. A variable with a natural high variance due to the unit will have a higher load than variables with a natural smaller unit. To avoid such a bias, the z-transformation is used. In the case of machine learning a standardization is also advised (Lantz, 2013). Given x is a random variable with the expected value $E(x) = \mu$ and a variance of $Var(x) = \sigma^2$ with $\sigma = \sqrt{Var(x)}$ the standardized variable Z is calculated by the following.

$$Z = \frac{x - \mu}{\sigma} \tag{6}$$

The Z is now standard normal distributed with $Z \sim \mathcal{N}(0, 1)$. This transformation is repeated p times for each x in $\boldsymbol{X}$. In a real world data set, when the real data distribution is not known and therefore using the arithmetic mean and the empirical standard deviation, the standardization is called studentization. The studentization can be seen as the equivalent from the z-test to the t-test.

On the standardized data set X_Z we can now apply the principle component analysis. The central idea is to generate p principle components, called PCA_1 to PCA_p, each carrying as much variance as possible, starting with PCA_1. Hence, the principle component analysis tries to load maximal information on PCA_1, then PCA_2 and so on. We organize the information and can after wards drop the PCA's with low explained variance. As a drawback, PCA's are less interpretable and have no real meaning. The principle components methods uses the eigenvectors and eigenvalues of the covariance matrix $\mathbf{\Sigma}_Z$. Each $p \times p$ matrix has p eigenvectors and p eigenvalues corresponding to the eigenvectors. The eigenvectors of the covariance matrix are the axes where the most variance is orientated. We call this eigenvectors principle components. The eigenvalues are the coefficients added to the eigenvectors. Therefore, the eigenvalues describe the amount of variance contributed by each principle component. We rank the eigenvectors by their eigenvalues and will get the principle components in order of their importance (Abdi and Williams, 2010).

The principle component analysis is frequently used for dimension reduction in genetic data (Reich et al., 2008) as well in GWAS data for estimating population stratification (Price et al., 2006). We used the PCA method in Kruppa et al. (2017) and Kruppa and Jung (2017) to reduce the dimension of a multidimensional k-mer space into a three dimensional space and for the detection of outlier by introducing the three dimensional boxplot. Further, we used the eigenvalues of the covariance matrix in Kruppa et al. (2016) to compare the simulated covariance matrix $\mathbf{\Sigma}'$ with the predefined one $\mathbf{\Sigma}$.

1.6 Evaluating in silico simulation and classification methods

In contrast to the before described hypothesis testings of two groups and the decision if the null hypothesis can be rejected, in the case of classification a subject should be assigned to group given. The assignment of the subject can be done in different ways using different approaches. In the simplest case, we have two groups healthy people and ill people; a group of patient has cancer, the other group of patient not. In classification the data set is split into two data sets: the training and testing data set. We train a model or algorithm on a training data set consisting of 2/3 of the patients. Then we validate the model of the training process by the testing data set. Very often, the testing data set consists of the remaining samples of the whole data set. Nevertheless, there is the possibility to use a external validation from data of a different study or to use a temporal validation, with patient data from different time points in a study. In the latter, the patient correlation can cause problems. The results of the testing data set are then summarized in a contingency table shown in table 1.

Table 1 – The contingency table for the summary of a classification algorithm and used for the assessing of the quality. The columns indicating the known truth, therefore the known cancer status of the patients; the rows indicating the estimated or predicted cancer status. In hypothesis testing the false positive (FP) are controlled by the α error and the false negative (FN) by the β error.

		Response / Outcome / Condition				
		Positive/Present	n	Negative/Absent	n	Total
Predictor	Positive/Present	True Positive (TP)	a	False Positive (FP) or α error	c	$a + c$
	Negative/Absent	False Negative (FN) or β error	b	True Negative (TN)	d	$b + d$
			$a + b$		$c + d$	

Table 1 shows the possible outcomes of a classification algorithm. This table can be created on the training as well as on the testing data. We report the results of the testing or validation, because the results of the training data might be biased. This is due the fact, that we use the same subjects for model building and group assignment. Therefore, each patient in the test data set has a cancer status $y_i = \{0, 1\}$ inscribed in the columns. On the left column the sum of patients with present outcome (maligned) and on the right on the sum of the patient with absent outcome (benign). In the rows the predicted status of the patients with a predicted present outcome, below the sum of patients with a absent outcome. The prediction outcome will be at first a probability Pr to have the outcome status i.e. the probability Pr_i that the given patient is maligned. In the next step, the continuous probability outcome is dichotomized by a decision rule: $Pr_i \geq 0.5 \rightarrow 1$ and $Pr_i < 0.5 \rightarrow 0$.

Looking at the table 1 the upper left and the lower right field of the table representing a true classification. The patients in this fields have a positive or negative outcome and the classification algorithm classifies the patients in the correct manner. The lower left are the false negative findings, in hypothesis testing the β error, and in the upper right the false positive findings, also called α error in hypothesis testing. Given this table, we are able to calculate different measures for the goodness of the classification algorithm. Common measures are the sensitivity also called recall in machine learning,

$$Sensitivity = sens = recall = \frac{TP}{TP + FN} \tag{7}$$

the specificity also called selectivity,

$$Sensitivity = spec = selectivity = \frac{TN}{TN + FP} \tag{8}$$

and the corresponding receiver operating curve (ROC) with the area under the curve (AUC). The positive predictive value (ppv) or called precision in machine learning gives the ratio of true classified patients described by

$$Positive\ predictive\ value = ppv = precision = \frac{TP}{TP + FP} \tag{9}$$

Further measures are the accuracy describing the amount of true assigned patients of all patients with

$$Accuary = \frac{TP + TN}{TP + TN + FP + FN}. \tag{10}$$

All described measures are done on the contingency table after dichotomization. It is also possible to asses the probabilities directly and compare these to the outcome by the mean square error (MSE) and the root mean square error (RMSE). The MSE is an estimator for the imperfection of the fit $\hat{f}(x)$ of the training model to the real test data $f(x)$.

$$\mathrm{MSE}\left(\hat{f}(x)\right) = \mathrm{E}\left[\left(\hat{f}(x) - f(x)\right)^2\right] \tag{11}$$

$$\mathrm{RMSE}\left(\hat{f}(x)\right) = \sqrt{\mathrm{MSE}\left(\hat{f}(x)\right)} = \mathrm{E}\left[\left(\hat{f}(x) - f(x)\right)^2\right] \tag{12}$$

In general a lower MSE and RMSE is better. Therefore the model describes the data well and the error is small. As a general problem, the RMSE is sensitive to outliers. After the simulation is run a sensitivity analyis is also possible to check the influence of single samples in a experiment (Kleijnen, 2005). Schrider and Kern (2018) discuss in more detail the shown classification measures in a genetic context of population genetics.

Kruppa et al. (2014a) and Kruppa et al. (2014b) applied the MSE for the comparison of different machine learning algorithms on classification and probability estimation of class membership. Further, the ROC curves are used for the visualization of the sensitivity and specificity. The extended application was shown in Kruppa et al. (2013). In the actual work, Kruppa et al. (2016) and Kruppa et al. (2018b) uses the MSE and RMSE error to evaluate the simulation studies (Section 3.1). Kruppa et al. (2018a) uses different measures for the assessment of the classification quality of virus detection (Section 3.2).

1.7 Variance bias trade off

Like mentioned above the evaluation of the classification algorithm is done on the testing data. Because the training error tends to decrease as the model becomes more complex and therefore more flexible. This is called overfitting. Hence, the training data underestimates the testing error. Looking in more detail, the mean square error consists of two parts, the bias and the variance.

$$\mathrm{MSE}\left(f(x), \hat{f}(x)\right) = \mathrm{bias}^2\left(\hat{f}(x)\right) + \mathrm{var}\left(\hat{f}(x)\right) \tag{13}$$

In an ideal setting a classification algorithm would have a bias $\left(\hat{f}(x)\right)$ of zero combined with a low variance near to zero as well. In a real world setting this is very unlikely and hard to achieve. A low variance means, that the fitted line hits nearly each point in a regression analysis. Hence, the model does overfit the training data. On the other hand, a low bias means, that the model is a straight line to a cubic data model. The model has not enough parameter to fit a good regression line trough the data. Here the bias variance trade off becomes obvious. If a model fits to less parameters, the model does not describe the data. Otherwise, if to much parameters are estimated, the model tends to overfitting the data.

Finally, we can extend the mean square error, which describes the reducible error, by the irreducible error or noise σ^2. The expected prediction error (EPE) describes the reducible error combined with the noise, i.e. irreducible error.

$$\mathrm{EPE}\left(f(x), \hat{f}(x)\right) = \underbrace{\mathrm{bias}^2\left(\hat{f}(x)\right) + \mathrm{var}\left(\hat{f}(x)\right)}_{\text{reducible error}} + \underbrace{\sigma^2}_{\text{noise}} \tag{14}$$

Figure 3 shows the dependency of the model complexity and the different types of errors on the left subplot. The expected prediction error is the sum of the Bias^2 and the variance plus noise. While the Bias^2 decreases with model complexity, the variance will increase. Hence, the best model is a trade off between the Bias^2 and the variance located where the expected prediction error has its minimum. On the right subplot the dependency between the model complexity and the error of the training and test data is demonstrated. While in general the training error will decrease with the increase of the model complexity due to overfitting, the test error will decrease in the beginning and than increase with the model complexity. The trained model models to the training data with too much precision.

In section 3.1, we model the variance and noise on different biological genetic pathways (Kruppa et al., 2016, 2018b). In section 3.2, Kruppa et al. (2017) and Kruppa and Jung (2017) uses dimension reducing methods to decompose the covariance matrix and visualize the variance between the samples. In Kruppa et al. (2018a), we use a decoy database to determine the bias of each sample, because the true infection status of the biological sample is not known. Nevertheless, detection results will be delivered by the algorithms, even with no viral infection.

1.8 Summary

The clinical translation of basic research in genetics is based on data of a high quality. The data must be free of outliers and the statistical properties must be known. Hence, the reproducibility

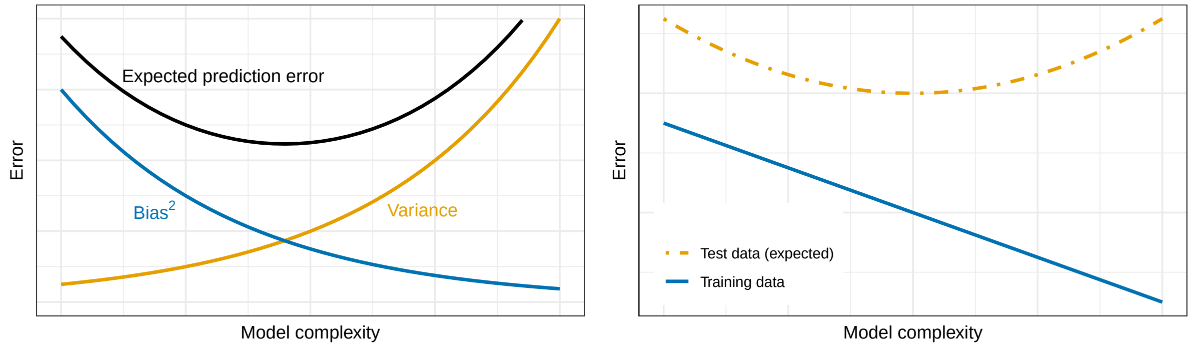

Figure 3 – Graphical representation of the variance bias trade off. On the left side with the increase of the model complexity the bias will decrease while the variance will increase. The expected prediction error is the sum of both errors plus noise. On the right side with the increase of the model complexity, the training error will increase into overfitting. The testing error increases after a given point again. The model is perfectly fitted on the training data. Modified from the supplement of James et al. (2013).

of experiments is connected to many factors. Knowing the data, assumptions for the right statistical methods can be made and the methods can be combined into one bioinformatical pipeline. Each of the used methods must beforehand tested and evaluated on artificial simulation data to understand the limitations of the algorithms. Hence, the generation of multivariate data is needed. Only a multivariate distribution can generate dependent data, which is mostly the case in genetics. On this multivariate data, the dimension reduction methods allow to filter only important variables and therefore reduce beforehand possible false positives. Finally, the visualization of data distributions and outcomes is a crucial step for evaluating bioinformatical results. The number of false positives in an experiment must be controlled, like in statistical testing, or estimated and reported by receiver operating curves or by other classification quality measures.

2 Research question / Aim of the work

Advanced statistical methods like machine learning or the analysis of high-dimensional omics data is often done in a pipeline like fashion. The pipeline is run without any visual inspection of the single steps. Therefore, without any consideration of the assumptions of the inherent statistical methods. Here, I present a collection of methods to achieve more robust statistical outcomes using i) computer simulation of correlated artificial data and ii) the visualization of complex data dependencies.

3 Results

The results section is divided into two parts. First, two publications dealing with in silico simulation of complex correlation structures are presented. We show the generation of multivariate data for Poisson distributed data as well for the binomial distribution. Both distributions are common in genetics, like next generation sequencing and therefore gene expression or the percentage of methylation sites in epigenome wide association studies. Second, three publication showing the visualization of high order dependencies between individuals are described. The dimension reduction of high dimensional viral sequence data and the outlier detection by three dimensional boxplots is presented. Further, both methods are used for the quality assessment for virus detection pipeline in the last publication. In addition, a decoy database is demonstrated for the estimation of false positive. The algorithms are available as R packages.

3.1 In silico simulation of complex correlation structures

In the following section two methods for the generation of multivariate data for the Poisson and binomial distribution is shown. While the multivariate normal distribution is theoretical and practical available Mi et al. (2009), the multivariate distributions of other statistical distributions must be generated by a iterative process. We present a simulation framework for correlated count data and a iterative genetic algorithm for the generation of correlated binary data.

3.1.1 A simulation framework for correlated count data of features subsets in high-throughput sequencing or proteomics experiments

Refers to: Kruppa, J., Kramer, F., Beißbarth, T., and Jung, K. (2016). A simulation framework for correlated count data of features subsets in high-throughput sequencing or proteomics experiments. Statistical applications in genetics and molecular biology, 15(5):401-414
https://doi.org/10.1515/sagmb-2015-0082

In my dissertation I used different machine learning algorithms to estimate the classification probabilities of belong to a given group (Kruppa et al., 2012, 2013, 2014a,b). Machine learning does not have directly assumptions to the data, but like support vector machine the user must decide which kernel to use to run the SVM algorithm. Depending on the machine algorithm, different tuning parameters must be chosen. Further, depending on the these chosen tuning parameters the results can differ in great extend (Kruppa et al., 2014a). I decided, that new developments in machines learning must be trained and evaluated first on data, where the truth is known. Afterward, the new algorithms can be used on real data sets, where the properties of the data can be extracted.

The microarray technology for the detection of different expression levels is based on intensities of different light spots. If a gene is expressed a specific region on the array will emit light. Depending of the array type and technology the emitted light or the calculation of the final signal might differ. Nevertheless, light signals as intensities, as a continuous outcome, is measured. The standard pipeline for analysis such expression data on a continuous scale is done by the `limma` R package, which is the actual state of the art for such data (Ritchie et al., 2015).

The central idea of the `limma` package is not to use the variance connected to the compared genes, but to include the whole variance of the sample in a Bayesian approach (Smyth, 2004). This is called a moderate t test (Yu et al., 2011). Therefore, the t test calculates the variance from the data that is available for each variant, while the moderated t test uses information of all the variants in the data set. Many bioinformatic analysis pipelines are based on the `limma` package and of the concept of using the whole data variance and not only from a small subset. In my work, Kruppa et al. (2016), we asked the question, how to generate correlated data of gene subsets of next generation sequencing (NGS) data? In contrast to micro array data, NGS data has not a continuous outcome, which is normal distributed, but read counts mapping to a given gene of a reference genome. The outcome "read counts" is Poisson distributed with $\mathbf{X} \sim Pois(\lambda)$ or negative binomial with $\mathbf{X} \sim NegBin(\lambda; \gamma)$ with γ as dispersion parameter describing the mean/variance ratio.

If we now want to generate correlated count data, we need a multivariate Poisson or negative binomial distribution. Both multivariate distributions are not available. Therefore, we used the multivariate normal distribution and rounded the generated continuous numbers to discrete numbers without digits. At first it seems that has not a big influence, but if the numbers are small, the effect becomes big. Rounding 1.5 to 2 has an big effect on the numbers. Rounding 54.5 to 55.0 has not a big influence. We compared effect of the rounding by different simulation settings. Further, the main advantage of the multivariate normal distribution is to simulate correlated data on the subject level. Hence, we were able to simulate gene sets, which have the same correlation among each other. This represents a biological pathway, where the assumption of independent genes can not be hold. One main task was to determine a realistic correlation matrix. Our approach was two fold. First, we simulated different correlation structures, which are common: autocorrelation structure of order 1, compound symmetry structure corresponding to a constant correlation, a blocking structure, and an unstructured random correlation (Shown in Figure 1.A of Kruppa et al. (2016)). Second, we estimated the correlation structure from real world data sets and used these correlation structures to generate artificial count data (Shown in Figure 1.A of Kruppa et al. (2016)).

We were able to achieve good performance in the case of the artificial data generation of constructed covariance matrices as well as on the usage of estimated covariance matrices out of real world data. Even if the overall count numbers are small, the covariance matrices differ not from the original one. Hence, we were able to use the multivariate normal distribution using rounding and estimated as well as constructed covariance matrix to generate correlated count data. In the next step, we were able to simulate multivariate distributed binary data in Kruppa et al. (2018b).

3.1.2 A genetic algorithm for simulating correlated binary data from biomedical research

Refers to: Kruppa, J., Lepenies, B., and Jung, K. (2018b). A genetic algorithm for simulating correlated binary data from biomedical research. Computers in biology and medicine, 92:1-8. https://doi.org/10.1016/j.compbiomed.2017.10.023

In the work of Kruppa et al. (2016) we used the rounding of a multivariate normal distribution to generate a multivariate Poisson distribution of count data. We checked if the simulated covariance matrix is nearly the same as the predefined covariance matrix by comparing the eigenvalues and calculating the RMSE as a distance measure. In the following work of Kruppa et al. (2018b) we generated a multivariate binomial data set consisting of binary outcome. In genetics the methylation pattern of given CpG sites show a binomial distribution of presence or absence of a methylation. In Kruppa et al. (2018b) the simulation is based on a genetic algorithm. A genetic algorithm mimics the mutation of the DNA to achieve new sets of numbers. While a Monte Carlo simulation is a pure random search a genetic algorithm is a random searching algorithm including genetic ideas like selection method, cross over, and mutation operators.

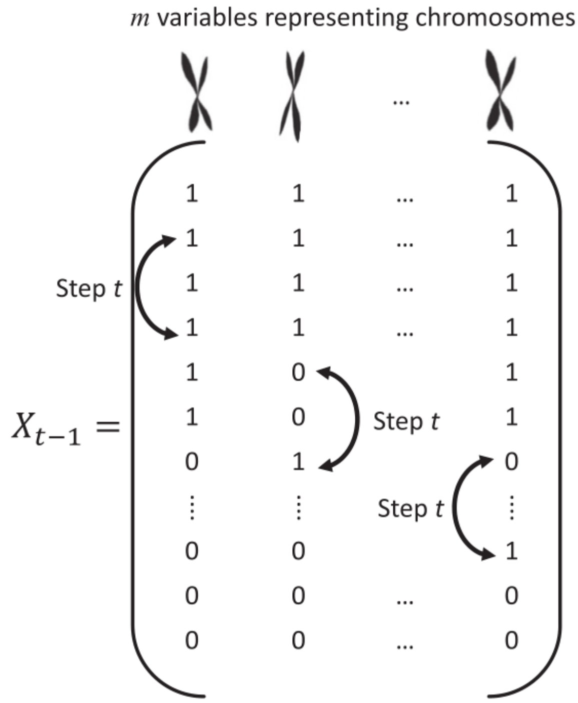

Figure 4 – Reduced example for the genetic algorithm. The columns can be seen as chromosomes representing m variables and the translocation as mutations. In each iteration step t two entries of the data matrix $\boldsymbol{X}$ are randomly swept. Taken from Kruppa, J. *et al.* (2018b). A genetic algorithm for simulating correlated binary data from biomedical research. Computers in biology and medicine, 92:1-8

The algorithm works as follows. First, a start matrix $\boldsymbol{X}_0$ with randomly assigned 0's and 1's is given, a desired covariance or correlation matrix Σ, and a set of stop criteria, when the genetic algorithm should stop the iteration process. As a common stop criteria, we use ϵ as the RMSE error for the difference between the desired $\boldsymbol{\Sigma}$ and the actual one $\boldsymbol{\Sigma}_i$ in iteration i. Further, we set the iterator counter to a maximal value of iterations depending on the computational power of the hardware. In each iteration step i two positions between x_{ij} and x_{ij} are swept. Then the empirical correlation on the margins is determined and checked, if the distance to the desired correlation matrix is smaller. If this is the case, the mutation will be saved and the algorithm goes on with the changed position. If the RMSE is higher, the change of the positions is discarded. This process of sweeping positions is done until the RMSE is below the threshold.

We achieved mean RMSE errors near to zero. With an decrease of the predefined correlation going to zero, more translocation steps are needed for a RMSE of zero. We run the simulation study on low dimensional setting with 1000 patients and three variables. Further, we used the algorithm on the high dimensional setting with up to 100 variables and a lower number of patients. In both cases we were able to achieve a small RMSE error and construct the desired correlation matrix. In general the RSME is smaller, if a larger sample size is given. Also the number of needed translocations to achieve the threshold ϵ is smaller, if the sample size becomes larger. In addition, we compared our approach to three existing ones. All three approaches can only achieve the same results as the genetic algorithm on low dimensional settings. In the case of high dimensional settings, we were only able to test the three competitors on 20 variables. With a higher number of variables the computational efforts can not be handled. Hence, the presented genetic algorithm by Kruppa et al. (2018b) is a flexible tool for the generation of multivariate binary data from a binomial distribution in a high dimensional setting.

Finally, we checked our approach on a real world example for novel carbohydrate ligands of C-type lectin receptors (CLRs). Glycan arrays will give binary data matrices where rows represent glycans of different types and columns represent samples. In brief, we applied our genetic algorithm to simulate glycan array data. Then we checked whether a global test procedures is useful tool to reveal gene set differences between experimental groups. Diagnostic plots allow to check visually the differences between the original and achieved correlation matrix. The deviations should be dispensed over the full matrix and not clustered on one position. Because in both cases the RSME would be the same. All described functionality is available in a corresponding R package.

3.2 Visualization of high order dependencies by dimension reduction

In the following sections we will use parts of the presented methods for data generation for simulation studies to evaluate different methods for the dimension reduction of high dimensional data. First, the kmerPyramid a tool to visualize k-mer distributions of higher order of DNA sequences using the principle component analysis (Kruppa et al., 2017). Further, the gemplots for the detection of outliers or suspicious samples out of the principle components of the variance/covariance matrix (Kruppa and Jung, 2017). Both tools are used for the quality assessment for the final virus detection without a reference genome of the host described in Kruppa et al. (2018a). All three methods are available as R packages with corresponding code and examples.

3.2.1 kmerpyramid: an interactive visualization tool for nucleobase and k-mer frequencies

Refers to: Kruppa, J., van der Vries, E., Jo, W. K., Postel, A., Becher, P., Osterhaus, A., and Jung, K. (2017). kmerpyramid: an interactive visualization tool for nucleobase and k-mer frequencies. Bioinformatics, 33(19):3115-3116.
https://doi.org/10.1093/bioinformatics/btx385

The genetic code consist of only four letters indicating four nucleobases: adenine (A), cytosine (C), guanine (G), and thymine (T), beside uracil (U), which is only present in RNA data. In Kruppa et al. (2017) we focus on the DNA sequence. A string of DNA can be cut into smaller pieces of the length k. This smaller pieces of length k are called k-mers. The smallest possible k-mer is the 1-mer, which consists of $\{A, C, G, T\}$. All possible 2-mer's can be seen as a matrix off all two by two combinations of $\{A, C, G, T\}$. Hence, we can count how many 2-mers of each representation we will find in our DNA sequence. Adding more dimensions to the matrix allows to determine higher k-mers. The 2-mer matrix has the the properties like a variance/covariance matrix seen in formula 2.

$$\text{2-mer} = \begin{array}{c} \\ \text{A} \\ \text{C} \\ \text{G} \\ \text{T} \end{array} \begin{array}{c} \begin{array}{cccc} \text{A} & \text{C} & \text{G} & \text{T} \end{array} \\ \begin{pmatrix} \text{AA} & \text{AC} & \text{AG} & \text{AT} \\ \text{CA} & \text{CC} & \text{CG} & \text{CT} \\ \text{GA} & \text{GC} & \text{GG} & \text{GT} \\ \text{TA} & \text{TC} & \text{TG} & \text{TT} \end{pmatrix} \end{array} \tag{15}$$

We were able to use the principle compounded analysis as dimensional reducing method to achieve a smaller representation of the data if we look at k-mers larger than three. This is important because each species has a special fingerprint of nucleobase distribution. The individuals might differ, but the overall distribution of the nucleobases is the same for a species. Nevertheless, the deviation is so small, that we will not be able to use this information on the level of nucleobases. Therefore, we k-mers of higher order to achieve a separation. Using principle compounded analysis we were able to plot the k-mer distribution of a higher order larger than $k = 3$. By doing this we project the higher order of the k-mers distribution into the 3-dimensional space of the principle components. Interestingly, this dimension reduction will form always a pyramid structure.

The edges of the pyramid will always include the single nuceleobases. Therefore, even a 1-mer dimension reduction looks in the 3D PCA space like a pyramid. Adding more k-mers the pure k-mers consisting only of one nucleotide will be on the tops of the pyramid and the mixtures of them, like $\{AG, ..., CG\}$, on the edges between the pure tops. We added a additional layer by counting the umber of the k-mer appearance indicated by bubbles. We now have a visualization of higher k-mer distribution in DNA sequences. Hence, we are able to compare different species. We can calculate different measures between the pyramids, because each species will be projected into a pyramid. On this distances we can build up phylogenetic trees or other relationship measures. Further we can also look at the single gene level and decide if a gene is maybe transposed from a different species. Like viral genes, which are build in into a host genome or bacteria, which are including resistance genes from other sources. Finally, we can also use the kmerPyramid for the quality control of next generation sequencing reads. We can check if all reads have nearly the same distribution has the source sample or if we get artifacts, like reads consisting only of one nucleotide.

The kmerPyramid was used to help and find more reproducible findings by the viral detection pipeline published in Kruppa et al. (2018a). Therefore, to make the virus detection pipeline more robust. The kmerPyramid is a visual tool to be used, if the virus detection pipeline might have problems or find suspicious outcomes, like virus families of hosts never could interfere with the analysis sample. Finally, we used the kmerPyramid to check if given regions might be transmitted from a different virus or virus family. The kmerPyramid is available as R package.

3.2.2 Automated multigroup outlier identification in molecular high-throughput data using bagplots and gemplots

Refers to: Kruppa, J. and Jung, K. (2017). Automated multigroup outlier identification in molecular high-throughput data using bagplots and gemplots. BMC Bioinformatics, 18(1):232. https://doi.org/10.1186/s12859-017-1645-5

The detection of outliers in biological data is a complex task. First, a biological definition of a outlier must be given. Depending on the biological context, a sample might be a outlier and will be discarded, like in the comparison of patients with a illness to healthy patients. In this case the ill and healthy patients should somehow be the same on the genetic level. A healthy patient with a genetic expression like a ill one might be an outlier. The sample might have the wrong label or the measures might be biased. Hence, the sample will be dropped. On the other hand, we might want to analyse samples detected as outliers, like the detection of novel viral strains or new phylogenetic taxa in a biological sample. In this case we are searching for the outlier to analyze.

In genetics a single sample has thousands of covariate measured like SNPs or expression of genes. All the variables together describe the single biological sample. To compare each sample to the other ones, the dimension must be reduced. Normally, the outlier detection is done after dimension reduction by the principle compounded analysis. The first two components are plotted in a 2-dimensional space. In Kruppa and Jung (2017) we are extend the 2-dimensional plotting by a additional third dimension . Therefore, we present the gemplot as a extension of the bagplot, the 2-dimensional boxplot. Therefore, we are able to visualize a additional dimension of principle components.

In our work we were able to show the advantages of the gemplot on different simulation settings with artificial data. Here, we used the multivariate normal distribution with a autoregressive covariance matrix to model the dependencies between the genes. Therefore, the genes were not independent modeled. In addition, we used a real world data example of kidney tumors and control samples. We were able to show the advantage of the third PCA dimension by detection further outlier, which would be unseen on the first component and the first and second combined.

In the theme of the virus detection we used the gemplot to detect host samples which might be outliers given the measured variables and sequence reads. Therefore, we were able to select interesting samples beforehand and use these samples for the next steps in virus detection procedure. Combined with the information from the kmerPyramid (Kruppa et al., 2017), we were able to achieve more robust detections of viral strains, which could afterward be reproduced in the wet lab. The gmplot is available as R package.

3.2.3 Virus detection in high-throughput sequencing data without a reference genome of the host

Refers to: Kruppa, J., Jo, W. K., van der Vries, E., Ludlow, M., Osterhaus, A., Baumgaertner, W., and Jung, K. (2018a). Virus detection in high-throughput sequencing data without a reference genome of the host. Infection, Genetics and Evolution, 66:180 - 187.
https://doi.org/10.1016/j.meegid.2018.09.026

In my recent work, we faced a special problem in high-throughput sequencing data in virus detection (Kruppa et al., 2018a). So far, the discussed data was high dimensional with much more variables p than samples n. In the case of virus detection by high-throughput sequencing data single biological samples are used. This is the most extreme high dimensional setting with one biological sample $n = 1$. Therefore, we have one single animal infected with a potential pathogen. We used high-throughput sequencing data to reveal the infection of harbor seals with a batai virus (Jo et al., 2018) or the detection of novel canine circovirus strains and bocavirus (Piewbang et al., 2018a,b).

A biological sample, like a dog, died of a pathogen. There are many specialized and very sensitive and specific laboratory tests to determine if a given virus is in the sample. Due to the enormous variety of viral strains not all tests can be conducted limited by time and monetary costs. High-throughput sequencing allows to sequence and detect all the DNA in a given sample. The data produced consist of millions of sequence reads up to 300bp, but very often have lower read length around 100bp. Each sequence read is a fragment of the whole DNA in the sample consisting of the dog DNA, different viral DNA, bacteria DNA and other organisms. In standard pipelines, developed for human tissue, first the reads are mapped to the human reference genome. Then all reads belonging to the human DNA can then removed. If the genome reference of the host is not available or has a lower quality, the host reads can not be removed from the sample. In addition, a single biological sample has not one single virus but whole families of same types or a variety of harmless viruses like the herpesvirus in humans.

Therefore, we had the idea to skip the filtering of host reads. Instead, we mapped the sample sequence reads to all available virus sequences of the NCBI Genbank of approx. 2.4 million DNA sequences. Further we translated the DNA reads to the corresponding amino acid sequences and mapped these reads to approx. 3.3 million virus amino acid sequences of the NCBI Genbank. This overcomes the problem, that host reads will be included into the detection list. On the other hand, we will always detect a virus. It will always map some reads to a sequence of the 2.4 million viral references. Overall, adding a second layer of the amino mapping results in more confidence of the finding. To overcome this problem, we designed a decoy database of viral sequencing reads. The decoy database consists of random shuffled sequence reads of the same size as the original reference genome of the 2.4 million viral reads. While running the detection pipeline, we draw $n_{dr} = 1000$ decoy reads from the decoy database and added these decoy reads to the sample reads.

Table 2 shows the possible outcome of the mapping. A virus read can be mapped to the true virus, the false virus or the decoy sequences. On the other hand, a decoy read can be mapped to the virus or to the decoy reference. Therefore, we can determine the true positive rate for the decoy reads (tpr_d) and the false positive rate (fpr), if a decoy read is mapped to the virus. reference. Using a biological sample, we can not distinguish between the true virus (a, d) and the false virus (b, e). We will find a 2x2 table again. Finally, we can estimate how much a read will be mapped multiple times by determining the deviation of the mapping of the one thousand decoy reads.

We validated the decoy approach by a simulation study with random drawn reads from different virus strains. Further we analyzed two samples of a tinamou and a fin whale of DNA and RNA sequencing data. In both samples the infection was know. The tinamou was infected with a avian hepadnavirus and the dolphin with a morbillivirus. We were able to detect both strains using the virus detection pipeline. Finally, we were able to detect virus stains even if a host reference genome is not available or the coverage of reads is very low. Further, the results can be judged by a decoy database, which allows to assess error rates so that the quality of the final result can be classified.

Table 2 – Contingency table of the possible outcomes by the decoy approach for the viral detection pipeline. In real world examples the true virus can not be distinguished from the false virus. The decoy reads allow to assess the error rates in the detection results.

		Reference			
		True virus	False virus	Decoy	Total
Read	Virus	a	c	e	$a+c+e$
	Decoy	b	d	f	$b+d+f$
		$a+b$	$c+d$	$e+f$	n

In this work we combined the results of the kmerPyramid (Kruppa et al., 2017) and the gemplot (Kruppa and Jung, 2017) to serve as quality control for the final virus detection run of the host sample with unknown host genome. A final overview of the top 25 hits of the pipeline is finally visualized in a complex figure with additional information (Supplementary material, Kruppa et al. (2018a)). This is done after the sequence reads have been classified. The order is determined by DNA and amino reads of the detected viral strains. The pipeline is available as R package.

4 Discussion

4.1 Crisis of reproducibility in science

I started my habiliation with a quote of Tukey and I would like in the end refer to him. Tukey (1980) proposed that we need both exploratory and confirmatory studies. The idea of exploring the data in a circling process is already mentioned in his work. From an idea a question emerged and determines a design. Maybe the possible design will not match the question and therefore a circle of readjusting will begin. How has this idea passed on in the scientific community and how can we balance exploratory and confirmatory studies? This is especially important, if we want to generate new findings and translate these findings into the clinical everyday work. In the following different additional solutions are discussed. With the increase of available data and variables the amount of significant results also increases. Very often the inflation of the type I error is ignored or the influence is not known. Therefore, each bioinformatic pipeline run will produce some significant results. Form here on it becomes more complex, if we want to produce more repeatable results. In the case of clinical translation this is the fundamental layer. If a experimental finding can not repeated, the finding is a false positive finding and therefore meaningless for the patient. Not for the scientific community, a false positive might carry information, but the patient will not profit from a not working medical treatment.

The idea of translation medicine is to transfer the findings from basic research to the patient. Often said "from bench to bedside". If a experiment can not be reproduced the result is not valid and therefore can not be used to cure a sick patient. There is no evidence anymore, that the basic research has found a mechanism in the sickness that can be used for a medicament. Recent developments, like the QUEST criteria emerging of the attributes of robust and innovative research (MERIT), are suggesting a broader spectrum of criteria to asses translation (Kip and Dirnagl, 2019). The central idea of the QUEST criteria are to combine different strategies to map robustness, reproducibility, and confirmation in a research project. The combination of priority setting, which references and literature are available, the strategies for establishing scientific rigor, how to handle bias and are there protocols and guidelines available, the transperancy and dissemination of results, is the study registered or open source strategies, and finally the participation, how will the study stackholder, from patient to funder, be included in the research. A combination of all these principles will help to overcome problems of reproducibility.

Fanelli (2018) stated, that the crisis is not a really crisis but a challenge and is not undermining the scientific enterprise as a whole. Maybe, the expectations on science are unrealistic (França and Monserrat, 2018). If the sample size is small, different results or a failure in replication should not cast doubt on the experiment findings. The sources of different errors must be modeled and controlled correctly. Especially, real world studies have a much higher variability than *in vivo* experiments. Peng (2015) offers a final way out by investing more time in the statistical education. Data collection has to become much to cheap and easy. Therefore, much data is collected without a analysis strategy. Statistical software and standardized data analysis protocols must be combined with clean data to teach people the basics of statistical analysis. Hence, in our work we always published our simulation and analysis code together with simple to complex examples as R packages. All source code is available on GitHub (`https://github.com/jkruppa`). Users and developers can look up the examples and the fundamental R code.

Miotto et al. (2017) describes the problems of data quality in deep learning. Challenges are open beside the great opportunities on the fields of clinical imaging, literature review, electronic health records, and genomics. Data in health care is mostly heterogeneous with data sparsity, redundancy, and missing values. To train valid deep learning algorithm clean and well structured data is demanded. The interpretability of machine learning and deep learning prediction outcomes are often treated as black boxes and must be communicated in a more sophisticated way to convince the medical professional of the usage. Especially in genetics more computational modeling techniques can be applied by deep learning, again the key for using such techniques is a good data quality (Eraslan et al., 2019).

4.2 Exploring data correlation patterns

Wickham et al. (2014) proposed to generate at first tidy data. The data should be imported into a software, here the statistical programming software R. In a next step, a great effort should be made to achieve “tidy data”. Tidy data should mean more than to clean data form missing and wrong labels. The cleaning is also included but tidy data is more. A tidy data set should always look the same. If two tidy data sets $\boldsymbol{X_1}$ and $\boldsymbol{X}_2$ are compared, the single entries might differ, but the overall structure should be the same. Hence, Wickham et al. (2014) puts the programming into a larger theme, shown in figure 5. The data will be imported and after wards “tidied” by a special theme. After the data is tidy, a exploring phase will be performed. The exploring phase is a circle, while different transformation on the data is visualized and then modeled. If the model does not fit to the data, different transformation might be better and feasible. Tong (2019) gives guidance for good data modeling. Finally, the model with the best properties to the data is communicate to the public. Mogil and Macleod (2017) supposed therefore only to publish experiment results with conformation. A journal article should follow the best clinical-research practices to lower false conclusions drawn form animal models and publication bias. Hence, Mogil and Macleod (2017) aim is to strengthen the translation medicine by better exploring the data in more circles.

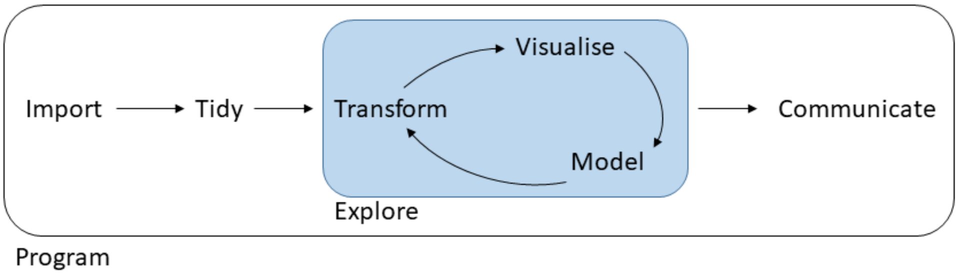

Figure 5 – Programming theme for tidy data. Data will be imported, tidied and explored. After the exploring phase, including transforming, visualizing, and modeling the data, the final results will be communicated. Modified from Wickham and Grolemund (2016).

We used the circle of exploring the data supposed by Wickham et al. (2014) before we communicate the results in Kruppa et al. (2018a). Virus detection is a very complex process. Most importantly, the bioinformatical pipeline will always give a result of a detected virus. Beside the decoy approach discussed in the section 3.2, the focus is now on the communication of the results shown exemplary in figure 6. The host sample was a fin whale. The fin whale died of a possible virus infection. Looking at the most important measure, the number of DNA sequence reads mapping to a reference genome, the MF360246 Frog virus 3 isolate has 134589 sequence reads mapped in contrast to the Aj608288 Dolphin morbillivirus with 39490 sequence reads mapped. From the raw numbers the Frog virus seems to be more “significant” or “relevant” than the Dolphin virus. We would now communicate to the biologist, that the fin whale sample is infect by a frog virus. This would be beside this sample a stunning result, because normally no frog samples are analyzed as a host. Hence, a species boundary has been crossed. We already know from Jo et al. (2017), who was able to detect the dolphin morbillivirus in the wet lab, that the fin whale died of a morbillivirus. The detection of the frog virus would be a false positive finding.

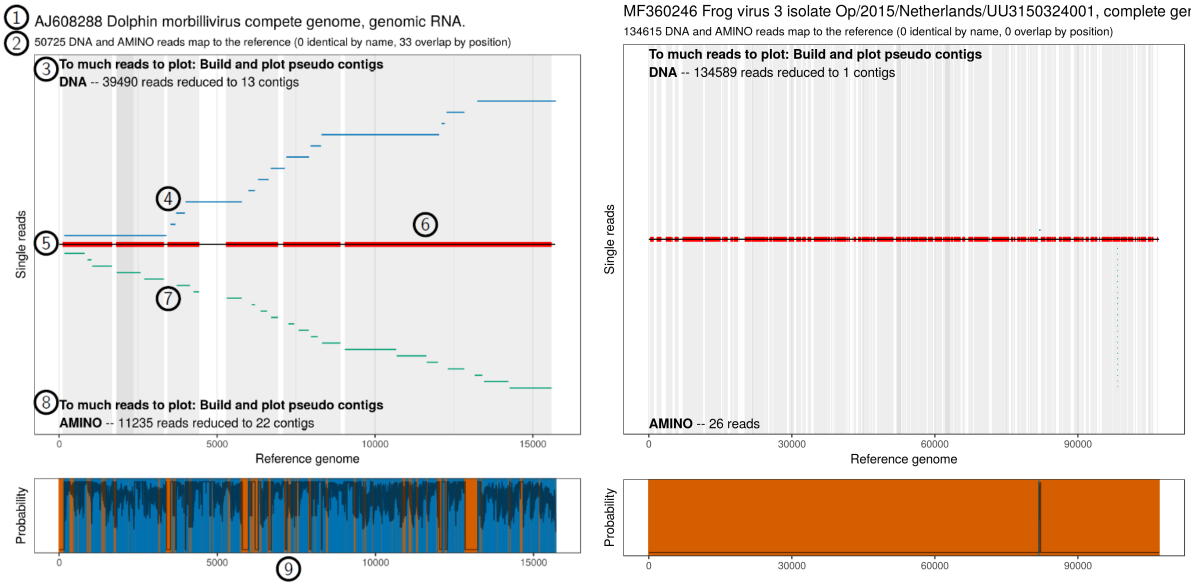

Figure 6 – Visualization of virus detection by next generation sequencing (left) and problematic mapping (right). The name and the NCBI GenBank ID (1), the number of overall mapped DNA and amino acid reads is reported (2), the number of aligned reads by the algorithm (3), the single mapped reads on the reference genome (4), the reference genome (5) with genes indicated by red lines and gray shaded area (6), mapping of the amino acid reads (7), the number of mapped amino acid reads (8) and the coverage of the reference by the DNA reads (9). Taken from Kruppa, J., Jo, W. K., van der Vries, E., Ludlow, M., Osterhaus, A., Baumgaertner, W., and Jung, K. (2018a). Virus detection in high-throughput sequencing data without a reference genome of the host. Infection, Genetics and Evolution, 66:180 - 187.

To achieve more robust findings, we advanced the circle of data exploring by further visualization. Beside the raw number of mapped DNA sequence reads, we plotted the reads by the position of the mapping to the genome. Figure 6 shows that the reads are homogeneous mapped to the dolphin sequence (4), but on one point stacked on the frog genome. As a second layer, the amino mapping gives more confidence. The DNA reads are translated into amico acids and then mapped to the proteome. Again, the amino reads are spread over the full dolphin reference (7). Finally, we also show the coverage of the reads in a second subplot (9). If the genome of the host is covered, a blue color is shown. In addition, a black line indicates the frequency of the most mapped base over alle DNA sequence reads on this position. All information brought together, the evidence is clear, that the fin whale was infected by a Dolphin morbillivirus instead of a Frog virus. The decision would be different on raw mapped read numbers.

In Kruppa et al. (2018b) we also used a visualization to determine the deviation between the outcome correlation matrix $\mathbf{\Sigma}'$ and the predefined one $\mathbf{\Sigma}$. The genetic algorithm is a random walk and we measure the deviation by the RMSE. A small RMSE points out that the two correlation matrices $\mathbf{\Sigma}'$ and $\mathbf{\Sigma}$ are equal in the sum of their deviations. The sum, represented

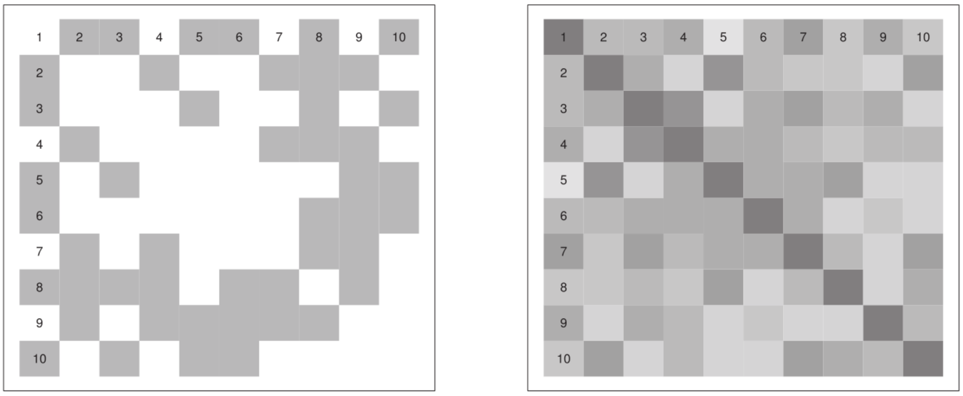

Figure 7 – Diagnostic plot for the evaluation of the outcome correlation matrix to the predefined one. In an ideal setting the deviation is spread homogeneous over the matrix without any cluster. Taken from Kruppa, J. *et al.* (2018b). A genetic algorithm for simulating correlated binary data from biomedical research. Computers in biology and medicine, 92:1-8

by the RMSE, might be misleading, because a region of high deviation and a region of lower deviation would be averaged out. To avoid such biased matrices, we added a diagnostic plot shown in figure 7. The colors should be spread equally to indicate a good correlation matrix. Without these visualization misleading correlation matrices could be generated by the genetic algorithm. Hence, a tested new algorithm on these biased data would get similar biased results. In our previous work of Kruppa et al. (2014a), we showed different probability distributions and the complications of estimating such distributions by machine learning. The visualization of distribution is important to judge the final outcomes, especially if these outcomes are reduced form multi dimensional space to single numbers. The results are supported by Jebb et al. (2017) and Kuznetsova et al. (2018). Jebb et al. (2017) describe exploratory data analysis (EDA) as a tool that helps maximize the value of data. They present different graphical tools to detect data patterns, which help researches to foster reproducible research. Kuznetsova et al. (2018) describes the problems of the visualization of hierarchical biological data. Due to the fact that that most biomedical data is high dimensional, there are no direct visual and interactive tools available. Data visualization is the most crucial step in conveying biomedical results and therefore indispensable. The human understanding is limited to lower dimensions and therefore dimension reduction and well designed visualization tools must be applied to understand. Kuznetsova et al. (2018) focus on hierarchical data structures and shows different designs of illustration.

4.3 Problematic data correlation patterns

It is well known that correlation does not mean causation. To have a high accuracy does not mean to have a good classifier. Only a combination of quality measures can give evidence for a good classifier algorithm. Beside the discussed variance/bias trade off pattern recognition can be problematic on specific data collections. Libbrecht and Noble (2015), Leung et al. (2015) and Minhas et al. (2019) describe different problematic settings in genetic and biomedical data sets. Some of the problematic settings are also discussed in Kruppa et al. (2014b) in context of machine learning. Nevertheless, the concerning properties can be generalized for other methods. Algorithms are mostly trained and tested on ideal data settings or data, which properties fitting to the methods. In the following different types of problematic features are discussed in the context of pattern recognition or classification. Most of them can be faced by good data exploratory tools like visualization or by simulation studies to measure the deviation by a set of common classification quality measures.

The accuracy of a training algorithm might be heavily biased if a imbalanced class size is present (Haixiang et al., 2017). If we assume nine patients in the test data with a benign tumor status and one patient with a maligned status $y_{test} = \{0, 0, 0, 0, 0, 0, 0, 0, 0, 1\}$, hence a overall prevalence of 10% in the training and testing data set. Now we train the algorithm on the data. On the test data we getting only 0's predicted. Nevertheless, we will get an accuracy of 90% correctly predicted patients. Tough, we have not predicted one maligned patient. This scales also for higher sample size with a overall imbalanced class size. Hence, the algorithm has a very high accuracy by predicting only one class overall. A receiver operating curve as visualization tool will reveal the misleading prediction. A high correlation between the variables in a data set will inflict the feature selection and classification. The body weight and the body height are strongly correlated. Hence, an algorithm will have problems to decide, which of both will have a good selectivity to divide the classes of each other. Misleading correlation patterns can also be caused by outliers in the patients groups. Sometimes such correlation can be removed beforehand or can be seen by a principle component analysis. To test how an algorithm can handle complex correlation structures, a simulation of a multivariate data set must be conducted. This comes together with often very heterogeneous data. The different sources of biomedical data are causing a high variance, which must be modeled in a correct way. The best model can not be found in a first step and must be found by tuning and adjusting the hyper parameter of the machine learning algorithms. Finally, machine learning algorithms can not handle missing data. This can become a severe problem, if all missing data points must be removed and the missing values are totally spread at random over the full data set. Hence, a large data set can become a very small one, if all missing values must be removed. Data imputations methods might help, but must be visually inspected. The new artificial data is not a biological representation.

The pattern recognition in biomedical data, especially in genetic data sets is challenging. More data an different omics layers is generated and produced (Figure 8). Chaki and Dey (2019) describes an open task of generation digital genomic data. Beside the expose of private information if genomic data is stored, different data bases exist with different storing properties exists. Chaki and Dey (2019) stated, that the storage and the procedure of analysis must be enhanced to have a better exchange and comparability of the experiments. In addition, genetic data does not exists in one state. Different preprocessing steps, normalization, and dimension reducing methods are multiplying the amount of data to be stored. Finally, a better professional training in the fields of machine learning in practical and theoretical knowledge is demanded from Chaki and Dey (2019).

4.4 Open questions and ongoing research

In our work, we were able to generate multivariate distributions representing gene sets or methylation patterns in different groups of patients. These clusters can be generated with different outcome properties like Poisson, Gaussian or binomial distribution. The next step would be a multi omics approach. The simulation of different gene sets defined by their corresponding KEGG or GO pathways would be the first step. On these data a second layer of methylation differences can then be applied. The effect on the outcome is then not determined by the single gene effect but also by the effects of the methylation pattern and the interaction between both omics layer (Karczewski and Snyder, 2018). Depending on the amount of layers the computational problems can scale very fast. In addition, the complexity of possible research questions will also increase.

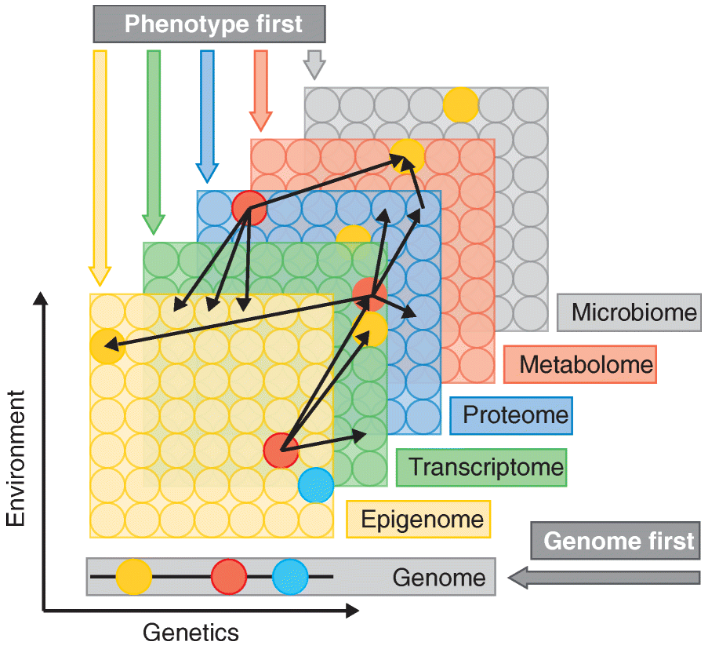

Figure 8 – Different layers of omics including the genome, as baseline layer, the epigenome, the transcriptome, the proteome, the metabolome, and the microbiome. The analysis can be driven by the genome, as first layer, or by the phenotype. Figure 1 of Kruppa et al. (2012) can be seen as a genomic layer represented as GWAS. The arrows indicating possible interactions between the omics layers. Taken from Hasin et al. (2017).

Figure 8 shows the different layers of omics. Very often the genome level will be set as baseline. On the changes of the genome the other omics layers, the epigenome, the transcriptome, the proteome, the metabolome, and the microbiome, will be adapted and modeled. On the other hand, the phenotype can drive the effects as first layer. Depending on the phenotype the different omics layers are then modeled. We plan to simulate the different interactions between the layers to test and develop new methods for the interaction of multiple data omics layers (Rohart et al., 2017). In addition, the source of the biological sample also plays an important role and can have effects of the viability. In the case of methylation, a sample has different CpG sites methylated depending on the biological tissue (Rodger and Chatterjee, 2017). The different concentration of different types of cells in a tissue can alter the methylation state and therefore bias the analysis

(Zheng et al., 2018). For different tissues and organs data bases of cell compositions are available. The oral tissue is not covered. The generation of cell fractions for oral tissue will be one of the next tasks.

Du et al. (2010) describes the outcome variables of a methylation analysis in epigenetics. A methylation analysis is special in the case, that two outcome measures will be produced. From the laboratory machine m-values are generated and reported. These m-values follow a normal distribution with two peaks, one peak describing the distribution of methylated sites and another peak picturing the unmethylated sites. The m-values are than transferred into β-values, which describe the percentage of methylation at a given CpG site. Hence, the m-values are normal distributed and the β-values follow a binomial distribution. Comparing different treatments or the influence of a risk factor on the methylation state, differences in m-values are calculated. Due to the good statistical properties of the normal distributed m-values the analysis is run on the m-values. As a drawback the differences in m-values are not biological interpretable like the difference in percentage of the β-values. The transformation of differences of m-values into differences of β-values is not directly possible. A next research project will deal with this question for reliable and biological interpretable effect measures in epigenetics.

Preprocessing biomedical data is a common step in the bioinformatical data analysis. Each bioinformatic pipeline has at least one quality step or normalization method included. Hence, the analysis will not be conducted on the raw data extracted from a laboratory machine but on the transformed scale. In the metabolome analysis the preprocessing can alter the results of the analysis pipelines in a huge margin. This is mainly caused by the 3D structure of the metabolic data. In our future work, we examine the effects of different preprocessing steps on the annotation of metabolome data. The aim should be to generate a gold standard database with well defined peaks of different properties. The gold standard database can then be used for the assessment of different analysis pipelines and will allow to compare these methods on the same basis.

5 Summary

The clinical translation of basic biomedical research is challenging. In an ideal setting omics experiments would deliver biomarker, which can be then used for the classification of cancer status or other serve diseases. Often this is only the case for very specific genetic markers of genes in a special functional gene, like the p58 gene in human breast cancer. Other findings in mouse experiments or human cell tissue can not always be reproduced in clinical trails. Here the question emerged, does this mean, that there is no effect, hence a false positive finding, or does the biology is such different that a transfer from mice to human of the gained knowledge is not possible. In this work, I focused on the methodical aspects of reproducible research. How can scientist produce reliable and reproducible biological findings? In classical hypothesis testing the type I error or the false positive rate is controlled by different methods like the Bonferroni adjustment or in bioinfomatics the Benjamini Hochberg method. In this case multiple treatment groups are compared and after wards adjusted in that way, that a global type I error of 5% is hold. Classification and prediction methods like machine learning ask a different question. Does a new patient belong to a group given a statistical model. In the center of a classification task stands the model or the algorithm. The model can only be as good as the training data. Understanding the properties of the data is the key to achieve a good classification and prediction algorithm. This comes especially important in a high dimensional data setting, where more variables are modeled than patients are available.

In my work, I introduced different simulation approaches for the generation of complex data with dependent correlation structures, which representing complex biomedical data. Normally, data can be easily generated by drawing samples from a given normal, Poisson, or binomial distribution depending on the outcome of interest. In this simple case, the samples are all independent from each other. The multivariate normal distribution allows to generate dependent samples with a predefined correlation matrix. Hence, we can simulate complex dependencies between genes ordered in pathways or patients doctored in different clinics. Nevertheless, the outcome must be overall normal distributed. In the first section of this work, we extended the possibility of generating multivariate data from a Poisson distribution. A Poisson distribution is demanded, if the outcome consists of count data. Using the multivariate normal distribution and rounding to the next number, we were able to generate multivariate count data. We showed that this simple transformation can hold the predefined correlation matrix, even under very small count numbers, where the rounding has a larger effect. Further, we used the genetic algorithm to generate multivariate binary data following a binomial distribution. The genetic algorithm is a random walk, related to the Monte Carlo simulation. In an iterative process single values of a predefined matrix are swept and the differences to the intended correlation matrix is calculated. Depending on the marginal correlation the iteration process needs more time. We were able to show, that genetic algorithm can generate multivariate distributed binary data in a fast manner. In addition, the algorithm runs especially well on high dimensional data. Combining these to approaches, we were able to generate multivariate data beside the standard normal distributed outcomes. The processes are iterative, but achieve low deviations from the researcher predefined correlation matrices. The methods allow us to find the limitations of methods and problematic data settings, where a algorithm might produce biased results or a unacceptable amount of false positive or false negative findings.

In the second part of my work, I introduced different dimension reducing methods. While the above presented approach does model the correlation and dependencies on the level of the variables, the next approaches focus more at the level of the subjects. The principle component analysis uses the above described correlation between samples to reduce the overall dimension of the data. This is especially important in a high dimensional setting, which is very often the case in genetic data. The dimension reduction of sequencing data allows to visualize high k-mer distributions. A reduction generates always a pyramid independent of the dimension of the k. Hence, it is possible to compare the sequence properties and composition of different organisms or genes. The composition of genes can be compared to each other and possible transferred genes, like antibiotic resistance in bacteria, can be detect. Next, the gemplot allows to extend the 2-dimensional boxplot in the third dimension. Hence, it is possible to consider an additional principle component for the outlier detection by a principle component analysis. We used both methods in combination for the quality control of samples for the virus detection pipeline by next generation sequencing data. The detection in itself is a easy task, because every bioinformatic pipeline will detect a virus in any host sample. This is due the fact, that the sequencing reads are very short and a organism is always infected by viral particles or strains ignoring any integrated viral DNA of the host genome. Hence, I introduced a decoy database to estimate possible false positive and false negative rates. Further, a visualization tool to judge the discovered virus by many additional measures like amino mapping. Further the position and spread of the sequence reads on the detected viral genome.

To concise this work, two additional approximative methods to achieve multivariate binomial and Poisson distributed data is now available. Hence, any binomial and Poisson distributed correlated artificial data can be produced to test the limitations of algorithms. Further, visualization and explanatory methods for high dimensional sequencing data is at disposal. The combination of both achievements allows to attain more robust methods and strengthen the steps to clinical translation.

6 Acknowledgement

Science is to solve complex problems not alone, therefore most achievements cannot be reached by one person. This especially true for such things like scientific publications in peer reviewed journals. First of all, I want to thank all the people who taught and trained me, supported my research, corrected my writings, and discussed my ideas as strange they might be. To make a list of those people, would be very long and cannot be complete. I try nevertheless. I would like to thank, in chronological order, Prof. Ludwig Hothorn, Prof. Andreas Ziegler, Prof. Inke König, Prof. Tim Friede, Prof. Klaus Jung and especially Prof. Geraldine Rauch, who at last encouraged me to write this habilitation. Included are all the great colleagues of all my scientific stations in universities of Hannover, Lübeck, Göttingen, again Hannover and finally Berlin. My work would not be possible without all your input.

Good work cannot be stustained without a balance and support in private life. I am deeply grateful to all people, who supported me outside of university and give me a feeling of normal life in good and in the more difficult times.

References

Abdi, H., Williams, L.J., 2010. Principal component analysis. Wiley interdisciplinary reviews: computational statistics 2, 433–459.

Aulchenko, Y.S., Ripke, S., Isaacs, A., Van Duijn, C.M., 2007. Genabel: an r library for genome-wide association analysis. Bioinformatics 23, 1294–1296.

Burton, A., Altman, D.G., Royston, P., Holder, R.L., 2006. The design of simulation studies in medical statistics. Statistics in medicine 25, 4279–4292.

Chaki, J., Dey, N., 2019. Pattern analysis of genetics and genomics: a survey of the state-of-art. Multimedia Tools and Applications , 1–32.

Cheadle, C., Vawter, M.P., Freed, W.J., Becker, K.G., 2003. Analysis of microarray data using z score transformation. The Journal of molecular diagnostics 5, 73–81.

Díaz-Uriarte, R., De Andres, S.A., 2006. Gene selection and classification of microarray data using random forest. BMC bioinformatics 7, 3.

Du, P., Zhang, X., Huang, C.C., Jafari, N., Kibbe, W.A., Hou, L., Lin, S.M., 2010. Comparison of beta-value and m-value methods for quantifying methylation levels by microarray analysis. BMC bioinformatics 11, 587.

Eraslan, G., Avsec, Ž., Gagneur, J., Theis, F.J., 2019. Deep learning: new computational modelling techniques for genomics. Nature Reviews Genetics , 1.

Fanelli, D., 2018. Opinion: Is science really facing a reproducibility crisis, and do we need it to? Proceedings of the National Academy of Sciences 115, 2628–2631.

França, T.F., Monserrat, J.M., 2018. Reproducibility crisis in science or unrealistic expectations? EMBO reports 19, e46008.

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., Bing, G., 2017. Learning from class-imbalanced data: Review of methods and applications. Expert Systems with Applications 73, 220–239.

Hasin, Y., Seldin, M., Lusis, A., 2017. Multi-omics approaches to disease. Genome biology 18, 83.

Hothorn, T., Bretz, F., Westfall, P., 2008. Simultaneous inference in general parametric models. Biometrical journal 50, 346–363.

Huber, P.J., 2002. John w. tukey's contributions to robust statistics. Annals of statistics , 1640–1648.

Ioannidis, J.P., 2005. Why most published research findings are false. PLoS medicine 2, e124.

James, G., Witten, D., Hastie, T., Tibshirani, R., 2013. An introduction to statistical learning. volume 112. Springer.

Jebb, A.T., Parrigon, S., Woo, S.E., 2017. Exploratory data analysis as a foundation of inductive research. Human Resource Management Review 27, 265–276.

Jo, W.K., Grilo, M.L., Wohlsein, P., Andersen-Ranberg, E.U., Hansen, M.S., Kinze, C.C., Hjulsager, C.K., Olsen, M.T., Lehnert, K., Prenger-Berninghoff, E., et al., 2017. Dolphin morbillivirus in a fin whale (balaenoptera physalus) in denmark, 2016. Journal of wildlife diseases 53, 921–924.

Jo, W.K., Pfankuche, V.M., Lehmbecker, A., Martina, B., Rubio-Garcia, A., Becker, S., Kruppa, J., Jung, K., Klotz, D., Metzger, J., et al., 2018. Association of batai virus infection and encephalitis in harbor seals, germany, 2016. Emerging infectious diseases 24, 1691.

Karczewski, K.J., Snyder, M.P., 2018. Integrative omics for health and disease. Nature Reviews Genetics 19, 299.

Kip, M., Dirnagl, U., 2019. Quest criteria - attributes of robust and innovative research.

Kleijnen, J.P., 2005. An overview of the design and analysis of simulation experiments for sensitivity analysis. European Journal of Operational Research 164, 287–300.

Kleijnen, J.P., 2017. Design and analysis of simulation experiments: Tutorial, in: Advances in Modeling and Simulation. Springer, pp. 135–158.

Köster, J., Rahmann, S., 2012. Snakemake - a scalable bioinformatics workflow engine. Bioinformatics 28, 2520–2522.

Kruppa, J., 2009. Simultaneous confidence intervals for fixed effect parameters in a linear mixed model. Master's thesis. Leibniz Universität Hannover.

Kruppa, J., Jo, W.K., van der Vries, E., Ludlow, M., Osterhaus, A., Baumgaertner, W., Jung, K., 2018a. Virus detection in high-throughput sequencing data without a reference genome of the host. Infection, Genetics and Evolution 66, 180–187. doi:10.1016/j.meegid.2018.09.026.

Kruppa, J., Jung, K., 2017. Automated multigroup outlier identification in molecular high-throughput data using bagplots and gemplots. BMC bioinformatics 18, 232. doi:10.1186/s12859-017-1645-5.

Kruppa, J., Kramer, F., Beißbarth, T., Jung, K., 2016. A simulation framework for correlated count data of features subsets in high-throughput sequencing or proteomics experiments. Statistical applications in genetics and molecular biology 15, 401–414. doi:10.1515/sagmb-2015-0082.

Kruppa, J., Lepenies, B., Jung, K., 2018b. A genetic algorithm for simulating correlated binary data from biomedical research. Computers in biology and medicine 92, 1–8. doi:10.1016/j.compbiomed.2017.10.023.

Kruppa, J., Liu, Y., Biau, G., Kohler, M., König, I.R., Malley, J.D., Ziegler, A., 2014a. Probability estimation with machine learning methods for dichotomous and multicategory outcome: Theory. Biometrical Journal 56, 534–563.

Kruppa, J., Liu, Y., Diener, H.C., Holste, T., Weimar, C., König, I.R., Ziegler, A., 2014b. Probability estimation with machine learning methods for dichotomous and multicategory outcome: Applications. Biometrical Journal 56, 564–583.

Kruppa, J., Schwarz, A., Arminger, G., Ziegler, A., 2013. Consumer credit risk: Individual probability estimates using machine learning. Expert Systems with Applications 40, 5125–5131.

Kruppa, J., van der Vries, E., Jo, W.K., Postel, A., Becher, P., Osterhaus, A., Jung, K., 2017. kmerpyramid: an interactive visualization tool for nucleobase and k-mer frequencies. Bioinformatics 33, 3115–3116. doi:10.1093/bioinformatics/btx385.

Kruppa, J., Ziegler, A., König, I.R., 2012. Risk estimation and risk prediction using machine-learning methods. Human genetics 131, 1639–1654.

Kuznetsova, I., Lugmayr, A., Holzinger, A., 2018. Visualisation methods of hierarchical biological data: A survey and review. International SERIES on Information Systems and Management in Creative eMedia (CreMedia) 2017/2, 32–39.

Lantz, B., 2013. Machine learning with R. Packt Publishing Ltd.

Leipzig, J., 2017. A review of bioinformatic pipeline frameworks. Briefings in bioinformatics 18, 530–536.

Leung, M.K., Delong, A., Alipanahi, B., Frey, B.J., 2015. Machine learning in genomic medicine: a review of computational problems and data sets. Proceedings of the IEEE 104, 176–197.

Libbrecht, M.W., Noble, W.S., 2015. Machine learning applications in genetics and genomics. Nature Reviews Genetics 16, 321.

Maronna, R.A., Martin, R.D., Yohai, V.J., Salibián-Barrera, M., 2018. Robust statistics: theory and methods (with R). Wiley.

Mi, X., Miwa, T., Hothorn, T., 2009. mvtnorm: New numerical algorithm for multivariate normal probabilities. R Journal 1 (2009), Nr. 1 1, 37–39.

Minhas, F., Asif, A., Ben-Hur, A., 2019. Ten ways to fool the masses with machine learning. arXiv preprint arXiv:1901.01686 .

Miotto, R., Wang, F., Wang, S., Jiang, X., Dudley, J.T., 2017. Deep learning for healthcare: review, opportunities and challenges. Briefings in bioinformatics 19, 1236–1246.

Mogil, J.S., Macleod, M.R., 2017. No publication without confirmation. Nature News 542, 409.

Peng, R., 2015. The reproducibility crisis in science: A statistical counterattack. Significance 12, 30–32.

Piewbang, C., Jo, W.K., Puff, C., Ludlow, M., van der Vries, E., Banlunara, W., Rungsipipat, A., Kruppa, J., Jung, K., Techangamsuwan, S., et al., 2018a. Canine bocavirus type 2 infection associated with intestinal lesions. Veterinary pathology 55, 434–441.

Piewbang, C., Jo, W.K., Puff, C., van der Vries, E., Kesdangsakonwut, S., Rungsipipat, A., Kruppa, J., Jung, K., Baumgärtner, W., Techangamsuwan, S., et al., 2018b. Novel canine circovirus strains from thailand: Evidence for genetic recombination. Scientific reports 8, 7524.

Price, A.L., Patterson, N.J., Plenge, R.M., Weinblatt, M.E., Shadick, N.A., Reich, D., 2006. Principal components analysis corrects for stratification in genome-wide association studies. Nature genetics 38, 904.

Purcell, S., Neale, B., Todd-Brown, K., Thomas, L., Ferreira, M.A., Bender, D., Maller, J., Sklar, P., De Bakker, P.I., Daly, M.J., et al., 2007. Plink: a tool set for whole-genome association and population-based linkage analyses. The American journal of human genetics 81, 559–575.

Reich, D., Price, A.L., Patterson, N., 2008. Principal component analysis of genetic data. Nature genetics 40, 491.

Ritchie, M.E., Phipson, B., Wu, D., Hu, Y., Law, C.W., Shi, W., Smyth, G.K., 2015. limma powers differential expression analyses for rna-sequencing and microarray studies. Nucleic acids research 43, e47–e47.

Rodger, E.J., Chatterjee, A., 2017. The epigenomic basis of common diseases. Clinical epigenetics 9, 5.

Rohart, F., Gautier, B., Singh, A., Lê Cao, K.A., 2017. mixomics: An r package for 'omics feature selection and multiple data integration. PLoS computational biology 13, e1005752.

Saeys, Y., Abeel, T., Van de Peer, Y., 2008. Robust feature selection using ensemble feature selection techniques, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer. pp. 313–325.

Sanchez, S.M., 2005. Work smarter, not harder: guidelines for designing simulation experiments, in: Proceedings of the Winter Simulation Conference, 2005., IEEE. pp. 14–pp.

Schrider, D.R., Kern, A.D., 2018. Supervised machine learning for population genetics: a new paradigm. Trends in Genetics 34, 301–312.

Smyth, G.K., 2004. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. Statistical Applications in Genetics and Molecular Biology 3, 1–25. URL: https://doi.org/10.2202/1544-6115.1027, doi:10.2202/1544-6115.1027.

Thissen, D., Steinberg, L., Kuang, D., 2002. Quick and easy implementation of the benjamini-hochberg procedure for controlling the false positive rate in multiple comparisons. Journal of Educational and Behavioral Statistics 27, 77–83.

Tong, C., 2019. Statistical inference enables bad science; statistical thinking enables good science. The American Statistician 73, 246–261.

Tukey, J.W., 1979. Robust techniques for the user, in: Robustness in statistics. Elsevier, pp. 103–106.

Tukey, J.W., 1980. We need both exploratory and confirmatory. The American Statistician 34, 23–25.

Wasserstein, R.L., Lazar, N.A., et al., 2016. The asa's statement on p-values: context, process, and purpose. The American Statistician 70, 129–133.

Wickham, H., Grolemund, G., 2016. R for data science: import, tidy, transform, visualize, and model data. " O'Reilly Media, Inc.".

Wickham, H., et al., 2014. Tidy data. Journal of Statistical Software 59, 1–23.

Woolf, S.H., 2008. The meaning of translational research and why it matters. Jama 299, 211–213.

Yu, L., Gulati, P., Fernandez, S., Pennell, M., Kirschner, L., Jarjoura, D., 2011. Fully moderated t-statistic for small sample size gene expression arrays. Statistical Applications in Genetics and Molecular Biology 10. URL: https://doi.org/10.2202/1544-6115.1701, doi:10.2202/1544-6115.1701.

Zheng, S.C., Breeze, C.E., Beck, S., Teschendorff, A.E., 2018. Identification of differentially methylated cell types in epigenome-wide association studies. Nature methods 15, 1059.

Erklärung

§ 4 Abs. 5 der Habilitationsordnung des Fachbereiches Mathematik und Informatik der Freien Universität Berlin

Hiermit erkläre ich, Dr. rer. hum. biol. Jochen Kruppa, dass

- weder früher noch gleichzeitig ein Habilitationsverfahren durchgeführt oder angemeldet wurde,
- die vorgelegte Habilitationsschrift ohne fremde Hilfe verfasst, die beschriebenen Ergebnisse selbst gewonnen sowie die verwendeten Hilfsmittel, die Zusammenarbeit mit anderen Wissenschaftlern/ Wissenschaftlerinnen und mit technischen Hilfskräften sowie die verwendete Literatur vollständig in der Habilitationsschrift angegeben wurden,
- mir die geltende Habilitationsordnung bekannt ist.

....
Datum

....
Dr. rer. hum. biol. Jochen Kruppa