

# Linear Regression with `lm()` using fold sampling

David Banh - AskExplain

04/01/2022

## Linear Regression using sample summaries runs faster, retains accuracy

Summarising or Folding the samples is a way of reducing the total number of samples to a manageable number in order to run prediction algorithms on modern day machines. The folded samples are then unfolded to predict the full dataset.

```
# Removes one feature at a time and uses it as the variable to be predicted (y variable)

# Total permutations :
permutation_test_number <- 100

# Run SVD decomposition of samples to a reduced sample space
source("./decompose_sample_space.R")

# Run gcode encoding of samples to a reduced sample space
source("./encode_sample_space.R")
```

A way to fold the total number of samples while retaining the original sample structure is done via Generative Encoding (gcode): [https://github.com/AskExplain/gcode/tree/alpha\\_test\\_v2022.1](https://github.com/AskExplain/gcode/tree/alpha_test_v2022.1)

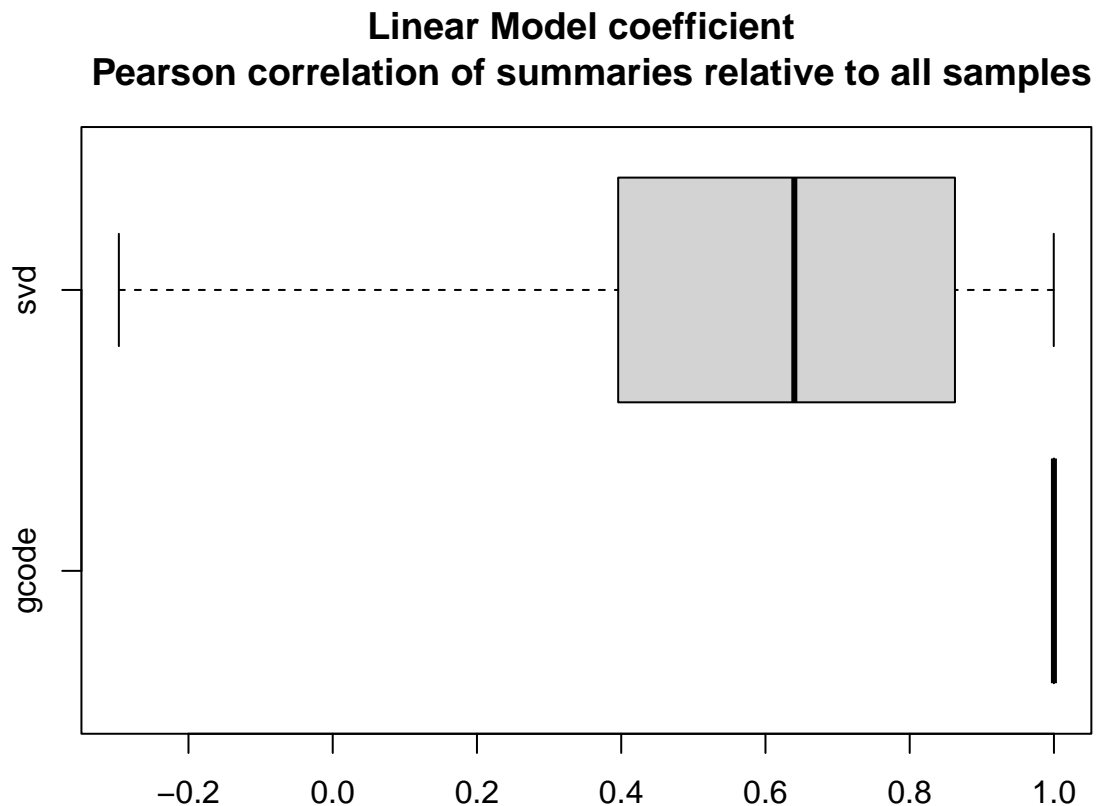
Boxplots of Linear Model coefficients for the features after summarisation compared to using all samples

```

boxplot(data.frame(gcode = total_cor.gcode,
                  svd = total_cor.svd),
        outline = F,
        horizontal = T,
        main = "Linear Model coefficient \n Pearson correlation of summaries relative to all samples")

```

In essence, the linear regression coefficients are calculated for the features using all samples, and then correlated with the corresponding coefficients when using the summary samples from either SVD or gcode. The output is a correlation  $R^2$  value for each run, giving a distribution for multiple runs on different permutations of the training set.



Boxplots of mean absolute error and runtime are plotted for every unique run of linear regression via base package via `lm()`.

```
# Mean Absolute Error
```

```
boxplot(data.frame(gcode = total_mae.gcode,
```

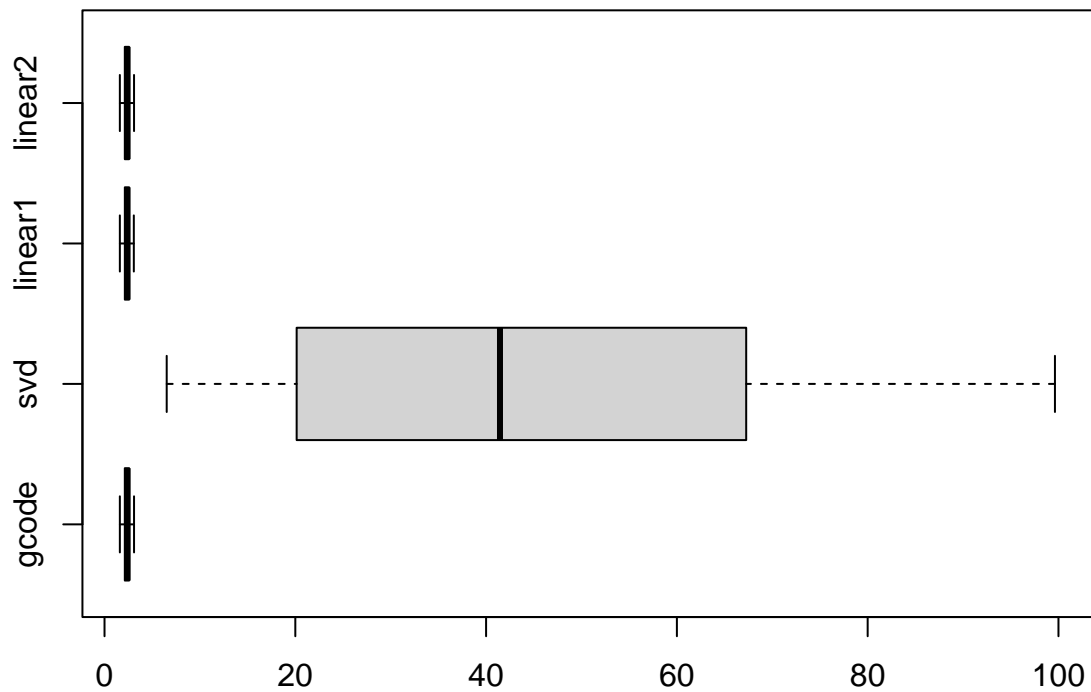
```

        svd = total_mae.svd,
        linear1 = total_mae.lreg.1,
        linear2 = total_mae.lreg.2),
outline = F,
horizontal = T,
main = "Mean Absolute Error distribution")

```

Of great importance, the runtime does not include the running of the SVD or gcode algorithms.

## Mean Absolute Error distribution



*# Runtime*

```

boxplot(data.frame(gcode = total_time.gcode,
                    svd = total_time.svd,
                    linear1 = total_time.lreg.1,
                    linear2 = total_time.lreg.2),
outline = F,

```

```
horizontal = T,  
main = "Runtime distribution")
```

