

Final Presentation

Group 3

November 27, 2017

Group 3 Objective: Association between dependent and independent variables

- ▶ Create summaries and visualizations of how the dependent variable is associated with different independent variables. Here, we will try to discover if there are characteristics of the drugs that are associated with effectiveness against TB. This group will need to come up with ways (and code) to analyze that in the data. This might include generalized linear models, scatterplots, and possibly other supervised learning methods.

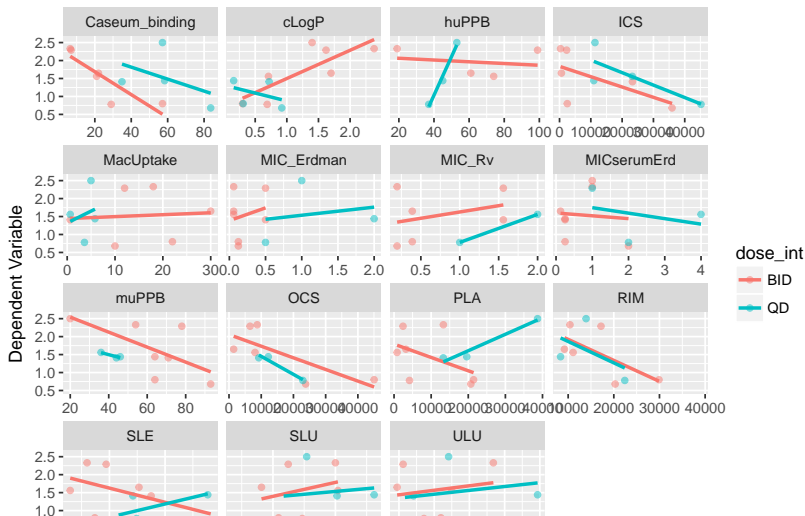
Visualize univariate variables in a scatterplot function, by drug dose

To visualize univariate variables correlations with the outcome variables, a faceted scatterplot colored by dose was created. Inputs to this function include `peak_trough`, which evaluates the peak or trough (Cmax or Trough) respectively, of drug following injection. The `dep_var` options are either ELU or ESP (lung levels and spleen levels, respectively).

Scatterplot

Warning: Removed 8 rows containing non-finite values (st

Warning: Removed 8 rows containing missing values (geom



Interpretation

- ▶ It appears that the relationship between independent and dependent variables are dependent on dose for variables cLogP, huPPB, PLA, and SLE. The others are not dependent on dose.
- ▶ Dose may be an important factor to consider when evaluating the relationship between independent and dependent variables.

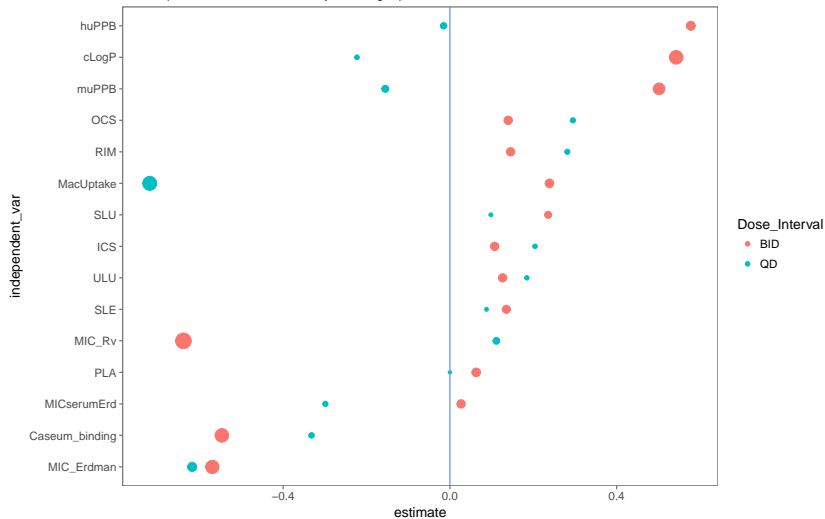
Fitting Linear Models Function (Example inputs Peak_trough = Cmax, Dep_var = ELU)

I fitted a linear model regressing dependent on independent variables to assess the relationships between them. Inputs to this function include peak_trough (options are Cmax or trough), or the dep_var (options are ELU or ESP). The units of scale for each variable were normalized so that we could compare across coefficients.

Coefficients Plot

Linear model coefficients as function of independent variables,
by drug dose and model uncertainty

smaller points have more uncertainty than larger points

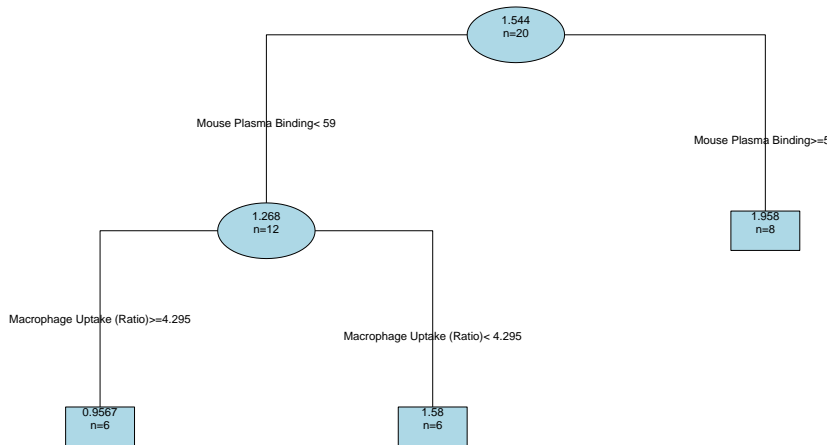


Interpretation

-The coefficient plot generated by this function shows each independent variable on the y axis and the respective model coefficient on the x axis. If the coefficient is negative, for example, as it is with MacUptake, an interpretation would be for every unit of change in the MacUptake, the ELU will decrease by 0.5 Units. Therefore, MacUptake has a negative relationship with ELU, decreasing the ELU. The diameter of the point represents the level of certainty of the coefficient in this model.

Regression Tree Function

Testing regression_tree function



Interpretation

- ▶ The number at the top of each node is indicating the mean of the outcome variable for the observations in that node (mean of 1.5 for node 1). Below each node is indicating what each split was based on. Splits are chosen based on a complexity parameter. Starting from node 1, the first split is made so that it leads to the greatest possible reduction in RSS. Node 3 is a terminal node because it only has 4 observations, which was the minimum number of observations a node can have to be considered (set in our function parameters). Given the 16 observations in node 2, another split is made that again gives the greatest possible reduction in RSS. This process continues until either the `min_split` or the `min_bucket` parameters are fulfilled for each node from the preset function parameters.

LASSO Function

The function for LASSO required the following inputs:

- ▶ Dependant variable
- ▶ Dose
- ▶ Dataframe, though the dataframe default is efficacy_summary

The output in the end is coefficients with smaller coefficients being restricted to zero.

Testing LASSO function:

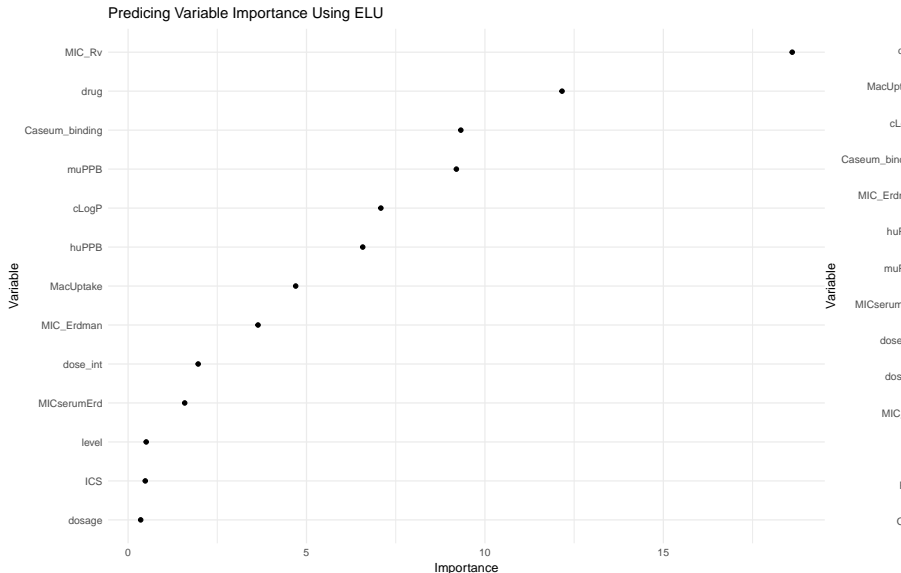
Interpretation:

- ▶ This function outputs raw coefficients and an intercept on a penalized maximum likelihood model. Models dropped from the model are represented with zeros. It computes a LASSO penalty for small coefficients and drops them, resulting in only the coefficient with the most leverage remaining.

RandomForest Function

- ▶ The user specifies which dependent variable they would like to use (either ELU or ESP). The user can also specify a dataset they would like to use, if one is not provided then a default dataset is utilized. The function `var_importance` takes the input and outputs a graph displaying the which variables are the best predictors of the input (either ELU or ESP).

Testing Random Forest



Interpretation

- ▶ This function utilizes the function `randomForest` to predict which variables are the most important predictors of the associated outcome. This model works by randomly creating small data nodes that are split using the best predictor from a subset of predictors randomly chosen at each node. In order to determine variable importance, the algorithm looks at how much the predictive error increases as all variables remain unchanged while one is permuted. The resulting output shows the % increase in the mean standard error for each variable considered individually. The higher the number the more important the variable for model building.