Creating a Kidney Transplant Risk Calculator using GEO datasets (Group 24)

460352996 470066919 480145820 480407614

GitHub code repository is here

This version was compiled on June 3, 2020

Background- Kidney failure is the final stage of renal disease and poses a major threat to the body as the excretory system fails to function properly. To combat kidney failure patients can choose two forms of treatment in terms of medical intervention; renal dialysis or organ transplantation. Renal dialysis is used to provide some kidney functionalities by removing waste, maintaining a safe balance of potassium, sodium and bicarbonate levels in the body and helps regulate blood pressure (kidney dialysis). However, renal dialysis places several restrictions on the patient, compromising their quality of life and is not the ideal choice for patients with end stage kidney disease. Alternatively, organ transplantation is a life saving treatment and is greatly preferred over renal dialysis as it has the potential to offer a better quality of life for the patient posing less restrictions on diet, working lifestyle and posing fewer long term health problems. While organ transplantation greatly preferred, kidney organ allocation has posed itself as a major resource allocation problem. Donors and patients need to be matched effectively and accurately to each other to not only preserve the life of the patient but also, maximise functionality of these limited kidney organs.

Transplant | Regression | Prediction | Immunosuppression

Aim and background.

Aim of the project. With this problem in mind we developed a tool to aid in the effective and accurate allocation of donor organs to their respective patients. The developed risk calculator will assist practitioners in their decision making and shall even inform the prescriptions for immunosuppressive drugs. The risk calculator was developed with the intention that it would be used in a clinical setting where shared decision making is implemented. According to Gordan (2013), shared decision making promotes patient centered care. It permits the integration of the nephrologist's expertise on renal allograft dysfunction with the patient's values and beliefs concerning future treatment. Within this clinical setting, our hope would be that the risk calculator provides an opportunity of discussion that concerns the nature of treatment prior to, during and post organ transplantation.

Multidisciplinary context. Changes being made.....

Target Audience. Should we have this here??????

Methods.

Data collection and developed models.

Part 1. Predicting acute rejection Acute Rejection (AR) calculator is based on data taken from the Gene Expression Omnibus, GSE120396, GSE120649 and GSE131179. We merged the three datasets together in order to achieve a larger sample size for better accuracy, to identify outliers and provide a smaller margin of error. However, because of the differences between datasets, such as number of genes, scale (counts per million) and the use of ensembl ids rather than gene ids, we have to perform some preprocessing before they can be merged.

Inside the GSE120396, GSE120649 and GSE131179 folders are 88, 16 and 34 zipped files with file extension txt.gz respectively. Each of these files has the gene expression count for one patient. They are unzipped and placed together into a table format.

To resolve the issue of different units of measurements, we perform data standardization. In particular, cpm and log2 transformation were performed on GSE120649 and GSE131179. The ensemble id in GSE120649 and GSE131179 were converted to gene symbols using the 'EnsDb.Hsapiens.v79' library from the Ensemble based annotation library from Bioconductor.

The three datasets were joined based on common gene symbols. To account for any technical differences in gene expression measurements between the three experiments, we perform quantile normalization. Quantile normalization is one of the most widely adopted preprocessing methods for analyzing microarray data. Quantile normalization reduces batch effect and removes technological noise by scaling the variables to have values between 0 and 1, ensuring the distribution of gene expressions from each array are the same. This allowed more robust predictions that can be generalised to different sequencing platforms (Qiu et.al, 2013).

From the high dimension of gene expression data between stable and acute rejection patients, we performed feature selection using the 'limma' function to identify and extract the top 100 significant genes of the kidney rejection dataset. These top 100 genes were used to build a comprehensive model that can predict an acute re

Function to read in the files, unzip them, place them into a table

Preprocess data in the GSE Raw folder into a table, save it as a txt file

```
gse_396 = preprocessing_fn("../data/GSE120396_RAW/")
write.csv(gse_396, "GSE120396_expression_matrix.txt")
```

Install the Ensembl based annotation library from Bioconductor Function takes a GSE table as input, and returns a GSE dataframe with two new columns, 'gene symbols' and 'gene ids', from the 'ensembl ids' using the library above.

```
library("EnsDb.Hsapiens.v79")

emsembl_to_symbol <- function(gse) {
    G_list <- select(EnsDb.Hsapiens.v79,
        key = rownames(gse),
        columns = c("SYMBOL"),
        keytype = "GENEID")
    df = as.data.frame(gse)
    df_symbol <- merge(df,
        G_list, by.x = 0,
        by.y = "GENEID",
        all.x = TRUE)
    return(df_symbol)
}
```

Log the two datasets, 649 and 179, and use the ensembl function to get the gene symbols for both datasets. Finally, merge the datasets by gene symbols.

```
# log 649!
log_649 <- log2(gse_649)
log_649[log_649 == -Inf] <- 0
df_649_symbol = emsembl_to_symbol(log_649)

# 179!
log_179 <- log2(gse_179)
log_179[log_179 == -Inf] <- 0
df_179_symbol = emsembl_to_symbol(log_179)

# left join both
df_both <- merge(df_179_symbol,
    df_649_symbol, by.x = "Row.names",
    by.y = "Row.names",
    all.x = TRUE)
df_both
```

We plot the distribution of patient gene expression measurements using a boxplot to see if we have to remove any patient that has measurements unlike the others.

```
p <- ggplot(melt(gse_396),
    aes(x = Var2, y = value)) +
    geom_boxplot(outlier.colour = "black",
        outlier.shape = 16,
        outlier.size = 0.5,
        notch = FALSE) +
    theme(axis.text.x = element_text(angle = 45,
        hjust = 1)) +
    labs(x = "patient",
        y = "expression value") +
    theme_minimal()
p
```

Part 2. Estimating Time to de novo DSA Presence The data and data dictionary used were provided by Dr. Jermaine wong.

Certain columns in the data were mutated as follows. 0 and 1 variables in the Sex_Cat columns were mapped to FEmale and Male respectively. The values in the agetxn columns were rounded and grouped into 3 categories '25-35','36-45', and '46-55'. C2epletMM were also grouped into categories,'<= 30 MM' and ' > = 30MM'.

Here we first create a survival object using the Surv() function, to which we feed in the "C2dnDSA", "C2daystodnDSA" information, which in turn is used as the dependent variable for the Kaplan-Meier Formula. The data set is filtered by the selected user input(Age group and Gender) and passed in as the Independent variable. We plot the Kaplan Meier Curve using the survfit() function.

Based on previous research studies conducted it was found that there was no significant correlation between BMI category and graft survival (Papalia et.al, 2010) Thus we focused on only using age and gender as phenotypic information when stratifying the data to fit a particular recipient.

Part 3. Predicting Operational Tolerance To predict operational tolerance, we collected the GSE22229 dataset that contained raw gene expressions from tolerant patients and normal patients (which still required immune suppression for stable graft function). GSE22229 dataset was a raw CEL dataset containing 58 zipped files, each pertaining to the gene expression for one patient that was either tolerant (19), undergoing normal immunosuppression (27), or standard controls (12). In this case, we only retained the tolerant and immunosuppressed patients for our analysis.

CEL files are data files created by the Affymetrix DNA microarray image analysis software, and contain estimated probe (sequence of DNA base pairs) intensity values extracted from biological chips called Affymetrix Genechips. Each probe was mapped to a specific gene symbol using the GPL570 Chip Description File (CDF).

Since the data is in the newer Affymetrix Arrays format (Gene ST arrays), we utilised the 'oligo' library and its functions, 'list.celfiles' and 'read.celfiles', to read a list of CEL files into our directory. This list was then converted from an AffyBatch object into an ExpressionSet (i.e. gene expression) using the 'rma' function from the 'pd.mogene.2.0.st' library. The 'rma' function, short for Robust Multichip Average, also simultaneously log2 transforms and normalises the gene expressions.

After converting our ExpressionSet object into a dataframe, we then investigated the gene expression distribution amongst patients using a boxplot to ensure that the data has been normalised and can be used for further processing.

The boxplot above demonstrates strong similarity in gene expression between patients and so our dataframe can be further analysed without concern of batch effects between sample measurements.

After quantile normalization and batch effect removal, we performed feature selection. Firstly, genes that were lowly expressed in both groups were removed by using the filterByExpr() function from the edgeR package; these genes do not provide much biological meaning and removing them allows for less statistical tests to be performed, as well as allowing greater reliability in observing the variance between different groups (Law et al., 2018).

We then selected the top 100 most differentially expressed genes between the two groups using multiple t-tests from the `limma` package. Finally, a review by Massart et al. (2017) suggested a collection of genes that were highly differential between tolerant and normal patients, and so these were also added to our final training dataset (if they weren't already filtered for previously).

Installing essential tac packages

```
library(limma)
library(affy)
library(oligo)
library(pd.mogene.2.0.st)
library(mogene20sttranscriptcluster.db)
library(GEOquery)
```

```
setwd("data/GSE22229_RAW/")
celFiles <- list.celfiles()
affyRaw <- read.celfiles(celFiles)
eset <- rma(affyRaw)

setwd("../../")
write.exprs(eset, file = "tolerance.txt")
my_frame <- data.frame(exprs(eset))
write.table(my_frame,
    file = "tolerance.txt",
    sep = "\t")
```

Evaluation Strategies. We decided to implement penalised logistic regression methods when creating a predictive model for Part 1 and Part 3 of our risk calculator. Logistic regression was utilised as it provides a probabilistic output for a specific risk, which may be more informative than a binary outcome. Furthermore, the penalised nature of some methods (e.g. Ridge, LASSO, Elastic Net) can address the overfitting or multicollinearity issue prevalent in large p, small n situations prevalent in gene expression data.

For Part 1 and Part 3 respectively, our model was trained using a 50-repeated 5-fold cross validation (CV) procedure. The performance of our model in predicting the CV test-set under Ridge, LASSO and Elastic Net methods were evaluated using three primary metrics: accuracy, AUC and the Brier Score.

In the case of class imbalance within the training dataset, the accuracy metric may suggest an inflated performance. As such, the AUC and Brier Score were also calculated.

- The AUC is a more robust metric with less bias to class size. Briefly, it can be thought of as the probability that a true-positive sample (e.g. AR patient) has a greater predicted risk than true-negative samples (e.g. normal patient).
- The Brier Score meanwhile complements the AUC by checking that the predicted risk of a sample is actually similar to the true value. For example, in a true-positive case (i.e. label = 1) with a predicted risk of 0.8, the Brier Score quantitatively measures how close the 0.8 value is to 1. Better predictions are reflected as a lower Brier Score.

To select our final models for Part 1 and Part 3 respectively, we quantitatively compared the accuracy, AUC and Brier Score from different penalised models (Ridge, LASSO, and Elastic Net) using boxplot visualisations.

We also evaluate the model based on robustness. Robustness refers to how well the model achieves its aim when it is applied to the general population, which is our target audience. We qualitatively analysed robustness by estimating how well the model will adjust if it is applied to real-world data, which may have potential issues such as missing information.This evaluation metric ensures the external validity of the product.

Results.

Final Model.

Deployment Process.

Discussion and Conclusion.

Potential shortcomings.

Future work and improvements.

Conclusion.

Reference List.

1.. Dayoub, J.C. Cortese, F., Anzic, A, 2018, The Effects of Donor Age on Organ transplants: a review and implications for aging research, Experimental Gerontology, Vol 110, pp. 230-240, Retrieved from <>

2.. Dorr, C. R., Oetting, W. S., Jacobson, P. A., & Israni, A. K. (2018). Genetics of acute rejection after kidney transplantation. Transplant International, 31(3), 263-277.

3.. Edgar R., Domrachev M., Lash AE. (2002). Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. Nucleic Acids Res, 30(1),207-10.

4.. Gordon, E.J., 2013, Opportunities for Shared Decision Making in Kidney Transplantation, American Journal of Transplantation, Vol.13, no.5 pp. 1149-1158.

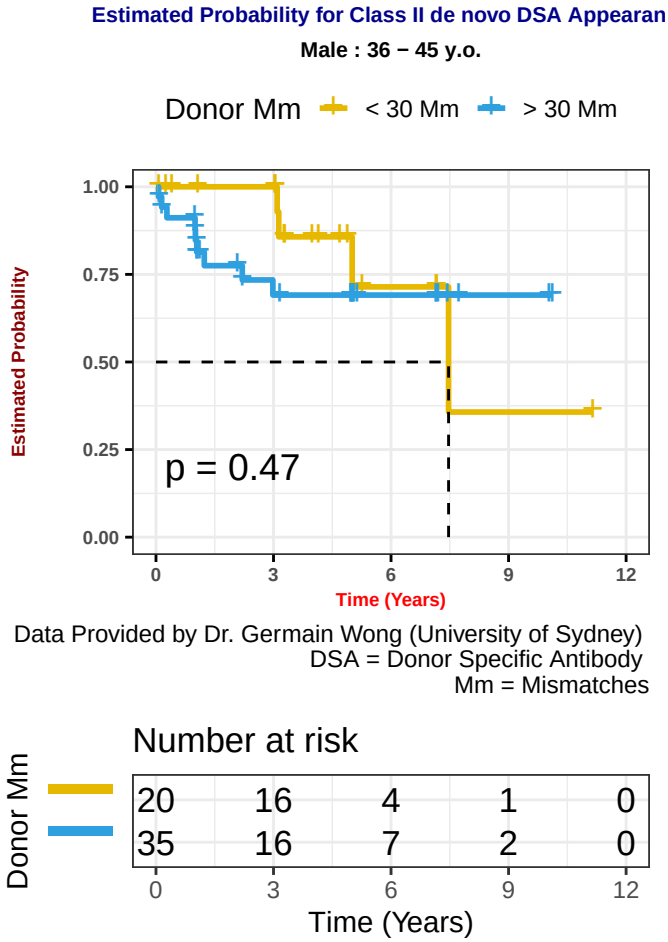
5.. Kleinbaum, D.G., 2005 Klein, M., 'Introduction to Survival Analaysis' in Survival Analysis: A self learning text, Springer, New York, NY pp.1-43.

6.. Massart, A., Ghisdal, L., Abramowicz, M., & Abramowicz, D. (2017). Operational tolerance in kidney transplantation and associated biomarkers. Clinical & Experimental Immunology, 189(2), 138-157.

7.. Tambur, A.R. 2018, HLA-epitope Matching or Eplet Risk Stratification: The Devil is in the Details, Front Immunol, Vol.9

Appendixes.

Appendix A. Kaplan-Meier curve: Estimated Probability for Class II de novo DSA Appearance



Appendix B. Penalised Logistic Regression: Risk of Acute Rejection

Appendix C. Penalised Logistic Regression: Reliance on Immunosuppression

Contribution Statement.

Johanna Jones.

Andrew Auwyang.

Eva Pu.

Niruth Bogahawatta.

Alex Wong.

460352996 470066919 480145820 480407614