# Test DoseRider

Pablo Monfort

2023-06-29

1. Create a GAMM formula:

```
# Define the formulas for the models

# Define the formulas for the models
base_formula <- create_gamm_formula(response = "counts",
                                    fixed_effects = "dose",
                                    random_effects = "gene",
                                    model_type = "base")

linear_formula <- create_gamm_formula(response = "counts",
                                      fixed_effects = "dose",
                                      random_effects = "gene",
                                      model_type = "linear")

cubic_formula <- create_gamm_formula(response = "counts",
                                     fixed_effects = "dose",
                                     random_effects = "gene",
                                     model_type = "cubic")

print(base_formula)
```

```
## [1] "counts ~ s(gene, bs = ’re’) "
```

```
print(linear_formula)
```

```
## [1] "counts ~  dose + s(gene, bs = ’re’) "
```

```
print(cubic_formula)
```

```
## [1] "counts ~  dose + s(gene, bs = ’re’) + s(dose, bs = ’cr’, k = 5) "
```

2. Create a summarized experiment:

```
omic <- "proteomic"
proteomics_data <- read.csv(
  "../..//Projects/TOX/project_tox/data/proteomic/BCI_vsn_impute_batch.csv",
  check.names = F, row.names = 1)
```

```
metadata <- read.csv(
  "../..//Projects/TOX/project_tox/data/proteomic/target.txt", sep = "\t")

colnames(metadata) <- c("SAMPLE", "CHANNEL", "EXPERIMENT",
                        "CELLTYPE", "CONCENTRATION", "INDEX","GROUP","name","dose","sample")

metadata <- metadata[metadata$CELLTYPE == "B",]
rownames(metadata) <- metadata$sample
proteomics_data <- proteomics_data[rownames(metadata)]

se <- create_summarized_experiment(proteomics_data, metadata)
```

Replace `proteomics_data` with your actual proteomics data object and `metadata` with the corresponding metadata object.

3. Estimate model parameters, only for RNASeq data:

```
if ( omic == "rnaseq"){
parameters <- estimate_model_parameters(se, formula)
}
```

4. Load gene sets from ConsensusPathDB:

```
file_path <- "../data/CPDB_pathways_genes.tab"
gmt <- load_consensupathdb_genesets(file_path)
gmt <- filter_gmt_by_size(gmt = gmt, minGenesetSize = 200, maxGenesetSize = 1200)
```

5. Perform the analysis on gene sets:

```
res <- perform_analysis(se,
                        gmt,
                        base_formula,
                        linear_formula,
                        cubic_formula,
                        dose_col = "dose",
                        sample_col = "sample",
                        omic = "proteomic")

save(res,file = "../data/res.rda")
```

### Check the results

```
load("../data/res.rda")
table(res$FDR < 0.05)
```

```
##
## FALSE  TRUE
##   100    22
```

```r
head(res[res$FDR < 0.05 & !is.na(res$FDR),], 25)
```

```
##                                                                       Geneset
## 1                                       Prion disease - Homo sapiens (human)
## 2   Pathways of neurodegeneration - multiple diseases - Homo sapiens (human)
## 4                                      Focal adhesion - Homo sapiens (human)
## 7                             Diabetic cardiomyopathy - Homo sapiens (human)
## 9                                   Alzheimer disease - Homo sapiens (human)
## 11                                 Huntington disease - Homo sapiens (human)
## 14                                     Thermogenesis - Homo sapiens (human)
## 25                                  Parkinson disease - Homo sapiens (human)
## 53                                                               Translation
## 55                                                      Metabolism of lipids
## 57                                                         Metabolism of RNA
## 59                                                                DNA Repair
## 61                                              Cellular responses to stress
## 65                                 Metabolism of amino acids and derivatives
## 68                                      Organelle biogenesis and maintenance
## 69                                                     Biological oxidations
## 70                                                  Neutrophil degranulation
## 73                                      SLC-mediated transmembrane transport
## 74                                              Transport of small molecules
## 92                                                      Innate Immune System
## 105                                   Cellular responses to external stimuli
## 116                          Processing of Capped Intron-Containing Pre-mRNA
##     Geneset_Size Genes    Base_AIC   Base_BIC  Linear_AIC Linear_BIC
## 1            273   139  -3161.7728 -2306.4459  -3175.1655 -2313.7242
## 2            475   202  -5314.1660 -3998.0166  -5327.8488 -4005.2112
## 4            201    72  -1345.2586  -947.3144  -1351.7160  -948.3159
## 7            203   104  -2684.3369 -2073.2024  -2713.4801 -2096.5204
## 9            369   165  -4038.7578 -2996.0800  -4055.2652 -3006.3012
## 11           306   160  -3915.9869 -2909.6652  -3935.1549 -2922.5775
## 14           232   100  -2399.2246 -1815.4191  -2431.4451 -1841.8522
## 25           249   135  -3328.1562 -2501.2141  -3351.8181 -2518.7895
## 53           307   203  -6214.9299 -4891.1043  -6217.4191 -4887.1019
## 55           645   219  -6061.3269 -4617.2699  -6105.1246 -4654.4943
## 57           583   331 -10989.6804 -8672.8933 -11022.1411 -8698.3711
## 59           322    80  -2644.6938 -2194.5797  -2653.6021 -2197.9265
## 61           553   238  -4842.3005 -3253.4600  -4842.7225 -3247.2314
## 65           339   158  -4534.1855 -3542.3264  -4543.5107 -3545.4097
## 68           231    68  -1372.5340 -1000.2350  -1388.0649 -1010.3669
## 69           219    62  -1889.1577 -1555.0165  -1898.5024 -1559.0554
## 70           486   261  -4564.4728 -2798.6318  -4577.5391 -2804.9538
## 73           243    38   -481.9744  -294.2558   -502.5536  -310.0172
## 74           641   149  -2729.7882 -1802.9219  -2754.5757 -1821.5242
## 92          1064   395  -7135.2061 -4302.1716  -7154.2288 -4314.0350
## 105          568   239  -4839.8518 -3243.3623  -4840.4261 -3237.2818
## 116          241   178  -6383.3229 -5245.2747  -6393.9157 -5249.5066
##       Cubic_AIC  Cubic_BIC P_Value_Linear P_Value_Cubic          FDR
## 1    -3175.1621 -2313.7057      1.226e-04     2.000e-03 1.284211e-02
## 2    -5327.8453 -4005.1924      1.058e-04     2.000e-03 1.284211e-02
## 4    -1360.2070  -947.1350      4.000e-03     2.000e-03 1.284211e-02
## 7    -2713.4793 -2096.5164      4.678e-08     1.487e-06 2.015711e-05
## 9    -4055.2608 -3006.2765      2.537e-05     4.987e-04 4.056093e-03
## 11   -3935.1546 -2922.5749      6.668e-06     1.481e-04 1.379471e-03
## 14   -2435.5871 -1831.7197      1.024e-08     1.557e-07 2.713629e-06
## 25   -3356.7105 -2508.2708      7.081e-07     6.395e-06 7.092636e-05
## 53   -6284.1974 -4938.7324      3.800e-02     6.161e-14 3.758210e-12
## 55   -6111.4241 -4646.8747      3.473e-11     1.339e-10 4.083950e-09
## 57  -11070.2444 -8726.9760      9.097e-09     2.854e-16 3.481880e-14
## 59   -2659.3219 -2189.0107      1.000e-03     5.000e-03 2.904762e-02
## 61   -4882.8770 -3272.6601      1.280e-01     2.885e-08 7.039400e-07
## 65   -4598.7988 -3586.2082      9.861e-04     5.123e-13 2.083353e-11
## 68   -1410.5776 -1017.3412      4.152e-05     3.080e-07 4.697000e-06
## 69   -1903.1323 -1552.2554      9.762e-04     3.000e-03 1.830000e-02
## 70   -4577.5372 -2804.9451      1.448e-04     2.000e-03 1.284211e-02
## 73    -502.4896  -309.6995      3.291e-06     7.755e-05 7.884250e-04
## 74   -2756.2110 -1814.5171      4.047e-07     4.997e-06 6.096340e-05
## 92   -7154.1922 -4313.8478      7.171e-06     1.583e-04 1.379471e-03
## 105  -4880.2362 -3262.7986      1.160e-01     3.479e-08 7.073967e-07
## 116  -6393.6447 -5243.2578      5.129e-04     7.000e-03 3.881818e-02
```

6. Plot smooth curves:

```
lP <-list()
j = 1
for (genest_name in unique(res[res$FDR < 0.05 & !is.na(res$FDR),]$Geneset)) {


  i <- find_geneset_index(gmt = gmt, geneset_name = geneset_name)

  geneset <- gmt[[i]]$genes

  long_df <- prepare_data(se, geneset, dose_col="dose",
                         sample_col = "sample", omic = "proteomics")

  cubic_results <- fit_gam(cubic_formula, long_df)

  p <- plot_smooth(cubic_results, long_df) + ggtitle(genest_name)
  lP[[j]] <- p
  j = j + 1
}
combined_plot <- cowplot::plot_grid(plotlist = lP, ncol = 3)

# Save the combined plot
ggsave("../../plots/Significant_Geneset.png", plot = combined_plot, width = 22, height = 22, limitsize =
```

7. Obtain Trend Chande Dose (TCD)

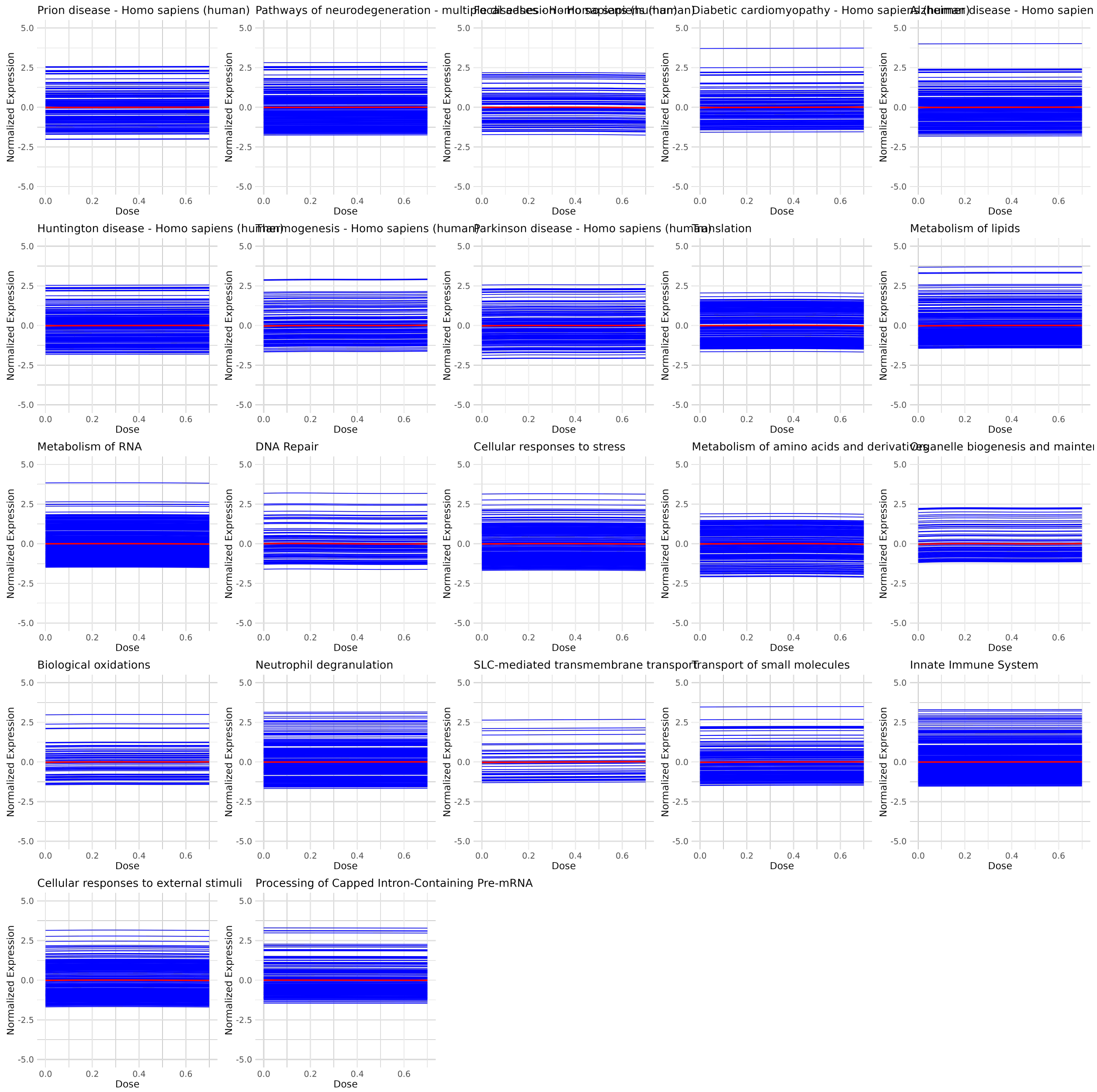

Figure 1: Smooth Curves for Significant Gene Sets. This figure showcases the smooth curves obtained for the gene sets identified as significant in the proteomic data analysis. Each subplot represents a distinct gene set, and the curves portray the relationship between dose and normalized expression levels. The analysis utilized the cubic model formula to fit the gene set data and employed a stringent false discovery rate (FDR) threshold of 0.05 to determine significance.