Project
Transplant kidney rejection
High Dimensional Data Analysis

Jan Alexander* Annabel Vaessens† Steven Wallaert‡

8/4/2020

Technical Report

Readers who are primarily interested in the overall conclusions without too much details on the methods can easily consult the **summarised results**, which accompanies this document. This is the technical report containing a more detailed discussion of the used methods and obtained results.

1 Exploratory Analysis

In this section general descriptive statistics are given and multiple methods for high dimensional data exploration are used.

1.1 Basic descriptive summary

In the complete dataset, 27% of the transplanted kidneys were rejected.

Several descriptive statistics (mean, sd, median, iqr, min, and max) were calculated for every gene and kidney rejection status combination. This resulted in 2 (accepted vs. rejected) distributions of every statistic across genes. Note that these statistics were only calculated to perform a visual inspection.

The resulted plot is presented in figure **??**. From this figure we can see there are, at least on this level, differences between groups. Most notable are the mean and median expression levels which tend to be closer to the overall mean (across groups) expression levels in the *accepted* group and more varying in the *rejected* group. There seems to be more variablity in the measures of dispersion in the *rejected* group. Finally there are minimal differences between the min/max expression level distributions, perhaps suggesting that gene expression levels in the *rejected* group are slightly less extreme than in the *accepted* group.

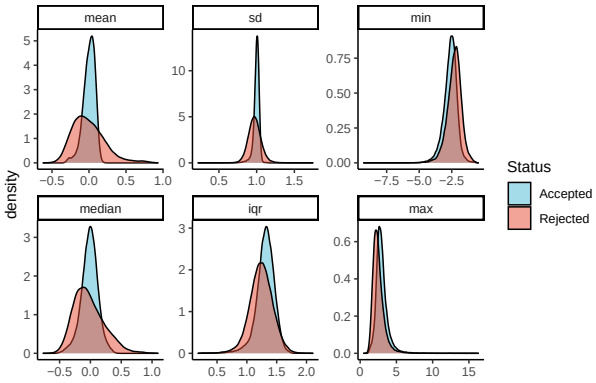


Figure 1: Descriptive statistics across grenes and between groups. Note that the data were centered before calculating the statistics.

1.2 Advanced exploratory analyses

Multiple methods for exploration and visualisation of high dimensional data were applied (sparse PCA, MDS, sparse LDA, LLE, ISOMAP, Sammon Mapping, Diffusion maps, and t-SNE), yet without clear results. Because the sparse LDA gave the best results we discuss the results here and refer to the appendices **??** to **??** for the results of the other techniques.

```
## Warning in lda.default(x, grouping, ...): variables are

## Warning in lda.default(x, grouping, ...): variables are

## Warning in lda.default(x, grouping, ...): variables are
```

The sparse LDA was performed to find potential candidate genes for future investigation. Due to computational constraints (our system ran out of memory) we needed to split the data set in 3 parts (each part consisting of 282 observations on 1.8225×10^4 genes). We considered this approach to be valid since we only used it as an exploratory tool. In total 116 genes (or 0.2121628%) had non-zero loadings.

*jan.alexander@ugent.be
†annabel.vaessens@vub.be
‡steven.wallaert@ugent.be

Because this still is a substantial amount, only the genes with loadings in absolute value larger than two standard deviations were further considered ($|v_i| > 2sd(v)$, with $i = \{1, ..., 116\}$, where v_i is the ith loading). This resulted in a list of 10 genes. The list of genes can be found in appendix **??**.

TO DO: plot invoegen, boxplot of scatterplot, + kiezen tussen ofwel op basis van 116 genen of best subset.

1.3 Conclusions Exploratory Analysis

No articulate distinction between the rejection status groups could be made with any of the used methods. This suggests that the main directions of the gene expression data do not correspond to the groups of the accepted/rejected kidneys.

Overall the results indicate that there certainly is relevant information at the genetic level w.r.t. transplant kidney rejection, in the sense that it was possible to make a certain distinction between groups, but not without a substantial overlap. The latter suggests that a relatively large part of the information given in the data set is not informative for the distinction of transplant kidney rejection.

2 Differentiating genes between kidney acceptance and rejection

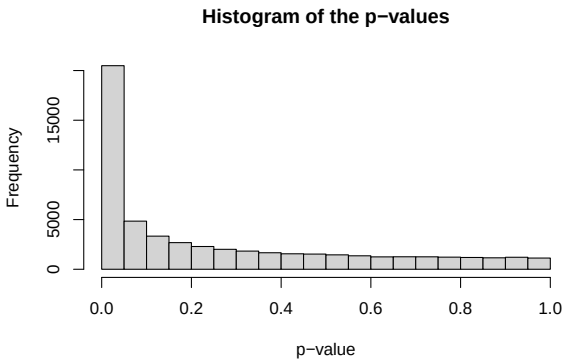
The hypothesis tests were performed on the uncentered data. Before testing, a visual inspection of several QQ-plots was done to verify whether the variables follow a normal distribution for both the accepted and rejected groups. The interested reader can find a few of these QQ-plots in the appendix "QQ-plots".

These QQ-plots showed that some genes were normally distributed, but also that some genes are not. Nevertheless, two-sided two-sample Welch t-tests were done on the uncentered data to determine whether the two groups can be differentiated based on the gene expression level for every gene. The Welch t-test was chosen because of unequal variances and difference in sample size between the two groups, even though it seems that not all genes were normally distributed.

The null hypotheses are $\mu_{rejected,i} = \mu_{accepted,i}$ with $i = \{1, ..., 54675\}$, and the alternative hypotheses are $\mu_{rejected,i} \neq \mu_{accepted,i}$ with $i = \{1, ..., 54675\}$.

To account for the multiple testing problem at this scale (54675 simultaneous hypothesis tests), the FDR is controlled at 0.10 through application of the method of Benjamini and Hochberg (1995).

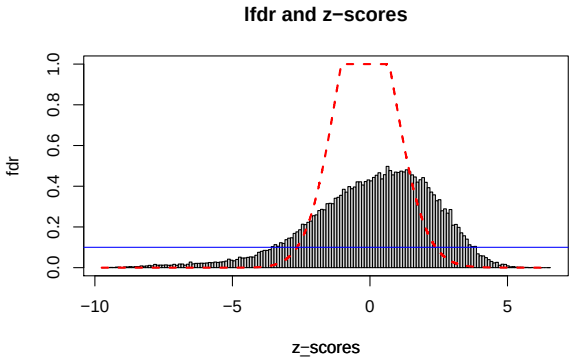
Histogram of the p-values



This histogram does not follow the uniform distribution, it has many small p-values. This indicates that for many variables, the null hypothesis can probably be rejected. Next, theBH95 method is done and the FDR is controlled at 10%.

There were 18081 rejected null hypotheses. As such we conclude that the gene expression differs between the accepted and rejected kidneys for those 18081 genes. As the FDR is controlled at 10%, it is expected to have around 1808 false discoveries.

Next, the z-scores are plotted and compared to the local false discovery rate.

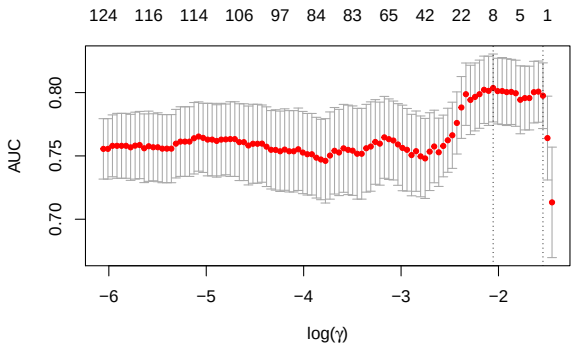


From the figure **??** can be concluded that a fdr < 0.1 can be obtained for negative z-scores, smaller than -2.5 and for very large z-scores, larger than 2.5. For these z-scores, it is more likely that when rejecting the null hypotheses, a true discovery is made.

3 Prediction of kidney transplant rejection

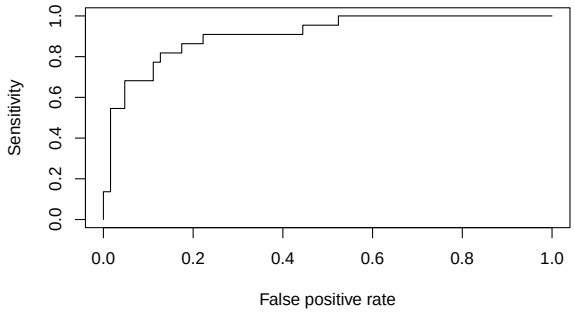
The dataset is split into a training and test dataset.

3.1 Lasso regression

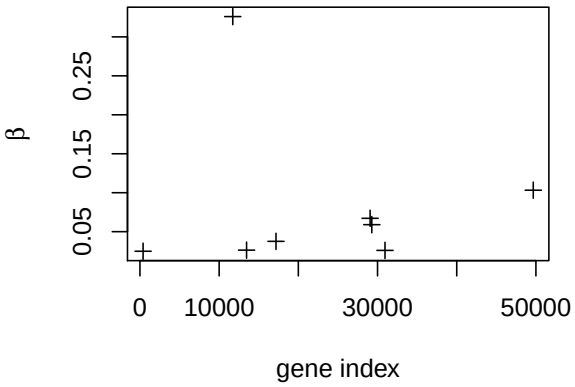


In the figure above, one can see that for γ equal to 0.1280675 , the area under the curve (AUC) is maximal for the train dataset based on a 10-fold cross-validation over the train dataset.

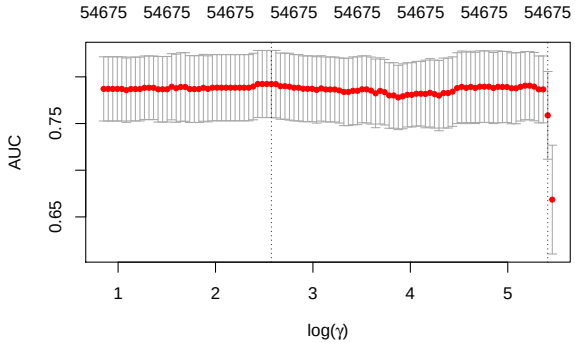
The ROC curve, estimated with the cross-validation dataset, is shown below:



This model uses 8 of the features. This is a considerable dimensional reduction. This is illustrated below. This figure shows the loadings of the 8 selected values.

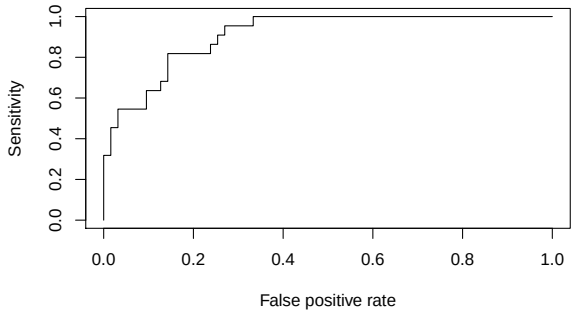


3.2 Ridge regression



In the figure above, one can see that for γ equal to 13.1081041 , the area under the curve (AUC) is maximal for the train dataset based on a 10-fold cross-validation over the train dataset.

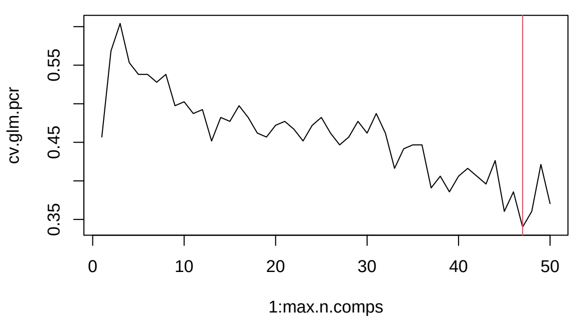
The ROC curve, estimated with the cross-validation dataset, is shown below:



AUC is 0.911255411255411.

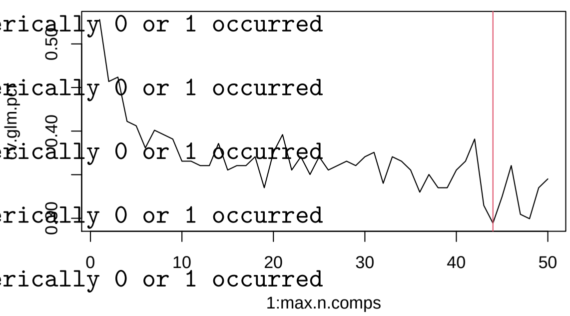
3.3 Principal component regression

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

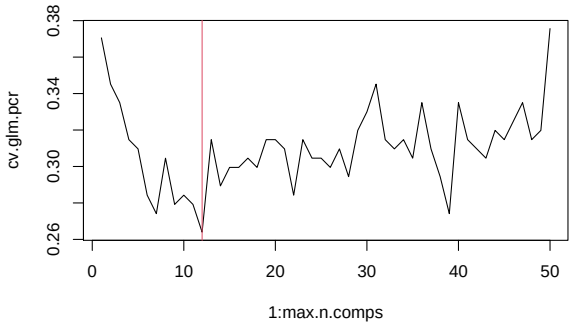



```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or
## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or
```



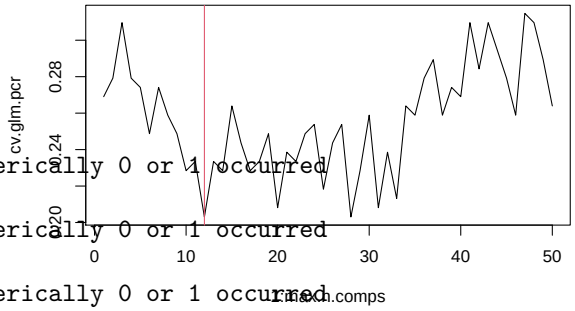

```
## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or
```



Warning: glm.fit: fitted probabilities numerically 0 or

Warning: glm.fit: fitted probabilities numerically 0 or

Warning: glm.fit: fitted probabilities numerically 0 or



Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

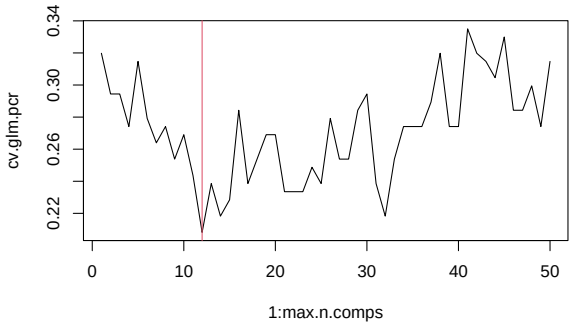
Warning: glm.fit: fitted probabilities numerically 0 or

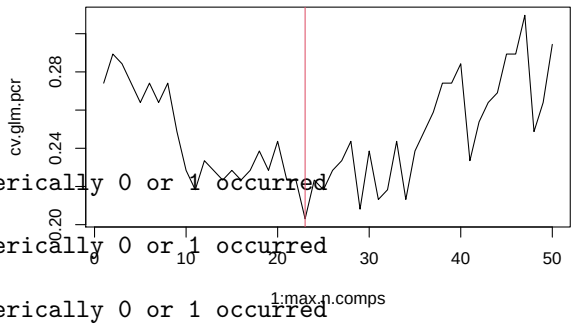
Warning: glm.fit: fitted probabilities numerically 0 or

Warning: glm.fit: fitted probabilities numerically 0 or

Warning: glm.fit: fitted probabilities numerically 0 or

Warning: glm.fit: fitted probabilities numerically 0 or





Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Warning: glm.fit: algorithm did not converge

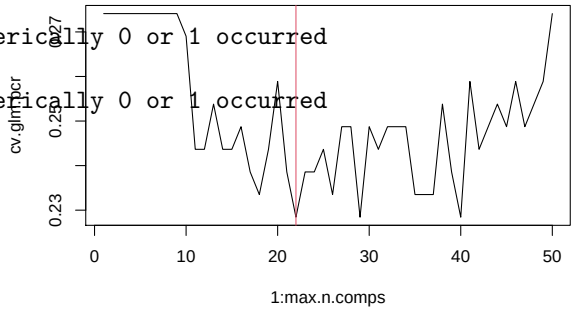
Warning: glm.fit: fitted probabilities numerically 0 or

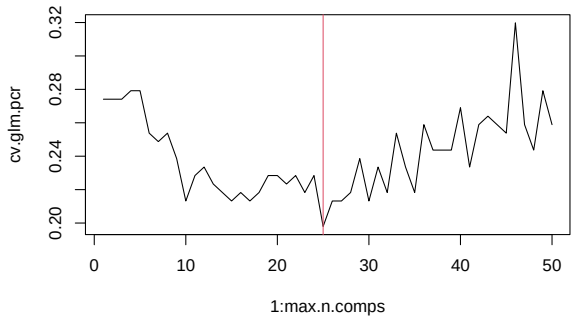
Warning: glm.fit: fitted probabilities numerically 0 or

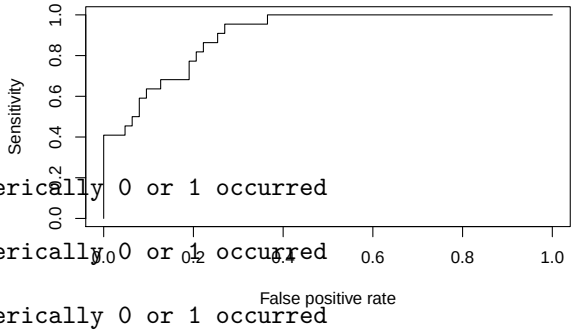
Warning: glm.fit: fitted probabilities numerically 0 or

Warning: glm.fit: fitted probabilities numerically 0 or

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```








```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

AUC is 0.9004329004329

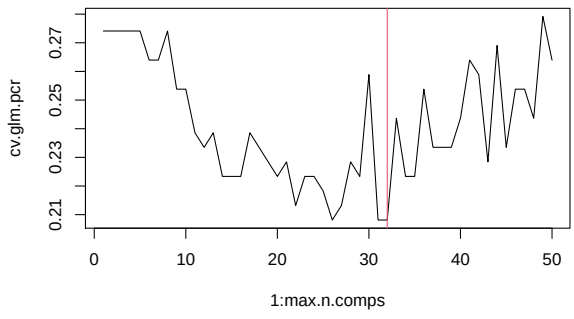
4 Conclusions

5 Appendices

5.1 Exploration methods for high dimensional data

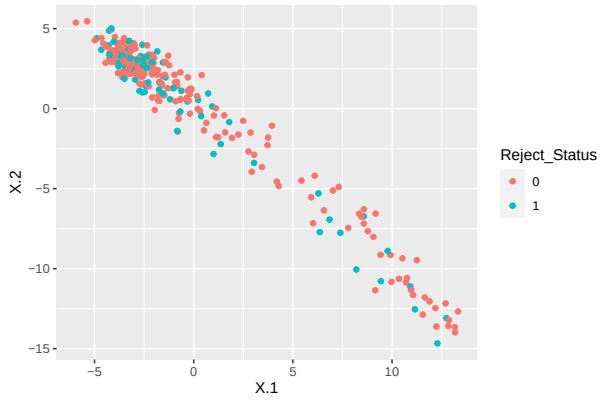
5.1.1 Sparse principle components analysis

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##  Fold  1  out of  5
##  Fold  2  out of  5
##  Fold  3  out of  5
##  Fold  4  out of  5
##  Fold  5  out of  5

## Call:
## SPC.cv(x = GeneExpression_C)
##
##
## Cross-validation errors:
##    Sumabsvs CV Error  CV S.E. # non-zero v's
## 1     1.200  3072720 2299.034            2.8
## 2     1.622  3072681 2299.517            5.4
## 3     2.044  3072639 2299.810            7.6
## 4     2.467  3072590 2297.630           10.2
```



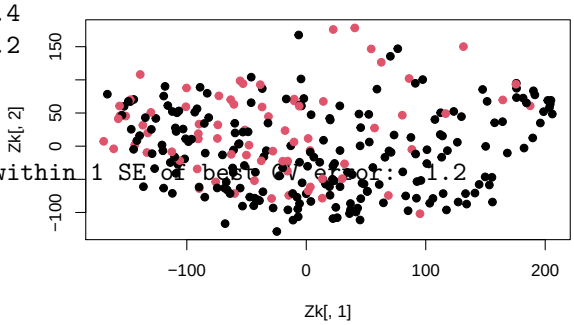
```
## 5     2.889  3072526 2297.306          14.5
## 6     3.311  3072459 2294.582          20.2
## 7     3.733  3072386 2292.579          24.6
## 8     4.156  3072296 2293.438          28.0
## 9     4.578  3072192 2295.093          33.4
## 10    5.000  3072074 2296.127          40.2
##
##  Best sumabsv value (lowest CV error):  5
##
##  Smallest sumabsv value that has CV error within 1 SE of best CV error:  1.2
```

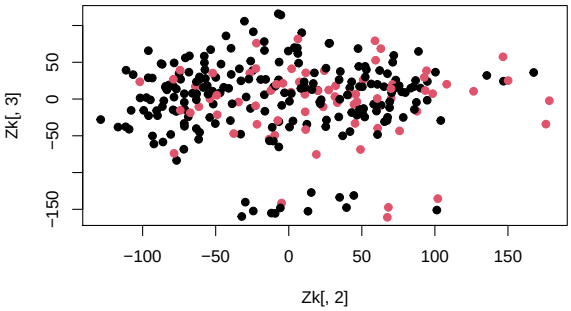
```
## 12345678910111213141516171819 20
## 12345678910111213141516171819 20
```



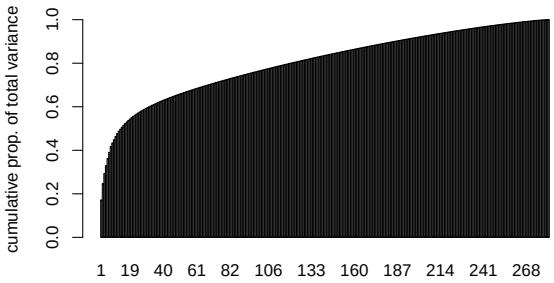
Unfortunately, naive sparce principle component analysis cannot be used to make a distinction between the accepted and rejected kidneys.

3.1.2 Multi-dimensional scaling:





In the biplots of the three first dimensions of the svd (**??**), no distinction can be made between rejected and accepted kidneys.

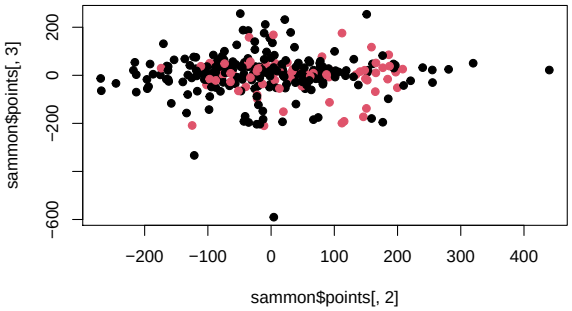


In the scree plot **??** it can be seen that the two first dimensions account for only 25% of the total variance in the dataset and the first three dimensions for 29%. To account for 80% of the total variance, 120 dimensions are needed.

5.1.3 LLE (locally linear embedding)

Locally linear embedding is a nonlinear dimension reduction method which finds a low-dimensional representation for each points' local neighbourhood by linearly approximating the data in each neighbourhood and returing these coordinates of lower dimension (Roweis and Saul, 2000).

From figures **??** can be seen that no distinction between the accepted and rejected kidneys can be made.



5.1.4 ISOMAP

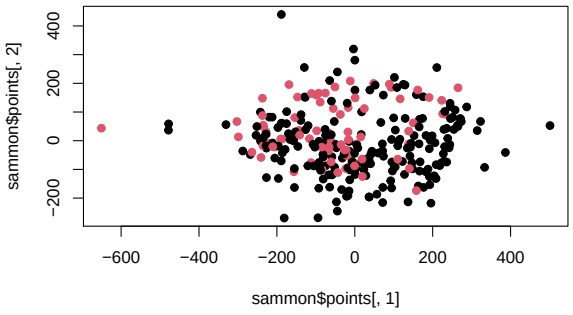
ISOMAP is a a nonlinear dimension reduction technique presented by Tenenbaum, Silva and Langford in 2000. It preserves rather the global properties of the data. It uses multidimensional scaling but then with incorporating the geodesic distances imposed by a weighted graph of the k neighbours of each point.

The parameter k is varied manually so that the maps are optimal. From figure **??** can be seen that with ISOMAP, it is also not possible to make a distinction between the group of accepted and rejected kidneys.

5.1.5 Sammon mapping

Sammon mapping is also a nonlinear dimension reduction method. The cost function of the Sammon method is similar to that of MDS, except that it is adapted by dividing the squared error in the representation of each pairwise Euclidean distance by the original Euclidean distance in the high-dimensional space. In this way, local structure is better preserved than in MDS. The result is in figures **??**: no distinction can be made between the two groups.

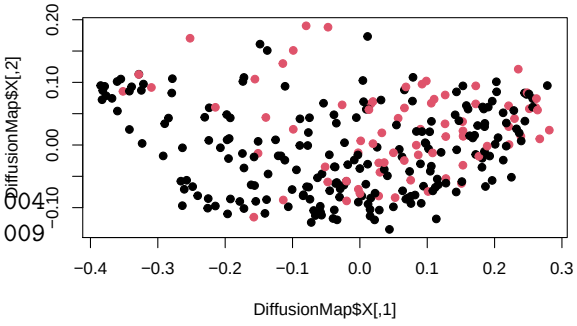
```
## Initial stress        : 0.28532
## stress after  10 iters: 0.21735, magic = 0.004
## stress after  20 iters: 0.16971, magic = 0.009
## stress after  28 iters: 0.13928
```



5.1.6 Diffusion maps

Diffusion mapping is also a nonlinear dimension reduction method based on the eigenvectors and eigenvalues of the data. The global strucure of the data is mapped by integration of local similarities at different scales. It works with probablities of point x randomly walking to y, and is based on the heat diffusion equation.

```
## Performing eigendecomposition
## Computing Diffusion Coordinates

## Warning in min(which(lam < 0.05)): no non-missing argum
## Inf

## Used default value: 3 dimensions
## Elapsed time: 149.1 seconds
```



From figure **??**, no distinction between the two groups can be made with diffusion maps.

5.1.7 t-SNE

t-stochastic neighbor embedding is a dimension reduction technique for visualising high dimensional data with a focus on preserving the local structure. The resulting plot can be seen in figure **??** and indicates again that a simple distinction between the two groups cannot be made. Yet, there seems to be

roughly two groups that differ in heterogeneity: one largely heterogeneous group (upwards left in the plot) and one group that is less heterogeneous, though far from homogeneous (middle to downward right in the plot).

```
## sigma summary: Min. : 2.26380310632853 |1st Qu. : 3.28803119583798 |Median : 3.72444955432746 |Mean

## Epoch: Iteration #100 error is: 14.4700050945832

## Epoch: Iteration #200 error is: 0.598830022722575

## Epoch: Iteration #300 error is: 0.592248383358065

## Epoch: Iteration #400 error is: 0.592158042814787

## Epoch: Iteration #500 error is: 0.592157966365228

## Epoch: Iteration #600 error is: 0.592157966249188

## Epoch: Iteration #700 error is: 0.59215796624904

## Epoch: Iteration #800 error is: 0.59215796624904

## Epoch: Iteration #900 error is: 0.59215796624904

## Epoch: Iteration #1000 error is: 0.59215796624904
```

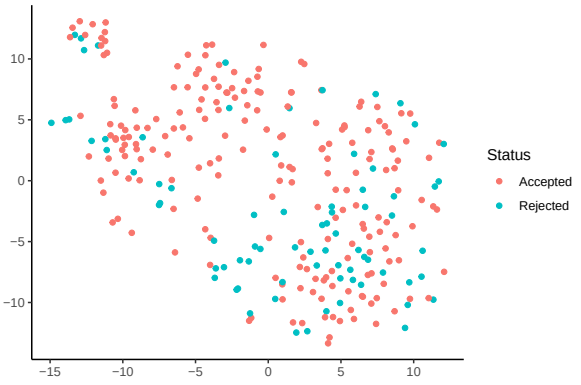
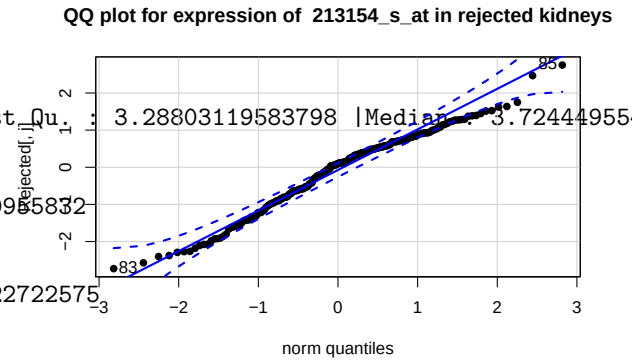
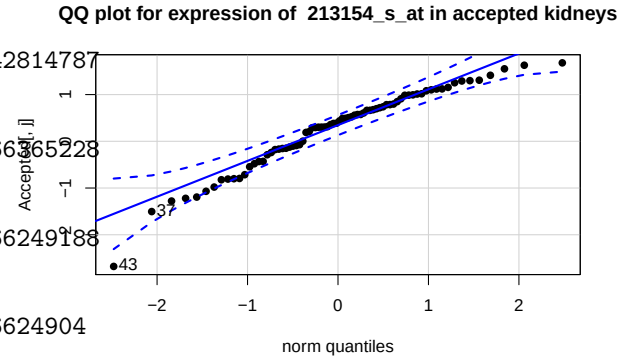
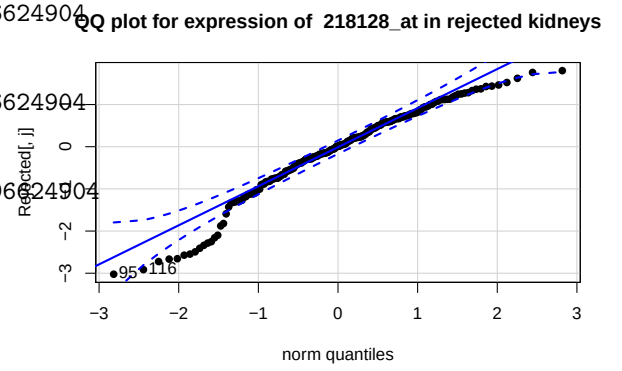


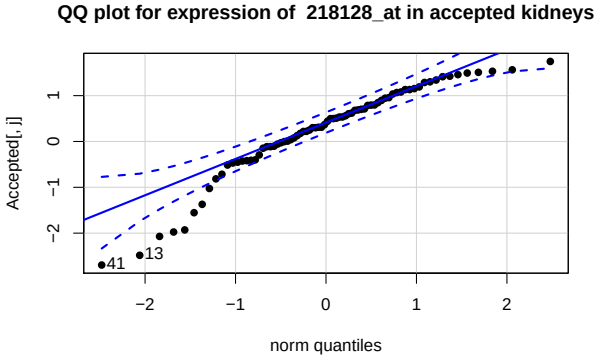
Figure 2: Two dimensional representation of the data through application of t-SNE

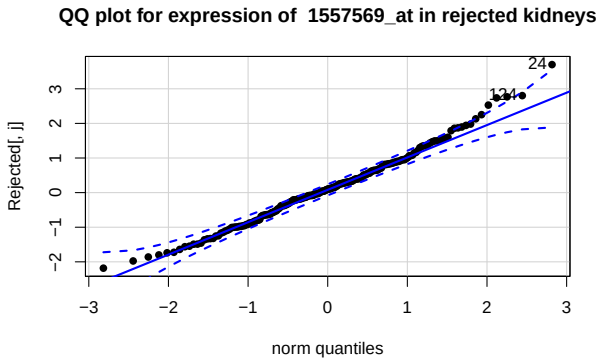
5.2 QQ-plots

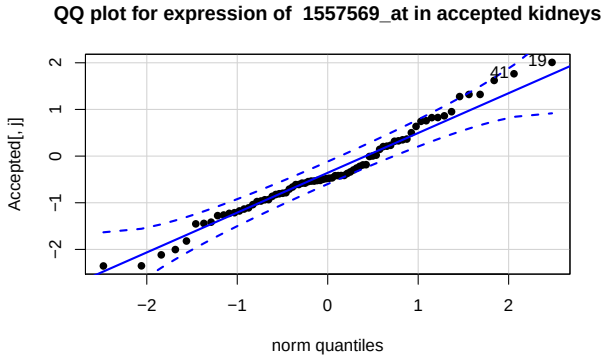












6 References

Benjamini Y and Hochberg Y, 1995. Controlling The False Discovery Rate - A Practical And Powerful Approach To Multiple Testing. Journal of the Royal Statistical Society. Series B: Methodological 57, 289-300.

Roweis ST and Saul LK, 2000. Nonlinear dimensionality reduction by locally linear embedding. Science, 290, 2323-2326.

Tenenbaum JB, De Silva V and Langford JC, 2000. A global geometric framework for nonlinear dimensionality reduction. Science, 290, 2319-2323.