TU Delft Master of Science Thesis

20 October 2014
Pages 1–17

Identification and Elucidation of Expression Quantitative Trait Loci (eQTL) and their regulating mechanisms using Decodive Deep Learning

Menno J. Witteveen*

Pattern Recognition and Bioinformatics group, Delft University of Technology, Delft The Netherlands

Thesis defense: 27-10-2014

Supervisors: Jeroen de Ridder, Marcel J.T. Reinders

ABSTRACT

Motivation: Identification and elucidation of eQTL has long been an active area of research. Finding *cis*-eQTL has been a manageable problem because of the limited number of candidates. Finding *trans*-eQTL has on the other hand been much more challenging because of the issue of multiple hypothesis testing. It has been suggested that additional information might alleviate this problem and although there has been some success using such methods no comprehensive data integration strategy has been developed.

Approach: In order to comprehensively solve the issue of multiple hypothesis testing in the context of *trans*-eQTL discovery this research introduces MASSQTL: A comprehensive data integration method that makes use of a deep neural network (DNN) to prune the *trans*-eQTL candidate space to a desired size with the objective of finding more significantly associated *trans*-eQTL.

Results: With MASSQTL many more *trans*-eQTL were found using a deep neural network filtering approach. The deep neural network outperformed other machine learning models showing that deep learning by use of complex hierarchical representations is able to model a diverse and sparse set of biological data. In addition to that the method provided new insight into the mechanisms underlying the regulatory architecture of gene expression.

Contact: M.J.WITTEVEEN@student.tudelft.nl

1 INTRODUCTION

The identification of expression quantitative trait loci (eQTL)[1] has long been of great scientific interest because of its potential to unravel the genetic contributions of complex traits (Innocenti *et al.*, 2011). Adding to this is the substantial evidence that is pointing to eQTL as having a crucial role in human evolution by effecting gene-expression from primarily non-coding regions of the genome (Fraser, 2013). The identification of eQTL can generally be subdivided into two key sub-problems. The first sub-problem is the identification of *cis*-eQTL, which are genetic variants that lie close in genomic space from the gene of which the expression is considered. These *cis*-eQTL relations are often easily identified because of the limited number of statistical tests that have to be performed. The limited number of statistical tests results in a less stringent multiple testing correction, leading to increased statistical power. The second sub-problem is the identification of *trans*-eQTL, which are genetic loci that have a large genomic distance from themselves to their respective RNA-expressing gene. The potential solution space for these *trans*-eQTL is substantially larger than for the *cis*-eQTL variants because of the great number of possible variant-expression pairs the can be generated over the Whole Human Genome. Considering this space leads to the need for stringent multiple hypothesis testing correction which greatly reduces the power to identify *trans*-eQTL. The issue of identifying *trans*-eQTL becomes especially pressing if it is viewed in the light of the discovery that despite lower effect-sizes of *trans*-eQTL their cumulative predictive power is much larger than that of *cis*-eQTL (Montgomery and Dermitzakis, 2011), making *trans*-eQTL discovery of central importance in the quest for knowledge about the regulatory landscape of the cell.

1.1 The Challenge

To counter the difficult issue of multiple hypothesis testing, a possible approach is to increase the sample size and combine it with candidate filtering to decrease the size of the solution space (Westra *et al.*, 2013). While this does increase the statistical power, other research contexts might not provide additional samples because they are not available or are difficult to obtain. It should also be noted that even in the case that many samples are available, the exponentially large number of possible variant-expression pairs for a Whole Human Genome remains of a prohibitive size without a candidate filtering step. A filtering step does, however, lead to this issue of selecting a relevant candidate set. This can be done by using a set of known biologically important genetic variants. It also can be done by filtering out variants that do not meet certain statistical criteria like for example being of a sufficient Minor Allele Frequency (MAF). These filtering approaches do, however, remain insufficient because they depend on prior knowledge that unambiguously indicates whether the variant-expression pairs should be included or not. In effect this leads to a chicken-and-egg type of problem: In order to discover whether the biological knowledge is relevant for the current situation it has to be tested for enrichment in *trans*-eQTL, but if one wants to use this information in an unbiased way it has to be available before *trans*-eQTL discovery.

*to whom correspondence should be addressed

[1] An eQTL is defined as a genetic variant (SNP) that has a statistically significant association with the expression of a certain gene.

© All rights reserved 2014.

At present this chicken-and-egg problem is not addressed and therefore it is imaginable that the currently used filtering procedures excluded strong candidates that have the potential to shed light on undiscovered biological mechanisms and functions, making the need for a more comprehensive filtering procedure evident. This becomes especially clear if one considers that different cell types display a large diversity in eQTL associated enrichments, with some enrichments being very cell specific (Fairfax and Knight, 2014)(Montgomery and Dermitzakis, 2011). This is a strong indication that every different eQTL discovery study would benefit from a tailored collection of auxiliary information if it is to be used for filtering. In a recent study an approach is taken that includes the use of a cell specific gene expression network to reduce the trans-eQTL candidate space (Aterido *et al.*, 2014). This approach does make use of a variety of biological information, but still does not alleviate the chicken-and-egg problem as was identified earlier. Adding to the challenge is the evidence that a better performing filtering procedure would likely be comprised of a complex combination of biological information, because of the interplay of genetic variation, DNA binding, chromatin structure and transcription at multiple different scales. (Kilpinen *et al.*, 2013)(Montgomery and Dermitzakis, 2011)(Arneodo *et al.*, 2011). Therefore even if the correct regulatory mechanisms are known this does not solve the issue of filtering in a context with complex multi-scale regulatory interplay. Also highly non-random *trans*-eQTL occurrence rates that can be found as a function of several forms of auxiliary biological information like the presence transcription factors and open chromatin point in the direction of a rich regulatory structure (Battle *et al.*, 2014). Unfortunately, to date non of these insights have been directly applied at the issue of pruning the *trans*-eQTL solution space, because of the unresolved chicken-and-egg problem.

Another related drawback of the current methods to identify *trans*-eQTL is the limited support they provide for integrating other biological information from diverse sources like for example the Gene Ontology (GO) database (Ashburner *et al.*, 2000), which contains various gene and gene-product annotations and the ENCODE repository (Bernstein *et al.*, 2012), which supplies a collection of measurements on functional elements. Providing a flexible integration paradigm for data from such auxiliary information sources would be interesting because data integration remains a key challenge in Computational Biology (de Ridder *et al.*, 2013) and also has the potential to increase the statistical power of the methods used for *trans*-eQTL discovery (Aterido *et al.*, 2014). Combined with the fact that many of these information sources are publicly available, makes algorithms that can perform comprehensive and complex data integration an attractive and potentially very fruitful research direction.

Apart from effective data integration there is the challenge of arriving at a more mechanistic interpretation of functional variation. Although it has been shown experimentally for some *trans*-eQTLs what the mechanisms underlying their regulatory architecture are, for most cases even the generic mechanisms are unknown (Fairfax and Knight, 2014). Similarly as with the previous issue of filtering *trans*-eQTL candidates, standard enrichment procedures do not take complex multi-scale interdependencies into account and thus do not provide deep insight into the regulatory architecture. Therefore it would be very desirable to have a method that performs comprehensive filtering while at the same time providing insight into the global structure of the regulatory architecture.

From the previously discussed methodological problems, we can conclude that to dissolve the challenge at hand a new method is needed that can accomplish the following.

- Radically reduce the *trans*-eQTL candidate space
- Incorporate prior knowledge in the form of auxiliary BioData
- Extract new knowledge about the resulting candidate space

For this a method is needed that can deal with the complex multi-scale interdependencies of these items. An ideal way to achieve this is to take a data-driven approach using deep learning. The rationale for this is that the primary concern of deep learning is the construction of complex data representations by building layered feature hierarchies (Bengio, 2009). The variant of deep learning that is used for this study comes in the form of deep feedforward neural networks, where multiple levels of abstraction are modelled by non-linear hidden layers. Because of this property this deep neural network (DNN) will be able to model the complex multi-scale phenomena that occur in the gene regulatory network using complex hierarchical representations (Leung *et al.*, 2014). For example this deep neural network can be useful for modelling multiple stages of a regulatory network at the sequence level and at higher levels of abstraction like for instance the chromatin structure. In recent times the complex representations created with deep learning methods have proven to be powerful. In the general & non-life-science areas of application, deep learning methods have now surpassed the state-of-the-art performance for many complex convolutional & hierarchical learning tasks (Bengio, 2009). Central to the promise of deep learning is the utilisation of parallel model structure to build efficient GPU-aided implementations that enable it to tackle daunting Big Data problems. This property and the great performance of deep learning on complex learning tasks makes it a tremendously promising technique for the life science. This is especially clear if one considers the challenges associated with the current explosion of biological data (Marx, 2013). However, in contrast to the great promise of deep learning in the life sciences only a few works have made use of its massive potential (Leung *et al.*, 2014)(Di Lena *et al.*, 2012)(Eickholt and Cheng, 2012).

1.2 MASSQTL: The comprehensive Systems Biology methodology

To address the identified challenges a new method, MASSQTL (Figure 1), was developed, which can incorporate diverse sets of auxiliary information by using it to construct a comprehensive deep learning filtering in the form of a deep feedforward neural network (DNN) to prune the solution space to a size small enough to find significant *trans*-eQTL SNP-expression pair candidates and thus call *trans*-eQTL status. The auxiliary data utilized in this study by MASSQTL includes an extensive selection of biological information from different sources including physical Protein Interaction Networks (PIN), gene annotation, evolutionary conservation, local sequence information and different functional elements from the ENCODE project. This vista of auxiliary BioData subsequently enables the deep neural network to model the complex interrelations between all these items by learning multi-scale abstractions. MASSQTL is however not conceptually limited to the current data selection and can incorporate any gene-mappable information into the *trans*-eQTL discovery problem.

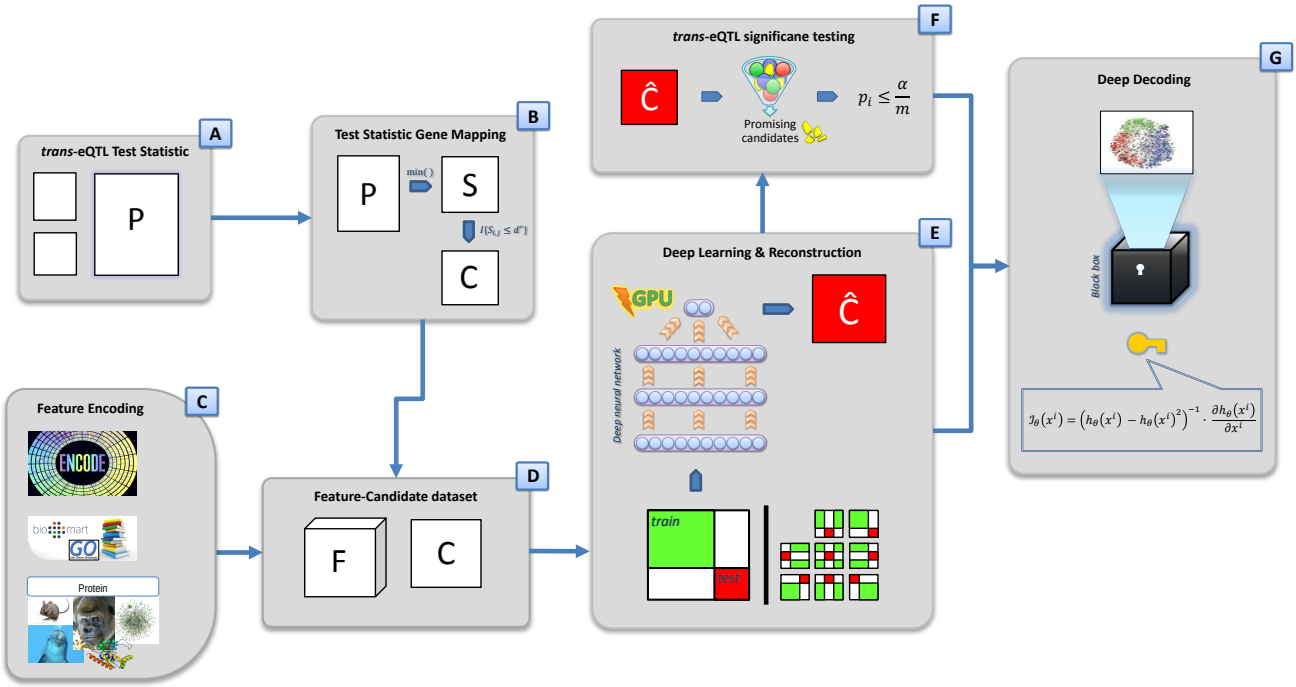

Fig. 1. An overview of MASSQTL: (A) Computing *trans*-eQTL Test Statistics. (B) Mapping the *trans*-eQTL Test Statistics to genes. (C) Encoding features using data from auxiliary information sources. (E) Cross-validation structure & deep learning (F) eQTL Significance Testing. (G) Deep Decoding: a method to do classifier reverse engineering in order to discover what structure is contained in the DNN, the archetypical black-box model.

Another attractive property of MASSQTL is the fact that after the *trans*-eQTL discovery the deep neural network enables one to build a Deep Decoder (DD) that gives insight into the particular biological information that is important for effective filtering of individual candidates. This importance evaluation is used by the DD procedure to discover distinct clusters of *trans*-eQTL that make use of specific yet undiscovered biological mechanisms and functions, giving a new and deeper insight into the global structure of the regulatory architecture of the cell.

The primal data source for this *trans*-eQTL study was data from the GEUVADIS study, which is a combination of RNA-Seq and Whole Genome Wide SNP-Array data from a selection of 337 lymphoblastoid cell-lines extracted from individuals participating in the 1000 Genomes Project; the most important reference dataset of human genetic variation (Lappalainen *et al.*, 2013).

2 APPROACH

MASSQTL consists of seven separate modules, which are graphical depicted in Figure 1. In this section we briefly describe the MASSQTL modules and we elaborate on them in more detail in the following sections. In the first module the statistical associations between genetic variants (SNPs) and gene expressions are computed. This results in a p-value for every SNP-Expression pair culminating in matrix P (Figure 1A). To enable information integration at the gene level the second module maps the SNP-Expression pairs onto their respective genes resulting in a square genomic-genes versus expression-genes score matrix, S. This is done by using a minimum operation on the SNP associated p-values, for a given genomic interval surrounding the gene on which the mapping is performed. Subsequent binarization of S for a given optimized numeric threshold d^* yields class matrix C (Figure 1B). This binarization is done to prepare for the upcoming classification procedure. In the third module every entry in C is combined with a descriptor, which is formulated as a 382 length feature vector that characterizes every gene-gene pair entry, with information from a comprehensive selection of auxiliary BioData from diverse sources (Figure 1C & Section 2.3). Because every gene-gene pair has a vector assigned to it, the collection of these vectors constitutes tensor F (Figure 1D), which has the same width and length dimensions as square matrix C, but has a larger depth of 382. C and F together constitute the completed Feature-Candidate dataset (Figure 1D). Next, this information is used to predict $\widehat{C}$, which is a reconstruction of C. This is done by training 9 DNNs using a customized 9-fold cross validation, that takes special care not to include information from the test set (red) into the training set (green), as this could result in statistical bias (Figure 1E). The reconstruction $\widehat{C}$ is then used to select promising *trans*-eQTL candidates for multiple hypothesis testing by retrieving them from the gene they were mapped on if $\widehat{C}$ identifies them as promising (Figure 1F). Finally in the last module the found *trans*-eQTL relations are investigated using Deep Decoding (DD) which is a procedure to reverse engineer the DNN, which results in deeper insight into the global regulatory architecture of the *trans*-eQTL relations (Figure 1G). The following sections expand on the modules in more detail.

2.1 Computing the *trans*-eQTL Test Statistics

After initial preprocessing the statistical associations between genetic variants (SNPs) and gene expression are computed (Figure 1A). For this computation, which is also commonly referred to as eQTL mapping, an additive linear model with covariates is used.

$$g = \alpha + \gamma x + \beta s + \epsilon \tag{1}$$

In this model output g is the gene-expression and α indicates the offset and γ is used to correct for covariates x. Variable β models the eQTL relation of variant information s and ϵ models the error and is assumed to be normally distributed. Since an additive linear model is assumed the Matrix-eQTL framework (Shabalin, 2012) provides an efficient way of mapping eQTL while correcting for confounding factors like population structure and technical variation. The efficiency of Matrix-eQTL was especially desirable because of the problem size. Matrix-eQTL computes p-values for specified SNP-Expression pairs. In this case this computation was done for every SNP-Expression pair resulting in a SNP-Expression association matrix P that was then used for subsequent processing. The *cis*-eQTL candidates were omitted from further analysis. More technical aspects of the eQTL mapping are discussed in 5.2

2.2 Mapping the *trans*-eQTL Test Statistics to genes

Next, all the p-values in association matrix P are to be mapped onto genes because most auxiliary information is available for gene-level abstraction. Genotype data are, however, complicated to analyse in gene-focused analyses because there is not a standard mapping of SNPs to genes. The reason for this is that multiple SNPs can cover each gene and its regulatory region. To assign SNPs to genes, a genomic interval is defined encompassing each gene and some specified number of bases upstream (5kb) of and downstream (2kb) from the transcribed region. All SNPs within this interval are then used to represent the gene. Given these SNPs, a SNP set association score can be calculated for a gene using a maximum statistic. The maximum statistic is the maximum single-SNP score (or smallest p-value) over all the SNPs assigned to the gene, thus making the best single-SNP score representative for that region (Xiong *et al.*, 2012). This maximum score mapping is done for all SNPs in matrix P and in effect transforms matrix P into matrix S (Figure 1B). This matrix S contains SNP-set association scores, with entry S_{ij} being the maximum association of the SNPs belonging to gene j on the i^{th} gene-expression. The gene on which the SNP mapping was performed is referred to as the genomic-gene.

Next the matrix S is binarized into class matrix C. This is done by assigning the entries in matrix S to the positive class, by assigning a one, if the representative p-values are smaller then threshold d^* and to the negative class, by assigning a zero, otherwise:

$$C_{i,j} = I\{S_{i,j} \leq d^*\} \tag{2}$$

Here, I is symbolizes the indicator function, which return the value one if the expression is true and zero otherwise. For numeric threshold d^* it ought to be stated that it is specifically designed to give the DNN the optimal potential to discover *trans*-eQTL and is computed using the Bonferonni correction (3) and the full p-value distributions (from P) and the SNP-set association scores (S). The computation procedure and rationale for the threshold d^*, are discussed in the materials & methods section (5.3).

2.3 Incorporating Auxiliary Information

An essential part of MASSQTL is the augmentation of a descriptor to every single binarized SNP-set association scores in C. This descriptor is formulated as a feature-vector containing information about the indicated expression-gene, the genomic-gene and their interrelations. This descriptor can useful for *trans*-eQTL discovery, because it enables one to *learn* the filtering with a DNN. So instead of doing *trans*-eQTL candidate filtering with a pre-specified set of auxiliary BioData, a DNN is supplied with descriptor containing a comprehensive set of Auxiliary Information in order to discover the patterns that are relevant for *trans*-eQTL.

The features for the expression-gene and the genomic-gene are both derived from gene specific features. These features include an extensive selection of information. The first selection is information on gene sequence, including GC-content, since it is of great importance in the formation of secondary and tertiary DNA structures (Arneodo *et al.*, 2011). Because the principal gene biotype (e.g. proteine coding or linc RNA) is considered to be an essential property of genes it is also part of the descriptor. A large and comprehensive collection of gene ontology annotations (GO) about for example the location (cytosol, nucleus,etc) or about the type or property of the gene (transcription factor, zinc ion binding, etc.) was deemed appropriate, because this indicates which types of activities a protein can perform. This in turn is useful to determine the biological characteristics (e.g. DNA binding or involved in metabolism). Additional protein information (e.g. complexity, signal domain) was considered since protein domains and structure are often revealing indicators of their function. Also evolutionary conservation across a wide range of species (e.g. Drosophila, Mouse, Dolfin) was included. Specifically a more tailored selection of species with a close evolutionary distance to humans (e.g. Chimpanzee, Gorilla, Gibbon) was added. This being a valuable addition because of the evidence that evolutionary forces and gene regulation are intimately intertwined at these evolutionary time-scales (Romero *et al.*, 2012). Cell type specific information from ENCODE in the form of function element calls like those of transcription factors and histones are included too, noting that the data availability for lymphoblastoid cell lines is particularly comprehensive.

For the descriptor parts specific to the interrelation of gene-gene pairs information, in the form protein interaction network (PIN) measures, were added. Examples include node degree, betweenness centrality and more. In section 5.6 the reader can find an explanation of more granularity on how all these descriptor parts were computed. If we create the descriptor for every gene-gene pair we arrive at tensor, F, as indicated in Figure 1 C&D. This is then combined with the binarized SNP set association scores, C, to complete the Feature-Candidate Dataset (Figure 1D).

2.4 Cross-validation structure & Deep Learning

The Feature-Candidate dataset is then transformed into a regular supervised machine learning problem by dividing the dataset into multiple independent cross validation folds (Figure 1E) and concatenating the respective descriptors in F into a matrix and the respective class labels in C into a vector. In order to make unbiased

predictions one needs to create independent train and test sets. Therefore, in the creation of test and train set, one must exclude samples that contain information that is used to compose both the test set and the train set. This is indeed the reason that for all cross-validation folds (Figure 1E) samples from the same column and row as the test set (red) and train set (green) are not included (white). These white parts are excluded because their presence in test or train set could lead to bias in the method because it would make test and train set dependent, since the parts contain information from both test and train set.

Next the full 9-fold cross-validation was performed by training the respective DNNs with custom and state-of the-art regularisation (section 5.4). This cross-validation resulted in a predicted reconstruction of C being the *trans*-eQTL selection matrix $\widehat{C}$. This matrix contains class-posterior probabilities from the positive class given a certain auxiliary-information-derived descriptor, which is mathematically expressed as $\widehat{C}_{i,j} = P(C_{i,j} = 1|F_{i,j})$.

2.5 Pre-selection of statistical tests and subsequent correction using Bonferroni

The reconstruction $\widehat{C}$ is subsequently used to preselect statistical tests from the solution space P by ranking the class-posterior from the positive class, $\widehat{C}_{i,j}$, in the respective testing blocks (Fig. 1E), in descending order and subsequently selecting top scoring entries and retrieving the *trans*-eQTL candidates that were mapped onto the genomic-gene of the entry. This subsequent selection of top scoring entries and retrieval of candidates is done until a selection of $m/9$ *trans*-eQTL candidates is retrieved and thus m candidates for all folds. Parameter m, which is the number of statistical tests that will be performed, is a value optimized in conjunction with threshold d^* (Section 5.3). The m selected *trans*-eQTL candidates are subsequently tested for their statistical significance using the Bonferroni correction (Goeman and Solari, 2014).

$$p_i \leq \frac{\alpha}{m} \qquad\qquad i \in 1, 2, ..., m \tag{3}$$

In this equation the i stands for the i^{th} statistical tests from the m total number of statistical tests performed. The variable α stands for significance threshold which in this case is taken to be equal to 0.05. The significant candidates were analysed further using Deep Decoding, which is discussed in the next section.

3 INSIGHTS

Now the gained insights will be presented in four sections addressing the performance, validity and the reverse engineering and the interpretation of the results of MASSQTL. First, the performance of MASSQTL is investigated by comparing different variants of the procedure. Following that is an investigation of the potential bias by doing a, so called, crossover validation. Then the DNN is reverse engineered into an interpretable form using Deep Decoding, in order to discover distinct clusters of *trans*-eQTL. Finally, these clusters are analysed from a biological perspective using enrichment, feature analysis and literature embedding.

3.1 Performance Analysis & Discovered *trans*-eQTL

Because actual *trans*-eQTL are rare, the challenge of constructing a *trans*-eQTL candidate filtering using the binarized SNP set association scores is an imbalanced machine learning problem. This results in the receiver operator curve (AUC) being a potentially misleading performance measure since it can be hard to differentiate results even if a model is able to retrieve substantially more positively labelled samples (Davis and Goadrich, 2006), which are very precious in the context of *trans*-eQTL discovery. Therefore two extra performance measures were added, being the precision and the recall (eq. 4). These are then used in conjunction with the AUC to better evaluate model performance. As a final and concluding performance measure, the model is evaluated on the primary objective of the MASSQTL method, being the number of found *trans*-eQTL (TQTL). All the measures were evaluated by averaging the scorings for all the special MASSQTL cross-validation folds, with the important exception of TQTL.

This because of the fact that *trans*-eQTL should be evaluated over the complete set and thus the sum of the discovered *trans*-eQTL for all folds should be used. In order to obtain an estimation of the stability of TQTL a normal distribution was assumed over the per fold number of found *trans*-eQTL. This was used to calculate the standard deviation and this standard deviation was subsequently readjusted for the full set using extrapolation, with probability theory. Next, the precision and recall were computed, by taking their mean across the folds. The reasons for this averaging being sensible will be discussed. To commence this discussion a formalisation of the precision and recall measures is given.

$$\begin{aligned} Precision &= \frac{tp}{tp + fp} \\ &= \frac{|\{\textit{Positive gene-pairs}\} \cap \{\textit{Retrieved gene-pairs}\}|}{|\{\textit{Retrieved gene-pairs}\}|} \\ Recall &= \frac{tp}{tp + fn} = \frac{tp}{p} \\ &= \frac{|\{\textit{Positive gene-pairs}\} \cap \{\textit{Retrieved gene-pairs}\}|}{|\{\textit{Positive gene-pairs}\}|} \end{aligned} \tag{4}$$

The precision and recall are defined in their usual form but also in a second formation, which makes clear that the measure are to be computed for gene-pairs from C, given a to-be-retreived set of gene-pairs, for a specific cross-validation fold. The size of this retrieval set depends strongly on m, but is not equal to m, because m concerns *trans*-eQTL candidates (gene-variant pairs). The number of to-be-retrieved *trans*-eQTL candidates was precisely specified for every fold ($m/9$). The related and smaller number of to-be-retrieved gene-pairs from C per fold was computed from these $m/9$ to-be-retrieved *trans*-eQTL, by counting the unique gene-pairs in this set. This could be done by looking to which genes the variants from this set mapped. With mapping the discussed procedure of section 5.2 is meant. The resulting number of gene-pairs was rather stable. This is because the number of SNPs mapped onto the genes did result in fluctuations in the number of to-be-retrieved gene-pairs, but stabilized for the large m, which was generally in the range of $5 \cdot 10^6$. This made it possible to report averaged precision and recall together with a stability estimate in the form of a standard deviation.

The performance of the DNN was compared with a Regularized Logistic Regression model (RLR), to investigate whether the

deep hierarchical feature representations of the DNN lead to performance increase. This RLR model was trained using stochastic gradient descent using the same iteration scheme as used for the DNN (section 5.4). Momentum was not used for the RLR model. The regularisation that was used for RLR was the L2-norm.

To get an overview of which features provide the most relevant information in the MASSQTL context, three features groups were created. The first group of features included all the BioMart features, which is a group of features that primarily consists out of annotations from the GO repository, thus providing a description of the candidates space in terms ontology. The second feature grouping was constructed using information from the ENCODE repository, including transcription factor and histone presence, thus giving a description of the candidate space with functional and regulatory elements. The third category was a protein specific one, being comprised of PIN network measures and protein conservations across species. All these feature groupings were analysed individually and in concert. DNN Network sizes were adjusted for the size of the input features (section 5.4). Trends in performance will now be discussed.

As can be observed, MASSQTL is able to uncover vastly more *trans*-eQTL, compared to the random case (RAN), in which the predictions were permuted for comparison (Table 1). It can also be observed that the DNN significantly outperforms the RLR model in the central TQTL measure, except in the last case of the protein features. A relevant remark is that, although the DNN does not significantly out-compete the RLR in terms of the other measures, the variance in performance for the different folds does play a crucial roll in the absence of the trend. If the difference between all the measures of DNN are compared against the RLR for the different folds using a paired t-test, all the measures except for the protein features report a significant difference.

Analysing the feature sets for the different models resulted in some interesting observations. As expected, we can observe that for the combined feature-set the performance was best. For the individual sets the ENCODE features clearly outperformed the rest, which is interesting since the relevance and validity of ENCODE has been debated for some time (Graur *et al.*, 2013)(Doolittle, 2013). Since we use this data in an independent dataset it must be concluded that ENCODE elements provide important descriptors for the structure of the regulatory architecture, which MASSQTL attempts to capture. Also it is interesting to note that, for the ENCODE features, the DNN leads to the largest relative increase in performance over the RLR. This can likely be attributed to the fact that a large set of transcription factors is part of these ENCODE features. This set can present an opportunity to the DNN, to model the underlying biology, where multiple functional & regulatory elements, like transcription factors act together to form characteristic distinct complexes, as is for instance observed by Filion *et al.* (2010) in *Drosophila* cells. Likely, the DNN is able to describe the regulatory interplay of these elements by using the information to construct deep representations. To further solidify this, experiments with one hidden layer were performed for the ENCODE data to see if this effected performance (Table 3). The observed decrease in performance showed that a *deep* representation is indeed needed.

For the BioMart feature-set a proper performance was observed too. Ontological descriptions of genes, in the form of GO terms, are likely very informative for pruning the solution space, because they, in a sense, aggregate all the biological knowledge that has been semantically expressed by the scientific community. Also it is quite notable that the RLR model here did not differ significantly in performance from the ENCODE feature-set. This again supports the idea that the structure of the regulatory architecture is best described by complex representations of underlying biological elements, compared to coarse semantical descriptions which do not capture these aspects. This reinforces the earlier identified issue with current methods for *trans*-eQTL discovery, being that there is substantive evidence that simple and coarse filtering will not work optimally in *trans*-eQTL discovery contexts because of rich regulatory interplay, that cannot be captured by simple descriptors.

Lastly, the feature-set containing the protein information performed less well as initially expected. However, because of observations described in following sections, this can be explained. Although some powerful network description measures were used, these measures cannot capture any structure that does not involve proteins and since many *trans*-eQTL have genes involved that are not known to be protein coding, the protein features will be unable to describe these genes. Therefore this feature-set is missing some important information needed for good performance. From this perspective,

Table 1. MASSQTL model performances

(a) ENCODE + BioMart + Protein Features

	AUC	Precision	Recall	TQTL
RAN	50.0 ± 0.4	0.03 ± 0.0	0.66 ± 0.0	215 ± 16
RLR	84.2 ± 0.8	0.85 ± 0.1	16.0 ± 1.0	6353 ± 144
DNN	85.7 ± 0.9	0.95 ± 0.1	17.9 ± 1.2	$\mathbf{7089 \pm 251}$

(b) BioMart Features

	AUC	Precision	Recall	TQTL
RAN	50.0 ± 0.4	0.02 ± 0.0	0.55 ± 0.1	219 ± 12
RLR	73.3 ± 2.6	0.59 ± 0.0	11.1 ± 1.2	4392 ± 115
DNN	73.8 ± 2.0	0.63 ± 0.0	11.8 ± 1.3	$\mathbf{4674 \pm 166}$

(c) ENCODE Features

	AUC	Precision	Recall	TQTL
RAN	49.8 ± 0.4	0.03 ± 0.0	0.55 ± 0.1	220 ± 16
RLR	83.2 ± 0.7	0.57 ± 0.0	10.9 ± 1.1	4328 ± 155
DNN	83.7 ± 0.9	0.68 ± 0.0	12.9 ± 1.6	$\mathbf{5114 \pm 215}$

(d) Protein Features

	AUC	Precision	Recall	TQTL
RAN	50.0 ± 0.5	0.03 ± 0.0	0.6 ± 0.1	239 ± 13
RLR	70.5 ± 1.7	0.23 ± 0.0	4.41 ± 1.5	$\mathbf{1762 \pm 213}$
DNN	70.0 ± 2.3	0.18 ± 0.1	3.33 ± 0.7	1328 ± 114

$\pm$ indicates the standard deviation. Top performances are shown in bold.

this feature-set does seem to perform rather well since 60% of the found *trans*-eQTL involve at least one gene that does not code for a protein. The fact that the RLR seems to perform better is also an unexpected observation, especially since there is evidence that a complex classifier performs better when predicting gene interactions in yeast (Hulsman *et al.*, 2014). Although questions can be raised whether such evidence translates to the case of MASSQTL, it did create the need for additional inquisition. This led to judgement that the DNN configuration was perhaps overcapacity for this case, explaining the difference compared to the case in yeast where the network features were transformed into a more comprehensive & expansive set.

3.2 Validation of MASSQTL

In the previous section we showed that MASSQTL is able to find vastly more *trans*-eQTL, compared to the random case. It could however be argued that the special cross-validation folds to construct $\widehat{C}$ as described in section 2.4 still has the potential to be biased because the train & test set are not completely independent. Imaginably, this bias could occur because the underlying statistical tests have correlation that can be exploited to obtain information about particular samples in the test set, by using the information contained in the training set. To substantiate that this does not occur a validation procedure is needed. This validation procedure (Fig. 2), that consists out of three sub procedures, takes the following shape. First the lymphoblastoid cell line samples are randomly divided into two groups: $H1$ and $H2$. Secondly, for these two groups of cell line samples the complete MASSQTL procedure is performed. These two experiments result into two respective reconstructions, $\widehat{C}$. Thirdly, the reconstruction, or predictions, from both the performed MASSQTL procedures in the form of $\widehat{C}$ are applied on one of the two datasets. This is done with the goal to see the difference in terms of the performance measures, and the overlap of the predictions for the specific *trans*-eQTL candidates. Finally, in order for the MASSQTL procedure to be considered unbiased, all the numbers of the respective performance measure types should not significantly differ from each other. Possible deviations from this number indicate bias has occurred to some extend. Next we will discuss the extend of this bias and whether it brings the validity of MASSQTL into question.

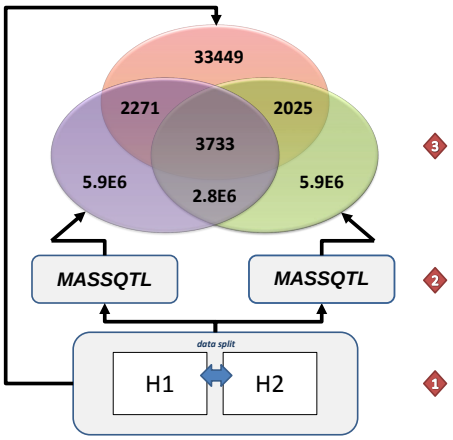

Fig. 2. An overview of crossover validation: **(1)** First the sample of the 1000 Genomes project are divided into two set, randomly. **(2)** Then the first parts of the MASSQTL method is used on both datasets, resulting in two predictions in the form of a $\widehat{C}$. **(3)** Finally, these two predictions are then applied on one of the sets, in order to discover *trans*-eQTL. The overlap between these three set is shown in the venn diagram.

Table 2. Crossover validation

(a) ENCODE + BioMart + Protein Features (self)

	AUC	Precision	Recall	TQTL
RAN	49.8 ± 0.3	0.03 ± 0.0	0.53 ± 0.0	224 ± 20
RLR	84.4 ± 1.4	0.71 ± 0.1	12.8 ± 1.5	5330 ± 269
DNN	85.1 ± 1.3	0.77 ± 0.2	13.9 ± 2.6	$\mathbf{5758 \pm 396}$

(b) ENCODE + BioMart + Protein Features (from crossfold)

	AUC	Precision	Recall	TQTL
RAN	49.9 ± 0.4	0.03 ± 0.0	0.63 ± 0.0	226 ± 14
RLR	84.7 ± 0.6	0.75 ± 0.2	13.1 ± 2.8	5560 ± 455
DNN	85.4 ± 0.6	0.80 ± 0.1	14.2 ± 1.9	$\mathbf{6004 \pm 279}$

$\pm$ indicates the standard deviation. Top performances are in bold.

As can be seen there is a minimal difference between the self prediction and the crossover prediction. This is a very good sign that the bias is not occurring. This is because it indicates that an independent dataset yield a predictor that is just as predictive as using the data directly with MASSQTL.

The further validate, the overlap between the predictors and the true class were investigated of which the result can be found in Figure 2.3. Investigating this makes it clear that not only does the crossover validation experiment result in similar number of found *trans*-eQTL, within the natural variantion of the predictions. It also shows substantial overlap (3733). Considering the enormous number of tests the predictors have to choose from, it is clear that MASSQTL is able to generate very powerfull and generalising predictors.

3.3 Deep Decoding

Using a method similar to (Simonyan *et al.*, 2013) and (Leung *et al.*, 2014) the 9 DNN models from their respective cross-validation folds were evaluated for feature importance, which is the first part of the Deep Decoding algorithm. In the second part of the Deep Decoding algorithm the resulting *trans*-eQTL feature importance vectors were clustered and visualized with t-distributed stochastic neighbour embedding (t-SNE) (Maaten and Hinton, 2008). The found clusters were then re-analysed in terms of the features, which led to some remarkable observations.

3.3.1 The rationale for using Deep Decoding Although there are powerful methods available for discovering structure in biological data by use of interpretable machine learning techniques,

MASSQTL required the development of a new & tailored analysis procedure. This is because of the specific context of the problem. In order to explain the need for a new & tailored analysis procedure, an overview of the available methods is given. This overview is then contrasted with the needs of MASSQTL. After this the new Deep Decoding procedure is introduced.

The majority of all interpretable machine learning methodologies can be classified into two categories. The first and the largest being, methods that yield interpretable results in terms of feature relevance, thereby focusing on the properties of the features. Examples of these methods include e.g. the mining of random forests (Ruiter, 2012) or the interpretation of linear models by investigation of the parameters as estimation of the feature relevance. A second and smaller section of the methods aims to identify meaningful sample clusterings using supervised and unsupervised techniques. This second section of methods therefore focusses on the properties of the samples. Examples possibly include multiple instance learning (MIL) (Amores, 2013) or different types of clustering (Grira *et al.*, 2004).

For the MASSQTL context, a drawback of both these approaches is that neither combines the power of both method categories in a synergistic fashion. Additionally, there is the issue of strong class imbalance, as encountered in this *trans*-eQTL filtering problem. Because of this imbalance, the AUC, as discussed, does not provide an expansive performance measure. However, most procedures that aim to boost interpretability rely on a substantial performance and generally judge that performance in terms of the AUC. This is because interpreting a machine learning model generally relies on all the structure of the model being useful for the stratification. In the case of an imbalanced machine learning problem, this can become a problematic requirement. Unfortunately, the AUC does not always reveal the violation of this requirement, as is the case with the MASSQTL method. As was discussed in the performance evaluation section, the AUC of MASSQTL was quite acceptable. This can deceiving since the precision is generally very low for the specified number of retrievals. This indicates that the DNN makes wrong predictions more often then it makes right ones. This means that the predictor is frequently sampled in areas of the feature space that generally do not yield correct predictions. In contrast to that, there are cases for which the model generalizes very well and these cases might be centred in, so called, hotspots of performance. There is evidence that this is occurring, which can be obtained in the following way. If one investigates figure 2, it can be observed that for the crossover-validation the correct predictions on the positive candidates show relatively more overlap, then the predictions on the positive candidates that are incorrect. This shows that if a prediction is correct it is more likely to be correct in general. Additionally, this means that model structure that was used for these predictions is literally more fitting and thus useful to investigate. Concluding, because of the hotspot phenomena and the fact that the structure of the positive class is of primary interest (*trans*-eQTL), a new method is required that can capture interpretable regularities for models that perform well for only a subspace of subset of cases.

Now, starting with the hypothesis that there are e.g. two distinct types of *trans*-eQTL that exert their effect by different regulatory mechanisms, the question becomes how to discover these types. A first example of these two types of *trans*-eQTL could perhaps be *cis*-eQTL that influence the expression of RNA molecules that interfere with the expression of other genes. A second example could be *cis*-eQTL variants that directly influence the expression of transcription factors that have expression effects downstream. In such an imagined case it is very likely that these two distinct types would have very different features, because information about the properties of this example (e.g. being an interfering RNA or transcription factor) are included in the *trans*-eQTL candidate descriptor. Next we assume, for the sake of example, that both the distinct types of *trans*-eQTL are generated by logistic linear model distributions with different parameter sets θ. Together in one feature space these two linear distributions combined will form a non-linear distribution, because the θ's for the types are not equal. This makes linear models unable to describe this distribution. In this case, a non-linear model would likely be able to map this non-linear distribution by, in effect, forming two linear models. Hence, for a well-fitted model, the points belonging to the respective types lie on a scaled gradient of the model-hypothesis in the features ($\mathcal{I}_\theta(x^i)$) that is directly related to the logistic linear model distribution parameters θ from the distinct type individual points originated from. The scaled gradient of the model-hypothesis is given by equation 5. We will now discuss this equation and then return to the imagined example of distinct *trans*-eQTL types. The complete derivation of this equation is given in section 5.5.

$$\mathcal{I}_\theta(x^i) = \underbrace{\left(h_\theta(x^i) - h_\theta(x^i)^2\right)^{-1}}_{\text{gradient-scaling}} \cdot \underbrace{\frac{\partial h_\theta(x^i)}{\partial x^i}}_{\text{hypothesis-gradient}} \tag{5}$$

Here, the vector $\mathcal{I}_\theta(x^i)$, is the feature importance measure. It is equal in length to x^i. The hypothesis, $h_\theta(x^i)$, is the probability estimate from the model, that sample i, with features x^i belongs to the positive class. The gradient-scaling term, as the name suggests, provides scaling for the gradient of the hypothesis. This is needed to be able to compare different magnitudes class-posterior estimations from the model. In the case of a logistic linear model, the feature importance measure, $\mathcal{I}_\theta(x^i)$, is simply equal to θ for all instances of x. In such a case θ can be used as a measure for feature relevance, aiding model interpretation.

Now we will return to the two imagined *trans*-eQTL types and envision the evaluation of the feature relevance measure $\mathcal{I}_\theta(x^i)$ for all x^i. This $\mathcal{I}_\theta(x^i)$ ought to be equal to the θ from the *trans*-eQTL type from which x^i originated. By performing clustering on $\mathcal{I}_\theta(x^i)$ it will be possible to rediscover the imagined *trans*-eQTL types. Note that, in order to compute a $\mathcal{I}_\theta(x^i)$ that will display these qualities, a machine learning method is needed that is non-linear and has well defined hypotheses and hypothesis gradients.

This is the case for the DNN. By using the back-propropation algorithm it is possible to compute this scaled gradient for the DNN. Because of the previous reasoning, the scaled gradient is prospected to be a local feature relevance measure and therefore will be able to provide characteristic representation for multiple distinct types in the classes. In order to investigate this further a validation experiment for DD was carried out of which section 5.5 provides details. Summarizing, this validation experiments strongly supported the expectation that DD is able to discover the prospected data structure, being the distinct types within classes.

3.3.2 Applying Deep Decoding to the discovered trans-eQTL Since it is now established that DD is able to discovered distinct

Fig. 3. Result of Deep Decoding, visualized with t-SNE. Two clusters of *trans*-eQTL can be clearly identied. Each cluster corresponds to a unique chromatin type: ECC or FHC. **ECC**: This chromatin type is euchromatin related, hence the name euchromatin cluster. **FHC**: The other cluster is a heterochromatin associated cluster, which does show activity, but with a predisposition towards regulatory functions.

sub-types a logical next step is to apply it to the discovered *trans*-eQTL. As discussed this will make the DD procedure focus on the hotspots of performance, thereby specifically mining patterns that are useful in general. The mined patterns were retrieved, per fold, in the form of feature relevance measures $\mathcal{I}_\theta(x^i)$. These feature relevance measures were subsequently visualized using t-SNE (Maaten and Hinton, 2008). The result of this visualisation can be seen in figure 3. The two clearly identifiable clusters were extracted from the t-SNE representation using k-means clustering.

Unfortunately, the feature relevance measures were not stable enough over all folds to be analysed in concert. Possible solutions to this issue, which can be investigated in future research are pointed out in the discussion & conclusion section. Although this prevented an aggregated visualisation and clustering clusters the types of the clusters were easily identified because of their closeness in the global t-SNE visualisation and similar enrichment patterns. Therefore the clusters were re-aggregated into the two distinct cluster types. These aggregates were then analysed in terms of the features, gene enrichment and the findings were embedded into existing literature. The result of this is the topic of the next section, actualizing the insights section.

3.3.3 trans-eQTL are organized into two distinct clusters Like stated, by performing DD is was possible to extract distinct clusters of *trans*-eQTL. Analysis concluded that the distinct clusters are biologically meaningful data representation. The local feature importance measures ($\mathcal{I}_\theta(x^i)$) were computed as discussed for the correctly identified positive candidate gene-pairs for the respective folds and subsequently combined. Analysis of the aggregate made it apparent that there are in fact two distinct types *trans*-eQTL; One being a euchromatin associated type called the euchromatin cluster (ECC) and the other being a developmental facultative heterochromatin associated type called the facultative heterochromatin cluster (FHC). Identification of this clusters was not possible without using $\mathcal{I}_\theta(x^i)$. Experiments using t-SNE visualisation and clustering were performed on x, which did not result in meaningful groupings. To get more insight into the clusters enrichment analysis was performed. The enrichment analysis of the identified clusters was done using DAVID (Sherman *et al.*, 2009).

The ECC type showed enrichment for nucleotide binding and metal-ion binding which are both indicated in active transcription machinery Bannister and Kouzarides (2011). Also CTCF was indicated in this cluster which is an indication that there are genes in the cluster that are involved with keeping the euchromatin structure intact by creation of boundaries between chromatin types(Bannister and Kouzarides, 2011).

The FHC type was enriched for the indicated histone types in figure 3. These histones are typically associated with heterochromatin(Zhou *et al.*, 2011). It was also interesting to see that FHC showed significant enrichment for the nuclear lumen, which is also indicated in facultative heterochromatin function (Zhou *et al.*, 2011). Also a strong enrichment signal was found for actin-binding, which is associated with cellular reprogramming (Miyamoto and Gurdon, 2013).

4 DISCUSSION & CONCLUSION

We showed that, with the help of deep learning, it is possible to integrating a diverse set of auxiliary BioData and find substantially More *trans*-eQTL and elucidate novel regulatory mechanisms.

It can be noticed that choice for gene mapping of the SNPs and not making these SNPs themselves part of the model had a specific justification. If SNP would have been analysed in their own context their specific property would have likely dominated the classification, because information about whether a SNP lies inside a transcription factor binding site would be very valuable information, but would not aid interpretation because it is likely to overshadow the classifier reverse engineering procedure, which is aimed at discovering an overview perspective on the regulatory architecture.

A point of discussion remains the choice to not separate populations when doing the eQTL mapping for this study, which is a more common approach. The separation of populations is regularly done because of issues with confounding factor, which cannot always be corrected for using covariates for the different populations (Hulse and Cai, 2013). Separating the populations does however lead to a smaller sample size and thus decreases statistical power. It should also be reckoned that the focus of this study was to investigate the global structure of the genetic control of gene regulation, by putting it into an evolutionary perspective, by analysing the human populations in concert. This is also the reason why a substantial amount of data was added about gene conservation over short evolutionary distances (2.3). Because the populations were analysed in concert, the overall statistical association, or in other terms the distributions of the p-values, is know to inflate somewhat, because adequate correction cannot be done, solely using covariates (Hulse and Cai, 2013). This inflation was deemed acceptable because of the central goal of finding global structure in gene regulation, in the evolutionary context, considering that, at short evolution time-scales, expression regulation and evolution are inextricably interlinked. Although this analysis led to a positive finding, it can be imagined that a separation of the populations would lead to different insights and perspectives.

Another point of interrogation should be the SNP set association score matrix (S). Because it contains continuous values it provides an option to perform regression and minimize the square error, which would provide an alternative to the outlined binarization and classification. The issue with the regression approach is however that such a procedure in this context has the tendency to concentrate on the bulk of the distribution, which primarily contains eQTL candidates with weak association. Of course weighting and sampling is optional, but there is no clear rational on how to perform this weighting and sampling. Since the Bonferonni correction, as discussed in 5.3 does provide a clear rational on how to proceed, binarization and classification was preferred over regression. Briefly we would like to explain that the choice of using the Bonferonni correction, in contrast to other less stringent corrections, is supported by the need for a *trans*-eQTL set with a minimal number of false positives, since it is to be used for Deep Decoding.

Related to this is the choice to determine binarization threshold d^* by using the outlined method (5.3). An alternative method could also be to optimise it by using Bayesian model parameter optimisation (Snoek *et al.*, 2012). This is however not preferred because of the fact that the primary interest lies in the *global* structure of the regulatory architecture. For example finding an high d threshold using the Bayesian model parameter optimisation might lead to a model that focusses primarily on esoteric *trans*-eQTL that are easy to discover, but are not part of the global mechanisms of expression regulation. Additionally, such an approach would also lead to new issue, being whether to decouple m for this procedure, and how to optimise it. For the DNN itself, there is the related and recurring theme of selecting the hyper-parameters. All these parameters including the regularisation of the RLR were set manually. An alternative, here, could also be the use of Bayesian model parameter optimisation, taking human experts out of the loop. The draw back of such an approach is that a lot more computational resources need to be available to run these optimisations.

The reason that the selection of classifiers was sparse was because of the fact that the algorithms in the selection have the innate property that they scale well with very large datasets. Because the DNN showed superior performance compared to the RLR, a non-linear method will likely be needed to match the performance of the DNN. Random Forest was also considered (Breiman, 2001), but was not included, because it was only feasible to train trees on small subsets of the data because of memory issues. Plus, since one of the central objectives was to gain insight into the regulatory architecture, a method was needed that would lend itself to interpretation. In the case of random forest this is challenging because although there are interpretable methods available (3.3), these are not suitable for the problem because of the class imbalance. DD was able to deal with this, because it can retrieve a local feature relevance scoring and thus is able to find interpretable structure in the data both in terms of the features and the samples (3.3).

As an additional innovation the feature importance measure, as was computed for DD, could be regularized in order to stabilized it. Examples of techniques similar to such an approach include the work on contractive auto-encoders (Rifai and Vincent, 2011) and the related but not functionally similar Fisher kernels as first discussed by Jaakkola *et al.* (1999). In the MASSQTL case, the scaled hypothesis gradient ($\mathcal{I}_\theta(x^i)$) could be added as an additional term to the cross-entropy (eq. 12).

Concluding, we have shown that MASSQTL provides a new lens to perceive regulatory variation and to inform new hypothesis and theories on forefront of biological science. By using deep learning in the form of a DNN, it became apparent that, for this bioinformatics problem, there is a need for deep hierarchical complex transformations in order to recover latent representation that are semantically meaningful. Ergo, in the context of eQTL discovery, the MASSQTL paradigm provides a platform to engage with complex multi dimensional data to obtain deep and meaningful insights. A next step is certainly experimental investigation of the generated hypotheses. Therefore, the found *trans*-eQTL ought to be validated, preferably, on a completely independent dataset, in order to minimize the risk of confounders. Likewise no guarantees can be made about the validity of the found distinct *trans*-eQTL types, although they can be validated in a similar fashion and do provide comprehensive insights and give context to direct experimental investigations.

5 MATERIALS & METHODS

The central aspects of MASSQTL are outlined in the approach section, but for sake of brevity some detailed technicalities were omitted, which will be elaborated here.

5.1 Data preprocessing (GEUVADIS)

5.1.1 RNA-Sequencing Data The Illumina RNA sequencing reads were mapped using the IsoDOT format (Sun *et al.*, 2014). This resulted in transcript counts per Ensembl gene ID, resulting in 46256 mapped transcripts annotated with their respective Ensembl gene ID. Sequencing read counts were normalized using a simple per sample normalisation. The expression counts of every respective gene for every sample were totalled and one was added. This was subsequently divided by the overall total count of the sample. To properly scale the distributions of the expressions a logarithmic scaling was applied. Although this normalisation procedure is relatively simple, it has proven to be effective for a variety of contexts (Dillies and Rau, 2013).

5.1.2 1000 Genomes SNP Data For the purpose of the GEUVADIS study the lymphoblastoid cell lines were from individuals from the 1000 Genomes were re-genotyped for quality control, using SNP arrays (Lappalainen *et al.*, 2013). Although this data showed little difference from the original SNP calls from the original Whole Genome Sequencing result, this SNP array data was used for subsequent analysis. SNPs from this array data with a call rate below 95%, MAF < 0.01, strong evidence against Hardy-Weinberg disequilibrium ($p < 10^{-6}$) were excluded from the dataset, leaving a final set of $1,327,016$ SNPs.

It should be noted that populations were analysed in concert because of the aspects that are of interest to this research, being the discovery of the global structure of regulatory architecture in a context of Human evolution.

5.2 *trans*-eQTL Mapping with covariates

As discussed previously the model for eQTL mapping was an additive linear model implemented using the Matrix-eQTL framework (Shabalin, 2012), which provides an efficient way of mapping eQTL while correcting for confounding factors like population structure and technical variation. Since *cis*-eQTL candidates were to be omitted from further analysis a filter window should be defined. In order to gaurantee that no *cis*-eQTL were included in the analysis a filter window of 5 mb ($5 \cdot 10^6$ bp) surrounding the gene of interest was taken, excluding all genetic variants within this window for further analysis.

As covariates a number of variables were added to the model. The first being the general population (e.g. YRI, CEU or CHB), which was encoded as a 1 if the respective individual was part of this population and a 0 otherwise. Secondly, a correction was performed on the gene-expression, by using multidimensional scaling (MDS), which is a procedure to remove non-specific technical variation from gene-expression data (Westra *et al.*, 2013). For this the first 4 principal components were used. For the genotype data a similar MDS approach was used, to correct for technical variation and possibly remaining population structure. The last covariate that was added was the gender of the individuals, which was also 0-1 coded for the respective genders.

5.3 Determination of binarization threshold d^*

In order to generate an optimal class matrix C to be able to predict its reconstruction $\widehat{C}$ it is necessary to calculate the optimal threshold parameter d^*. The first important realisation is that, because the Bonferonni correction with m tests is used, the *trans*-eQTL candidates fundamentally bifurcate into being either significant or non-significant. Therefore a binarization threshold[2], d, that exactly matches α/m will cater to the needs of the Bonferonni correction. This is the case because the binarization of S with threshold d will make the DNN aim to identify gene-gene pairs of which the best *trans*-eQTL candidate has a statistical association that is equal or better than threshold d. Rephrased, it directs the learning to separate the significant tests from the non-significant ones. Since d is dependent on$m$ the notation $d(m)$ will be used, here.

$$d(m) = \frac{\alpha}{m} \tag{6}$$

In this relation m is the same as used in the Bonferonni correction (eq. 3) and is defined as an integer value. Therefore there is a specific finite set of values $d(m)$ will be able to substantiate. Because the primary objective is to find the maximum number of *trans*-eQTL, the threshold $d(m)$ should be chosen such that the DNN has the optimal potential to identify *trans*-eQTL. Therefore ideal DNNs are *assumed* first, which perfectly reconstruct C. Next this reconstruction, $\widehat{C}$, can be used to select gene-gene pairs that have *trans*-eQTL candidates mapped to them with a p-value lower than threshold $d(m)$. Now having asserted that $I\{S_{i,j} \leq d\} = C_{i,j} = \widehat{C}_{i,j}$, mathematical relations can be determined that can be used to calculate the number of found *trans*-eQTL candidates given a certain m.

The task is therefore to optimize m, which will then result in d^*. A way to convey an intuition about this optimisation is by realising that any MASSQTL procedure with a specific number of tests m and $d(m)$ threshold has a specific performance as measured in found *trans*-eQTL. In this case ideal DNN are *assumed*. Next is the intuition that m needs to balance two trade-off. If for instance m would be taken very small (e.g. 5), the threshold $d(m)$ would be large, leading to a large class of positives. In that case the DNN would be able to deliver candidates that get significant, but in that case only a few (max 5) would be able to reach significance. On the other extreme one might give a very large m (e.g. 10^9), but in that case $d(m)$ would become very small ($5 \cdot 10^{-11}$) and pretty much no candidate would have the association to be considered significant and the situation would not differ very substantially from actually taking all the candidates and testing them all, which would means that no performance gain is possible since no selection is done. Between these two extreme there is an optimum. The following sections will formalize this intuition and turn it into functions that can optimized in a reasonable time. First a simplified example is presented to see how the number of *trans*-eQTL can be calculated given m. Next the simplified example is expanded to the actual case. Finally the optimisation of m is discussed. After this the optimized m is substituted into equation 6 in order to obtain d^*.

[2] If d is optimized, it changes notation to d^*

5.3.1 The simplified example case In order to transfer the coming concepts to the reader, a slightly simplified case is first assumed, being that $S_{i,j} = P_{i,j}$. This means that one and only one SNP-variant maps onto genomic gene (j). By assuming this for the example, the following relations can be used to compute the number discovered *trans*-eQTL ($h_a(m)$), given a specified number of performed tests, m.

$$\begin{aligned} n_P(m) &= \sum_i^{N_g} \sum_j^{N_g} I\{S_{i,j} \leq d(m)\} \\ h_a(m) &= \min\left(n_P(m),\, m\right) \end{aligned} \tag{7}$$

Here $n_P(m)$ indicates the number of *trans*-eQTL candidates, from P (=S), that reach threshold $d(m)$. This is calculated by determining which gene-gene pairs (i and j) in S reach threshold $d(m)$, which is done using indicator function I. In the relation N_g indicates the total number of genes. The double sums in the relation thus sum all possible instances of indicator function I for S. For these $n_P(m)$ *trans*-eQTL candidates, inclusion of any of these into the multiple testing correction of m test using Bonferonni would result in the test being significant. Next it must be noted that, because m tests are performed using the Bonferonni correction, at best, m can become significant, since that are all the test that are to be performed. This means that it is possible that the $n_P(m)$ candidates do not fit into the m to be performed tests. Therefore, since the n_P tests are perfectly recovered because of the *assumed* ideal model ($C_{i,j} = \widehat{C}_{i,j}$), the minimum of m and $n_P(m)$ is the number of discovered *trans*-eQTL ($h_a(m)$), given a certain pre-specified m.

5.3.2 The actual case Now the simplified example is to be expanded to the real case in which $S_{i,j} \neq P_{i,j}$. Still the DNNs are able to perfectly reconstruct C, but now multiple SNP-variants can be mapped onto genomic-gene j. This creates an new challenge since $\widehat{C}$ does not give the relevant information for all these variants, because it is based on genomic-gene top scoring variants. Therefore it is possible that passenger variants that do not have a strong enough association do get into the m eventually performed tests. This phenomena creates the need to break up $n_P(m)$ into a $n_P^+(m)$ and a $n_P^-(m)$. For these variables the relation $n_P(m) = n_P^+(m) + n_P^-(m)$ will hold. The incorporation of this new break up will lead to a new set of equations.

$$\begin{aligned} n_P^+(m) &= \sum_i^{N_g} \sum_j^{N_g} n_{i,j}^+(d(m)) \cdot I\{S_{i,j} \leq d(m)\} \\ n_P(m) &= \sum_i^{N_g} \sum_j^{N_g} n_{i,j} \cdot I\{S_{i,j} \leq d(m)\} \\ \mathbb{E}[h_a(m)] &= \min\left(n_P(m),\, m\right) \cdot \frac{n_P^+(m)}{n_P(m)} \end{aligned} \tag{8}$$

Here $n_P^+(m)$ indicates the number of *trans*-eQTL candidates, from P, that reach threshold $d(m)$. This is calculated by determining which gene-gene pairs (i and j) in S reach threshold $d(m)$, which is done using indicator function I, which is then subsequently multiplied with the number of expression-gene i associated SNPs mapped onto genomic-gene j that reach the threshold $d(m)$, which is denoted by $n_{i,j}^+(d(m))$. Execution of the double sum is performed as previous. It is important to realize that if $I\{S_{i,j} \leq d(m)\} = 1$ then at least one candidate mapped onto gene j will be able to reach threshold $d(m)$ ($n_{i,j}^+(d(m)) \geq 1$), because of the use of the maximum statistic as described in section 2.2.

The subsequent $n_P(m)$ is calculated in a similar way, with the difference being that $n_{i,j}$ is used instead of $n_{i,j}^+(d(m))$. The variable $n_{i,j}$ is the total number of expression-gene i associated SNPs mapped onto genomic-gene j. Thus, for the calculation of $n_{i,j}$, variable m is not needed. In addition to this, it is useful to note that for the first example case $n_P(m)$ was equal to $n_P^+(m)$, being the value one. Hence, separation of $n_P(m)$ into $n_P^+(m)$ and $n_P^-(m)$, was not needed.

Next the expected number of discovered *trans*-eQTL ($\mathbb{E}[h_a(m)]$) can be calculated. The ideal DNNs delivers n_P candidates which can be used for testing. The complexing issue with these candidates is that not all of them will turn out to be significant after inclusion into the m eventually performed tests. This is because of the passenger variants, $n_P^-(m)$. So in the case that m is not large enough to included the complete set n_P, the procedure is forced to randomly sample out of the set, which makes the ratio of $n_P^+(m)$ and n_P times m the expected value of $h_a(m)$. If however the size of m is large enough to encompass all $n_P(m)$ suggested tests, all the $n_P^+(m)$ test will be recovered and their significance will be called. Taking the minimum of $n_P(m)$ and m and multiplying it with the indicated ratio will result in the computation of $\mathbb{E}[h_a(m)]$ for both these two scenarios.

Finally the optimal m should be determined by simply scanning the full range of m-values with the objective to maximize $\mathbb{E}[h_a(m)]$, which yields the estimate of d^* after re-substitution into equation 6.

$$d^* = \alpha \left(\underset{m}{\operatorname{argmax}}\ \mathbb{E}[h_a(m)] \right)^{-1} \tag{9}$$

It must be noted that this rationale does not consider the cases of other, potentially promising, genomic-gene mapped variants that did not substantiate in the SNP-set association score, since this score is taken as representative for the genomic-gene.

Another issue is the risk of introducing statistical bias by determining d^* using the full p-value distribution. Therefore a cross-validation was performed according to the special MASSQTL scheme (2.4). This resulted in 9 estimates of d^*, which were very similar. To make sure the possible bias effect was negligible, the MASSQTL procedure was run using a value of $d(m)$ that was significantly different from these 9 different values, by assuming a normal distribution. The MASSQTL procedure was then rerun using this $d(m)$, to see if the end result, as measured in TQTL, was significantly different, which turned out not to be the case. Therefore it can be concluded that the outlined procedure for calculating d^* is, in this case, not significantly biased and that the procedure is robust to the specific choices of d^*, if it is within the estimated distribution of d^*.

5.4 Objective formulation & training of the deep neural network

As discussed in section 5.3 the learned filtering procedure for the *trans*-eQTL candidates was formulated as a binary classification problem. For the DNN, the nodes of the model were fully connected and parametrized by weights θ. The DNN consists out of multiple non-linear transformations performed by the hidden units of the network. The outgoing activation of each individual unit ν is denoted with a^l_ν with parameter l being the layer it resides in.

$$a^l_\nu = f\left(\sum\nolimits_n^{N^{l-1}} \theta^l_{v,n} a^{l-1}_n\right) \tag{10}$$

The outgoing activation, a^l_ν, is the sum of the weighted output from the previous $l-1$ layer activations. In the equation N^{l-1} indicates the total number of units in the previous layer, noting that a^0 and N^0 are the respective model input and the associated dimensionality. Parameters $\theta^l_{v,n}$ determine the weighting of the incoming activations from the previous layer, a^{l-1}_n. The use of f in equation 10 indicates the activation function used for the hidden units. Empirical explorations with different functions were performed, including sigmoid activation functions, hyperbolic tangent activation functions and rectified linear unit activation units (RELU)(Glorot *et al.*, 2011). The final DNN was constructed using hyperbolic tangent activation functions (TANH). As discussed earlier the inputs into the DNN were of a 382 feature-dimensionality. The Feature-Candidate dataset with tensor F with three matrix-dimensions and C with two were transformed into a regular classification problem by concatenation of the rows and columns into matrix X and vector Y, for the respective train or test set, taking the tailored cross-validation procedure into account (2.4). After completion X contains 382-length feature-vectors x^i with i indicating the i^{th} feature-vector of a total of ω feature-vectors. The associated matrix Y is a vector of length ω containing the class labels from C. Because the MASSQTL learning problem is a binary stratification task of significant *trans*-eQTL candidates versus non-significant ones the choice was made to use softmax in the output layer of the DNN. The output softmax function on the top of the DNN outputs hypothesis h^k_θ, which represents the probability that the sample descriptor x^i belongs to class k.

$$h^k_\theta(x^i) = \frac{\exp\left(\sum_n \theta^{top}_{k,n}\ a^{top}_n(x^i)\right)}{\sum_{k'} \exp\left(\sum_n \theta^{top}_{k,n}\ a^{top}_n(x^i)\right)} \tag{11}$$

In order to train an effective deep learning model it is necessary to provide an objective function that enables the optimization of the parameters θ. For this binary classification problem the cross-entropy is used.

$$\mathcal{L}(\theta) = \sum_{i=0}^{\omega} y^i \log(h_\theta(x^i)) + (1-y^i)\log(1-h_\theta(x^i)) \tag{12}$$

In which y^i is the i^{th} class label of $y \in \{0,1\}$. The function $h_\theta(x^i)$ indicates the hypothesis on the positive class ($k=1$) according to the DNN, with k being omitted from the formulation for sake of brevity.

For training, the weights θ were initialized using a normal distribution with mean zero and an 0.01 standard deviation. In order to make the learning process more effective stochastic gradient descent, with a momentum parameter update rule, was used.

$$\begin{aligned} \theta_t &= \theta_{t-1} + \Delta\theta_t \\ \Delta\theta_t &= \mu_t \Delta\theta_{t-1} - (1-\mu_e)\epsilon_e \nabla_\theta \mathcal{L}(\theta_t, \mathcal{D}^t) \end{aligned} \tag{13}$$

Where,

$$\begin{aligned} \epsilon_e &= \epsilon_0 \gamma^e \\ \mu_e &= \begin{cases} \frac{e}{\Omega}\mu_\Omega + (1-\frac{e}{\Omega})\mu_0 & \text{iff} \quad e < \Omega \\ \mu_\Omega & \text{iff} \quad e \geq \Omega \end{cases} \end{aligned} \tag{14}$$

With subscript t indicating a specific learning step, which is an evaluation of a mini-batch of 500 training samples, $\mathcal{D}^t$,through the network, with ϵ_e being the learning rate for that specific evaluation which was set at 1.0 (ϵ_0) at the start of training and was multiplied with 0.997 (γ) for every new epoch (e). An epoch refers to a full circulation of the training data through the DNN for learning. The parameter μ_t indicates the momentum, which can be viewed as the inertia of the adjustment of the overall learning. This momentum was linearly transformed during learning from 0.5 (μ_0) at the start of learning to 0.99 (μ_Ω) at the end of learning. In order to train the DNN the partial derivative of $\mathcal{L}(\theta)$ with respect to θ was taken in order to perform gradient descent learning using the back-propagation algorithm. The model was trained for 1200 epochs (Ω). Convergence was observed.

Because DNNs are very prone to overfitting state-of-the-art regularisation techniques were utilized. As an initial regularisation a quadratic weight rescaling constraint procedure was used (Srivastava, 2008).

$$\|\theta^l_{\nu,n}\|^2 \leq \xi \qquad n \in 1,2,...N^{l-1} \tag{15}$$

Here the left side of the equation is the squared length of the incoming weight vector for node ν. If this inequality constraint is violated the weight vector is scaled down to a length that does satisfy the constraint. This method does have some functional similarities to L2 regularisation, but is much better suited for the special case of DNN regularisation (Hinton and Srivastava, 2012). The regularisation proved effective, although the applied regularisation constraint was relatively mild ($\xi = 12$). It should also be noted that this constraint is again something that ties into the central directive of this research being the modelling & elucidation of the global structure of the gene regulatory architecture. This is because the quadratic weight constraint directs the DNN to model structure that is more distributed in nature.

As a central regularisation procedure drop-out was used (Hinton and Srivastava, 2012). In drop-out a certain percentage of hidden layer units in the deep neural network (50% in our case) are randomly shut-down during training. By doing this one inhibits the neurons from learning spurious co-adaptations that arises when neurons model patterns that are only useful when combined with other such neurons. The outcome of this is that instead of one single model being trained with K hidden variables, drop-out approximate the training of 2^K different DNNs, on a different subset of the training data. A perspective on this is that drop-out is in effect an extreme form of model averaging, because the random omission of neurons

from the learn process effectively transforms it into DNN with a different architecture, which is also much more efficient than a case in which all the respective model realisations are trained separately. Because of drop-out the neurons in the DNN are forced to learn patterns in their respective representation that are *generally* useful for the network as a whole and thus inhibits the formation of these spurious co-adaptations. An interesting & paradoxical observation is the fact that drop-out makes the individual neurons of the DNN unreliable and thereby makes the DNN globally more reliable.

The DNN was run using a custom implementation written in python by making use of a variety of additional packages. This includes the use of *Theano* which is a GPU math compiler that enables one to access GPU-accelerated computing by providing a tool set to program in the Nvidia CUDA environment (Bergstra and Breuleux, 2010). The developed implementation yielded a speed increase of approximate 60x compared to the native CPU implementation, which was essential for the processing of the large quantities of data in the Feature-Candidate dataset. Because of the large size of the Feature-Candidate dataset (>350 GB) it was impossible to store all the data in memory. Therefore the custom implementation of the DNN included the use of efficient HDF5 storage through the use of a combination of *pytables* and *h5py*, which enabled rapid data decompression of specified parts of the Feature-Candidate dataset from the disk into memory for DNN training. This option adds greatly to the argument that deep learning methods, which can be trained with stochastic gradient descent, provide a powerful solution in Big Data contexts.

Another connected point worth discussion is the way in which the data was divided into the mini-batches over time. Since the data did not fit into memory the data was loaded into memory using batches of $2.5 \cdot 10^6$ samples. This section of data was then used for training for one super-epoch, which is 15 normal epochs, after which a new set of data would be loaded. Because of the class imbalance issue ($P(y = 1) = 3.4 \cdot 10^{-4}$) all the positive samples were loaded into memory and of these 65% randomly selected samples were included for every super-epoch. A different $\frac{1}{10}^{th}$ size set that was sampled in the same way was used as validation set. It should be noted that because of the balancing sampling, the hypothesis, $h_\theta(x)$, is assumed to produce an equi-ranked estimator of $\widehat{C}_{i,j} = P(C_{i,j} = 1|F_{i,j})$, since this ranking is what in practice influences all the performance measures (AUPR,AUC and TQTL). It was important for the model to maintain the class imbalance during learning. Decreasing the negative class sampling probability below 99.9% led to a rapid deterioration of performance. Interestingly, including all the positive samples during training decreased the performance for both DNN and RLR, which is unexpected for most machine learning contexts. It should however be noted that one vital and often central assumption is violated in this machine learning problem, being that the samples of the learning problem are independent and identically distributed random variables (I.I.D). Because in many genomics problems genes show strong correlations, the assumption of the I.I.D is violated and provides an explanation for the observed effect. Remaking that the special cross-validation procedure (2.4) was used, to minimize possible bias that can result from the violation of the I.I.D. assumption. In that respect it was a positive sign that it was observed that the model over-fitted w.r.t the test set, if all the samples were included in the super-epoch, while this over-fitting was not observed in the validation set because it was from the same size-4 training block. This indicates that the tailored cross-validation is effective in minimizing bias. This is because the tailored cross-validation increases the independence between tests and train set. The swapping of the 65% positive samples for a new random set for every super-epoch directs the classifiers to model structure that is general between all the samples in the training set because the classifiers cannot rely upon specific dependency structure between samples in the training set, because these dependency links are destroyed every super-epoch by selecting a new set positive samples. This effect also provides explanation for why the quadratic weight constraint is such an effective regulariser in this learning problem, since it removes modelling structures that are not useful after a new super-epoch is loaded for learning, supporting that this learning scheme is effective for dealing with specific statistical dependencies between the individual samples.

5.5 Derivation & Validation of Deep Decoding

To derive $\mathcal{I}_\theta(x^i)$ (eq.5), a logistic linear model, from (Ng, 2000), was rewritten in the appropriate form and subsequently differentiate with respect to the input features x.

$$\begin{aligned} & h_\theta(x) = \frac{1}{1+\exp(\text{-}\theta^\intercal x)} \\ & \text{-}\log\left(\frac{1}{h_\theta(x)} - 1\right) = \theta^\intercal x \\ & \frac{\partial}{\partial x}\left[\text{-}\log\left(\frac{1}{h_\theta(x)} - 1\right)\right] = \frac{\partial}{\partial x}\left[\theta^\intercal x\right] \\ & \left(h_\theta(x) - h_\theta(x)^2\right)^{\text{-}1} \cdot \frac{\partial h_\theta(x)}{\partial x} = \theta \end{aligned} \tag{16}$$

Therefore the partial derivative together with hypothesis scaling displayed here provides a feature importance measure for a logistic linear model, if the model parameters are interpreted as such. Since the model is linear the parameters θ are invariant to different instances of x. This is different for non-linear models like the DNN. Since the DNN has the same structure as the logistic linear model, in the top, the same rational was applied.

$$\left(h_\theta(x^i) - h_\theta(x^i)^2\right)^{\text{-}1} \cdot \frac{\partial h_\theta(x^i)}{\partial x^i} = \mathcal{I}_\theta(x^i) \tag{17}$$

As can be seen, the formulation for the DNN does not change substantially. This is because the chain-rule. The only important difference is the acquired superscript i for the input features which accentuates the fact that $\mathcal{I}_\theta(x^i)$ should be evaluated for individual feature vectors. Now the evaluations of the formulation will result in local feature importance measures. This evaluation is the central technique of the Deep Decoding method. There is however at this point no theoretical justification for such an evaluation in terms of it being the *true* estimator of a local instance of θ. Hence, empirical justification needed to be devised.

In order to empirically validate Deep Decoding (DD), the method was tested on a synthetic dataset. The class-priors, size of the feature tensor F was taken to be equal in order to perform a validation with maximum relevance for the current context. As ground truth a normal distribution for all the features was taken. In order to see if DD can retrieve relevant substructure in the positive class two subtypes were generated, representing distinct *trans*-eQTL types. The first

distinct type was generated by selecting a random subset of 6 features and summing their values and assigning a top ranked number of samples to the positive class, such that the number of samples of this type equal half of the positive class-prior. The second type was generated in a same way with different selected features. The resulting t-SNE visualisation can be seen in figure 4. From that visualisation it can be seen that DD is able to decode the structure in the data.

5.6 Descriptor creation with auxiliary information

To train a deep learning model every sample in the problem needs to have an associated feature vector of equal length.

A comprehensive set of gene mappable information from 106 key BioMart categories including Gene ontologies, different homology scorings and GC-content was downloaded using ENSEMBLE BioMart. This information was then mapped on the 12368 genes in the dataset. This information was then transformed into a 12368 by 106 matrix, B. The BioMart catagories could be subdivided into three sub-types: nominal data, percentage data and counts. For every sub-type in the mapped BioMart categories the appropriate feature encoding procedure was applied. For nominal data this procedure is to encode for presence of a certain annotation for a certain gene by placing a one on the respective place in B. An inclusion criteria for this sub-type was that their frequency exceeded the 1%. For the percentage category the encoding was a number ranging from 0 to 1 located at the correct location in B. Missing values were few for this sub-type and were imputed using the column mean. For the counts the number was simply placed in the appropriate location in B and was later 0-1 scaled. There were no missing values for this sub-type. A comprehensive list of the included annotations and other information is detailed in the supplementary information (S2).

ENCODE data was also included and mapped onto genes. For this the focus was on the lymphoblastoid cell-lines panels from the ENCODE project, because of their potential relevance for this particular eQTL identification study. Data types included transcription factor binding and histone status. The data was encoded into a 12368 by 82 feature matrix, E. By determining whether the transcription factor or histone complex was present inside a leading 5kb window and 2kb lagging window surrounding the gene it was possible to determine whether the peakcall number should be included in the respective location of E. This value differed from a 100 to a 1000 and was later 0-1 scaled.

To combine the BioMart and ENCODE data matrix G was created by concatenating the gene information matrices from B and E into matrix one large matrix.

Using GeneMANIA a union of several protein interaction networks (PIN) was taken as described in (Battle *et al.*, 2014), yielding a single PPI network. For the resulting network four network features were computed: Degree difference, Degree mean, Hop count and Betweenness. This resulted in an in gene-gene-features tensor, N, of size 12368 by 12368 by 4. This tensor (3d matrix) was used in the following Feature-Candidate Mapping step. Details on how the calculation of the network features were performed is included in (Gleich, 2009).

Next Feature-Canditate Mapping was performed. The aim of this mapping is to assign a equal length feature vector to every sample in the dataset. Samples here are the Summarized eQTL Scorings of respective gene-gene pairs. Labels of these samples are the actual values of the Summerized eQTL Scorings as present in matrix S. To arrive a the final step of integrating the auxiliary information the Feature-Candidate Dataset, which is the combination of matrix S with associated feature vectors for every entry, the matching tensor F for matrix S was created by concatenating the information in G from expression and variant gene and adding to that the information that can be found in N at the respective position, yielding the complete Feature-Candidate Dataset.

Declaration of Interest: The author have no conflicting interest in the context of this research.

REFERENCES

Amores, J. (2013). Multiple instance classification: Review, taxonomy and comparative study. *Artificial Intelligence*, **201**, 81–105.

Arneodo, A., Vaillant, C., Audit, B., Argoul, F., DAubenton-Carafa, Y., and Thermes, C. (2011). Multi-scale coding of genomic information: From DNA sequence to genome structure and function. *Physics Reports*, **498**(2-3), 45–188.

Ashburner, M., Ball, C., Blake, J., and Botstein, D. (2000). Gene Ontology: tool for the unification of biology. *Nature genetics*, **25**(may), 25–29.

Aterido, A., Palacio, C., Marsal, S., Avila, G., and Julià, A. (2014). Novel insights into the regulatory architecture of CD4+ T cells in rheumatoid arthritis. *PloS one*, **9**(6), e100690.

Bannister, A. J. and Kouzarides, T. (2011). Regulation of chromatin by histone modifications. *Cell research*, **21**(3), 381–95.

Battle, A., Mostafavi, S., Zhu, X., Potash, J. B., Weissman, M. M., McCormick, C., Haudenschild, C. D., Beckman, K. B., Shi, J., Mei, R., Urban, A. E., Montgomery, S. B., Levinson, D. F., and Koller, D. (2014). Characterizing the genetic basis of transcriptome diversity through RNA-sequencing of 922 individuals. *Genome research*, **24**(1), 14–24.

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and trends in Machine Learning*, **2**(1), 1–127.

Bergstra, J. and Breuleux, O. (2010). Theano: a CPU and GPU math compiler in Python. *Proc. 9th Python conference*, **1**(2), 1–7.

Bernstein, B. E., Birney, E., Dunham, I., Green, E. D., Gunter, C., and Snyder, M. (2012). An integrated encyclopedia of DNA elements in the human genome. *Nature*, **489**(7414), 57–74.

Breiman, L. (2001). Random forests. *Machine learning*, pages 5–32.

Davis, J. and Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. *Proceedings of the 23rd international conference on Machine learning*, pages 233–240.

de Ridder, D., de Ridder, J., and Reinders, M. J. T. (2013). Pattern recognition in bioinformatics. *Briefings in bioinformatics*, **14**(5), 633–47.

Di Lena, P., Nagata, K., and Baldi, P. (2012). Deep architectures for protein contact map prediction. *Bioinformatics (Oxford, England)*, **28**(19), 2449–57.

Dillies, M.-A. and Rau, A. (2013). A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Briefings in Bioinformatics*, **14**(6), 671–683.

Doolittle, W. F. (2013). Is junk DNA bunk? A critique of ENCODE. *Proceedings of the National Academy of Sciences of the United States of America*, **110**(14), 5294–300.

Eickholt, J. and Cheng, J. (2012). Predicting protein residue-residue contacts using deep networks and boosting. *Bioinformatics (Oxford, England)*, **28**(23), 3066–72.

Fairfax, B. P. and Knight, J. C. (2014). Genetics of gene expression in immunity to infection. *Current opinion in immunology*, **30C**, 63–71.

Filion, G. J., van Bemmel, J. G., Braunschweig, U., Talhout, W., Kind, J., Ward, L. D., Brugman, W., de Castro, I. J., Kerkhoven, R. M., Bussemaker, H. J., and van Steensel, B. (2010). Systematic protein location mapping reveals five principal chromatin types in Drosophila cells. *Cell*, **143**(2), 212–24.

Fraser, H. (2013). Gene expression drives local adaptation in humans. *Genome research*, **23**(7), 1089–1096.

Gleich, D. F. (2009). *Models and Algorithms for PageRank Sensitivity*. Ph.D. thesis, Stanford University.

Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep sparse rectifier networks. *roceedings of the 14th International Conference on Artificial Intelligence and Statistics*, **15**, 315–323.

Goeman, J. and Solari, A. (2014). Multiple hypothesis testing in genomics. *Statistics in medicine*, **33**(September 2012), 1946–1978.

Graur, D., Zheng, Y., Price, N., Azevedo, R. B. R., Zufall, R. a., and Elhaik, E. (2013). On the immortality of television sets: "function" in the human genome according to the evolution-free gospel of ENCODE. *Genome biology and evolution*, **5**(3), 578–90.

Grira, N., Crucianu, M., and Boujemaa, N. (2004). Unsupervised and semi-supervised clustering: a brief survey. *A review of machine learning . . .* , pages 1–12.

Hinton, G. and Srivastava, N. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv: . . .* , pages 1–18.

Hulse, A. M. and Cai, J. J. (2013). Genetic variants contribute to gene expression variability in humans. *Genetics*, **193**(1), 95–108.

Hulsman, M., Dimitrakopoulos, C., and de Ridder, J. (2014). Scale-space measures for graph topology link protein network architecture to function. *Bioinformatics (Oxford, England)*, **30**(12), i237–i245.

Innocenti, F., Cooper, G. M., Stanaway, I. B., Gamazon, E. R., Smith, J. D., Mirkov, S., Ramirez, J., Liu, W., Lin, Y. S., Moloney, C., Aldred, S. F., Trinklein, N. D., Schuetz, E., Nickerson, D. a., Thummel, K. E., Rieder, M. J., Rettie, A. E., Ratain, M. J., Cox, N. J., and Brown, C. D. (2011). Identification, replication, and functional fine-mapping of expression quantitative trait loci in primary human liver tissue. *PLoS genetics*, **7**(5), e1002078.

Jaakkola, T., Haussler, D., and Others (1999). Exploiting generative models in discriminative classifiers. *Advances in neural information processing systems*, pages 487–493.

Kilpinen, H., Waszak, S. M., Gschwind, A. R., Raghav, S. K., Witwicki, R. M., Orioli, A., Migliavacca, E., Wiederkehr, M., Gutierrez-Arcelus, M., Panousis, N. I., Yurovsky, A., Lappalainen, T., Romano-Palumbo, L., Planchon, A., Bielser, D., Bryois, J., Padioleau, I., Udin, G., Thurnheer, S., Hacker, D., Core, L. J., Lis, J. T., Hernandez, N., Reymond, A., Deplancke, B., and Dermitzakis, E. T. (2013). Coordinated effects of sequence variation on DNA binding, chromatin structure, and transcription. *Science*, **342**(6159), 744–7.

Lappalainen, T., Sammeth, M., Friedländer, M. R., 't Hoen, P. a. C., Monlong, J., Rivas, M. a., Gonzàlez-Porta, M., Kurbatova, N., Griebel, T., Ferreira, P. G., Barann, M., Wieland, T., Greger, L., van Iterson, M., Almlöf, J., Ribeca, P., Pulyakhina, I., Esser, D., Giger, T., Tikhonov, A., Sultan, M., Bertier, G., MacArthur, D. G., Lek, M., Lizano, E., Buermans, H. P. J., Padioleau, I., Schwarzmayr, T., Karlberg, O., Ongen, H., Kilpinen, H., Beltran, S., Gut, M., Kahlem, K., Amstislavskiy, V., Stegle, O., Pirinen, M., Montgomery, S. B., Donnelly, P., McCarthy, M. I., Flicek, P., Strom, T. M., Lehrach, H., Schreiber, S., Sudbrak, R., Carracedo, A., Antonarakis, S. E., Häsler, R., Syvänen, A.-C., van Ommen, G.-J., Brazma, A., Meitinger, T., Rosenstiel, P., Guigó, R., Gut, I. G., Estivill, X., Dermitzakis, E. T., Palotie, A., Deleuze, J. F., Gyllensten, U., Brunner, H., Veltman, J., Cambon-Thomsen, A., Mangion, J., Bentley, D., and Hamosh, A. (2013). Transcriptome and genome sequencing uncovers functional variation in humans. *Nature*, **501**(7468), 506–11.

Leung, M. K. K., Xiong, H. Y., Lee, L. J., and Frey, B. J. (2014). Deep learning of the tissue-regulated splicing code. *Bioinformatics (Oxford, England)*, **30**(12), i121–i129.

Maaten, L. V. D. and Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning*, **9**, 2579–2605.

Marx, V. (2013). Biology: The big challenges of big data. *Nature*, **498**(7453), 255–260.

Miyamoto, K. and Gurdon, J. B. (2013). Transcriptional regulation and nuclear reprogramming: roles of nuclear actin and actin-binding proteins. *Cellular and molecular life sciences : CMLS*, **70**(18), 3289–302.

Montgomery, S. B. and Dermitzakis, E. T. (2011). From expression QTLs to personalized transcriptomics. *Nature reviews. Genetics*, **12**(4), 277–82.

Ng, A. (2000). CS229 - Lecture notes - Lecture 1. *CS229 Lecture notes*, pages 1–30.

Rifai, S. and Vincent, P. (2011). Contractive auto-encoders: Explicit invariance during feature extraction. *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, (1).

Romero, I. G., Ruvinsky, I., and Gilad, Y. (2012). Comparative studies of gene expression and the evolution of gene regulation. *Nature reviews. Genetics*, **13**(7), 505–16.

Ruiter, J. R. D. (2012). *Mining the forest: interpretable classification models in biology with application to scale-space signals*. Ph.D. thesis.

Shabalin, A. a. (2012). Matrix eQTL: ultra fast eQTL analysis via large matrix operations. *Bioinformatics (Oxford, England)*, **28**(10), 1353–8.

Sherman, B. T., Huang, D. W., and Lempicki, R. A. (2009). Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nature protocols*, **4**(1), 44–57.

Simonyan, K., Vedaldi, A., and Zisserman, A. (2013). Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv preprint arXiv:1312.6034*, pages 1–8.

Snoek, J., Larochelle, H., and Adams, R. (2012). Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, pages 2951–2959.

Srivastava, N. (2008). Improving Neural Networks with Dropout. *Geoffrey Hinton's Website*.

Sun, W., Liu, Y., Crowley, J. J., Chen, T.-h., Zhou, H., Chu, H., Huang, S., Kuan, P.-f., Li, Y., Miller, D., Shaw, G., Wu, Y., Zhabotynsky, V., Mcmillan, L., Zou, F., Sullivan, P. F., and Villena, F. P.-m. D. (2014). IsoDOT Detects Differential RNA-isoform Usage with respect to a Categorical or Continuous Covariate with High Sensitivity and Specificity arXiv : 1402 . 0136v1 [stat . AP] 1 Feb 2014.

Westra, H.-J., Peters, M. J., Esko, T. o., Yaghootkar, H., Schurmann, C., Kettunen, J., Christiansen, M. W., Fairfax, B. P., Schramm, K., Powell, J. E., Zhernakova, A., Zhernakova, D. V., Veldink, J. H., Van den Berg, L. H., Karjalainen, J., Withoff, S., Uitterlinden, A. G., Hofman, A., Rivadeneira, F., 't Hoen, P. a. C., Reinmaa, E., Fischer, K., Nelis, M., Milani, L., Melzer, D., Ferrucci, L., Singleton, A. B., Hernandez, D. G., Nalls, M. a., Homuth, G., Nauck, M., Radke, D., Völker, U., Perola, M., Salomaa, V., Brody, J., Suchy-Dicey, A., Gharib, S. a., Enquobahrie, D. a., Lumley, T., Montgomery, G. W., Makino, S., Prokisch, H., Herder, C., Roden, M., Grallert, H., Meitinger, T., Strauch, K., Li, Y., Jansen, R. C., Visscher, P. M., Knight, J. C., Psaty, B. M., Ripatti, S., Teumer, A., Frayling, T. M., Metspalu, A., van Meurs, J. B. J., and Franke, L. (2013). Systematic identification of trans eQTLs as putative drivers of known disease associations. *Nature genetics*, **45**(10), 1238–43.

Xiong, Q., Ancona, N., Hauser, E. R., Mukherjee, S., and Furey, T. S. (2012). Integrating genetic and gene expression evidence into genome-wide association analysis of gene sets. *Genome research*, **22**(2), 386–97.

Zhou, V. W., Goren, A., and Bernstein, B. E. (2011). Charting histone modifications and the functional organization of mammalian genomes. *Nature reviews. Genetics*, **12**(1), 7–18.

6 SUPPLEMENTARY TABLES AND FIGURES

Table 3. Additionally performed MASSQTL experiments and their performances: Some additional experiments were performed in order to obtain additional insights. One investiged aspect was the requirement of models to be *deep*, in order to obtain good performance. As can be seen the shallow neural network implementation perform worse then the deep version. Also the large number of training data was discussed. In order to investigate the need for a Big Data approach 50% of the data was sample to compare performance. It could be concluded that the large number of training samples was in fact needed. As last, the performance on the synthetic dataset is given$\pm$ indicates the standard deviation.

	AUC	Precision	Recall	TQTL
Single layer neural network, with all features	85.5 ± 1.1	0.91 ± 0.01	17.2 ± 1.4	6834 ± 236
Single layer neural network, with only ENCODE features	83.8 ± 0.9	0.67 ± 0.01	12.6 ± 1.5	5013 ± 204
DNN, trained with 50% of training data	84.5 ± 0.9	0.84 ± 0.01	15.8 ± 1.2	6250 ± 172

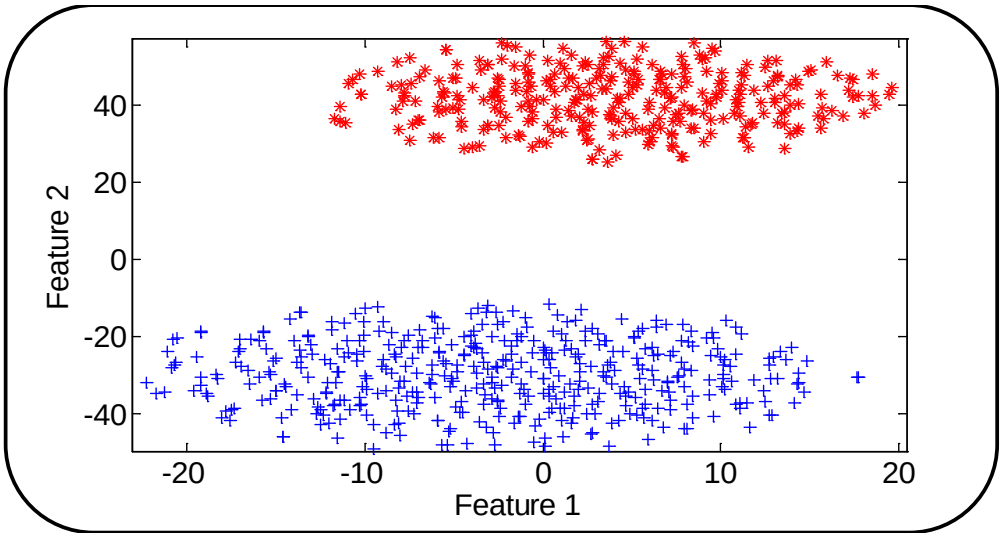

Fig. 4. Validation experiment result from DD. Is can be seen from this picture DD is able to retrieve the engineered sub-types in the data.