Prediction pipeline steps

1. Import data and tidy (data frame or tibble needs to be a 2D array with each participant on one row and each feature in one column).

Example:

A computer screen capture

Description automatically generated with low confidence

1. Split the data into testing and training sets (this isn’t always going to be possible if we have less data so if you have N>100 try this pipeline- if it doesn’t work (i.e., the prediction performance (on the test data) is terrible) then switch to the nested cross-validation models tutorial.
2. If working with fMRI or MRI data, we should regress out certain parameters (age, gender, mead framewise displacement) and use the residual values as features in the subsequent models. This is done in just the train set then we reduce the test set to just those features selected from the test set.
3. Manual feature selection to reduce the number of features we input into our model building initially. You can place all x hundred features into a model, but this can increase the search space drastically and result in poorer performance of the model- so reducing the features to only pertinent ones can improve model performance. There are several ways one can do this- from recursive feature selection and selection by filtering – which are important when just doing cross-validation as you still keep the test and train sets (folds) separate), to methods that don’t incorporate the outcome measure - PCA (reduce features into principal components), removing features that are highly correlated with each other, or simply by removing those that do not correlate with the outcome at a given p value threshold using a univariate filter- if doing test train I think this approach is fine, especially with large fMRI datasets.
4. Build your models. We are now going to take