ABA dataset analysis from Deforges 2019

Marc Galland

2023-02-10

Contents

Setup

Add a table of contents

Link to R Markdown guide (click me)

Disabling warnings and messages

This will keep your final PDF report clean from execution alarms, unnecessary text, etc.
This code chunck sets global options for the execution of each code chunk. You can disable warnings and messages globally this way.

```r
knitr::opts_chunk$set(echo = TRUE,
                      warning = FALSE,
                      message = FALSE,
                      collapse = TRUE)
```

Introduction

Q1 (0.5 point): In the publication of Deforges et al. 2019, the first paragraph of the “Material and Methods” section describes how plants were sown and treated (see paragraph below). What are the full names of each of the 4 hormones used? “ ” ” Arabidopsis thaliana seeds were germinated on agar-solidified half-strength MS medium for 10 d, after which the seedlings were flooded with a solution of half-strength MS containing 5 µM IAA, 10 µM ABA, 10 µM MeJA, 10 µM ACC, or no hormone for the untreated control. After 3 h of incubation, roots and shoots were split and harvested separately. For each of the 12 experimental conditions, 3 independent biological replicates were carried out at different times. “ ” ” Names are: - IAA: auxin,
- ABA: abscisic acid, - ACC: 1-amino-1-cyclopropane, - MeJA: methyl jasmonate.

Q2 (0.5 point): Can you name 3 different types of RNA that can be studied using RNA sequencing? - Long non-coding RNAs.
- Messenger RNAs.
- microRNAs.

Q3 (0.5 point): What is the sequencing platform used in this experiment? How many reads were obtain on average per sample?
Illumina HiSeq 2500, about 30 million reads per sample.
> “The libraries were sequenced on a HiSeq 2500 Illumina sequencer and about 30 million of paired-end reads per sample were obtained.”

Q4: (0.5 point): In the article, can you find a good complete one-sentence long definition of cis-NATs?
>“Cis-Natural Antisense Transcripts (cis-NATs), which overlap protein coding genes and are transcribed from the opposite DNA strand, constitute an important group of noncoding RNAs.”

Exercise 1: data import

We first load the `tidyverse` package that contains most of the data transformation functions we will need.

```
suppressPackageStartupMessages(library("tidyverse"))
suppressPackageStartupMessages(library("apeglm"))
suppressPackageStartupMessages(library("DESeq2"))
```

Import gene counts

```
raw_counts <- read.csv(
  file = "../gene_counts_and_samples2conditions/dataset02_ABA_arabidopsis_root_raw_counts.csv",
  header = TRUE,
  stringsAsFactors = FALSE) %>%
  # for DESeq subsequent data import
  column_to_rownames("gene")

# first five rows
head(raw_counts, n = 5)
##           root_control_1 root_control_2 root_control_3 root_ABA_1 root_ABA_2
## AT1G01010           2029           1481           2694       1353       1174
## AT1G01020           1626           1608           1895       1068       1413
## AT1G01030            150            230            375        108        232
## AT1G01040           3174           2599           4260       2271       2471
## AT1G01046             70             42            115         38         57
##           root_ABA_3
## AT1G01010       1404
## AT1G01020       1342
## AT1G01030        223
## AT1G01040       3410
## AT1G01046         93
```

Q5: Can you determine how many genes are present in the table?

```
nrow(raw_counts)
## [1] 28642
```

There are **28642** genes in the “dataset_01_IAA_arabidopsis_root_raw_counts.csv” table.

Q6 (1 point): determine the minimum and maximum gene expression in the control condition and in the hormone-treated condition.
Hint: if you use the tidyverse package to do this, use “pivot_longer()” to get your data tidy and create a new column for the biological replicate number. Find the gene that has the maximum count in the control condition and auxin-treated conditions.

```
raw_counts %>%
  rownames_to_column("gene") %>%
  pivot_longer(- gene, values_to = "counts", names_to = "sample") %>%
  separate(sample, into = c("tissue","condition","rep"), sep = "_") %>%
  group_by(condition) %>%
  summarise(minimum = min(counts),
            maximum = max(counts)) %>%
  head()
## # A tibble: 2 x 3
##   condition minimum maximum
##   <chr>       <int>   <int>
## 1 ABA             0  267156
## 2 control         0  519267
```

Gene with maximum expression in control condition.

```
# 519267
raw_counts %>%
  rownames_to_column("gene") %>%
  pivot_longer(- gene, values_to = "counts", names_to = "sample") %>%
  separate(sample, into = c("tissue","condition","rep"), sep = "_") %>%
  filter(counts == "519267")
## # A tibble: 1 x 5
##   gene      tissue condition rep   counts
##   <chr>     <chr>  <chr>     <chr>  <int>
## 1 AT3G09260 root   control   3     519267
```

Gene with maximum expression in ABA-treated condition.

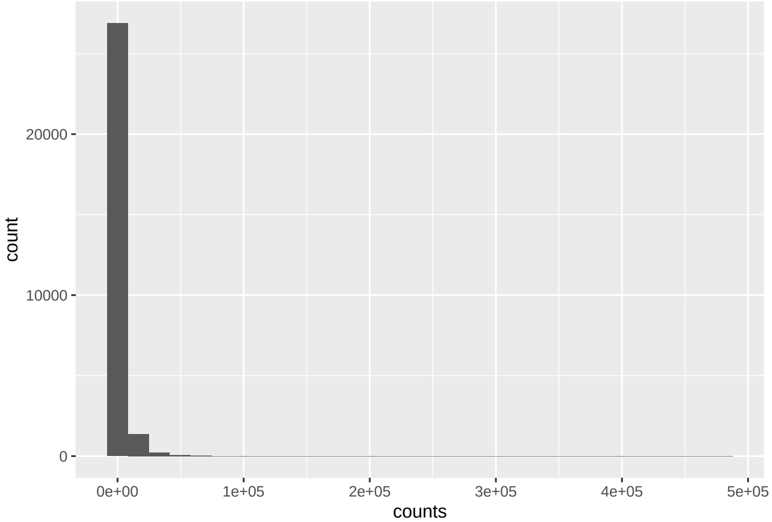
```
# 267156
raw_counts %>%
  rownames_to_column("gene") %>%
  pivot_longer(- gene, values_to = "counts", names_to = "sample") %>%
  separate(sample, into = c("tissue","condition","rep"), sep = "_") %>%
  filter(counts == "267156")
## # A tibble: 1 x 5
##   gene      tissue condition rep   counts
##   <chr>     <chr>  <chr>     <chr>  <int>
## 1 AT3G09260 root   ABA       3     267156
```

Conclusion: the same gene has the maximum expression in both conditions. From the TAIR website, it says: > “Encodes beta-glucosidase.The major constituent of ER bodies. One of the most abundant proteins in Arabidopsis seedlings."

Distribution of counts

Q7 (1 point):
A) plot the distribution of the gene counts of one sample using either the base R hist() function or the ggplot2 geom_histogram() function. What can you say about the distribution of the gene counts?
B) How can you display the distribution of an scale that better represents the distribution? Think about data transformation or the axis.

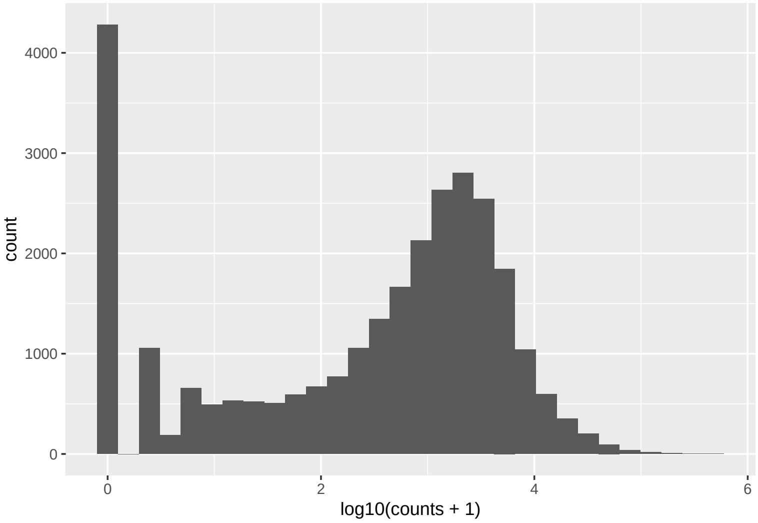
```
raw_counts %>%
  rownames_to_column("gene") %>%
  pivot_longer(- gene, values_to = "counts", names_to = "sample") %>%
  filter(sample == "root_control_1") %>%
  ggplot(., aes(x = counts)) +
  geom_histogram()
```



The count distribution is very skewed with a lot of data with counts < 50,000 counts. A few genes have a very high count number with the maximum being 479,700 counts.

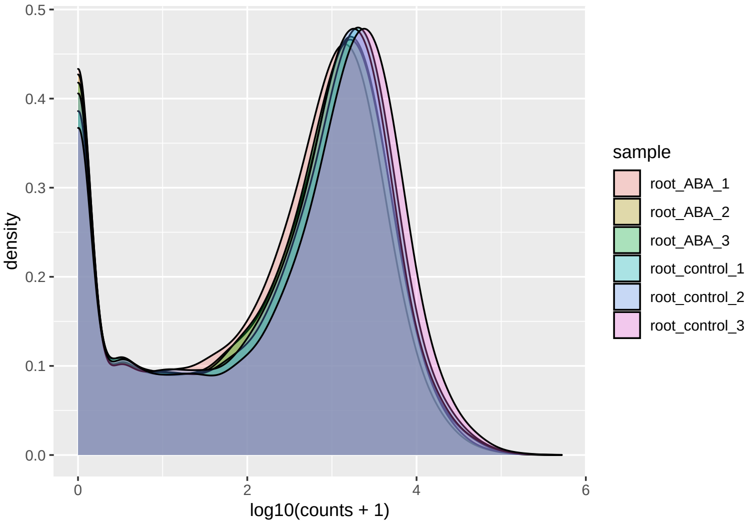
We can transform the data before plotting or use a transformed scale. Here, I use a log10 transformation with an offset of 1 (so that genes with 0 counts become log10(0+1)=0).

```
raw_counts %>%
  rownames_to_column("gene") %>%
  pivot_longer(- gene, values_to = "counts", names_to = "sample") %>%
  filter(sample == "root_control_1") %>%
  ggplot(., aes(x = log10(counts + 1))) + # log10 transform with offset
  geom_histogram()
```



Bonus: all samples overlaid.

```
raw_counts %>%
  rownames_to_column("gene") %>%
  pivot_longer(- gene, values_to = "counts", names_to = "sample") %>%
  ggplot(., aes(x = log10(counts + 1), fill = sample)) + # log10 transform with offset
  geom_density(alpha=0.3)
```



Conclusion: this means that the samples have comparable count distributions.

Exercise 2: differential expression

This `DESeqDataSet` object is used to store both data (gene counts) and metadata (sample to experimental condition correspondence) in one unique R object. Functions can be directly be applied to this object and corresponding results stored within the same object.

Import samples to condition

Q8 (0.5 point) Upload the "arabidopsis_root_hormones_samples_to_condition.csv" file to your R virtual machine. Then import this file into R and name the object samples_to_conditions. - Make a table that shows the number of biological replicates by condition. - Filter this table to keep only "control" samples and the samples related to your hormone of interest. You should have 6 samples in total (3 control and 3 hormone-treated).

```
samples_to_conditions <- read.csv(
  file = "../gene_counts_and_samples2conditions/arabidopsis_root_hormones_sample2conditions.csv",
  stringsAsFactors = F)

table(samples_to_conditions$condition)
## 
##     ABA     ACC control     IAA    MeJA 
##       3       3       3       3       3
```

```
samples_to_conditions_filtered <- filter(
  samples_to_conditions,
  condition == "ABA" | condition == "control")
samples_to_conditions_filtered
##           sample condition
## 1 root_control_1   control
## 2 root_control_2   control
## 3 root_control_3   control
## 4     root_ABA_1       ABA
## 5     root_ABA_2       ABA
## 6     root_ABA_3       ABA
```

Create the DESeqDataSetFromMatrix object

```
dds <- DESeqDataSetFromMatrix(countData = raw_counts,
                              colData = samples_to_conditions_filtered,
                              design = ~ condition)
```

You can have a quick peek at the number of genes, number of samples, etc. by calling the `dds` object.

```
dds
## class: DESeqDataSet 
## dim: 28642 6 
## metadata(1): version
## assays(1): counts
## rownames(28642): AT1G01010 AT1G01020 ... ATMG01400 ATMG01410
## rowData names(0):
## colnames(6): root_control_1 root_control_2 ... root_ABA_2 root_ABA_3
## colData names(2): sample condition
```

Call differential genes

Q9 (0.5 point): using the raw_counts and the samples_to_conditions, create a DESeqDataSet object called dds that will be used for DESeq2 differential analysis. Call the differential genes using the DESeq() function and call this object diff_genes. Filter this object to keep only the significantly differentially expressed genes (adjusted p-value < 0.01). Hint 1: the gene identifiers have to be assigned to the row names of raw_counts. Hint 2: convert the diff_genes object to a dataframe with “dds = as.data.frame(dds)”.

```
dds <- DESeq(dds)
```

```
diff_genes <- results(dds, contrast = c("condition", "ABA", "control")) %>% 
  as.data.frame() %>% 
  filter(padj < 0.01)
head(diff_genes)
##             baseMean log2FoldChange     lfcSE      stat       pvalue
## AT1G01060 2785.9357      0.7541364 0.1106737  6.814052 9.488759e-12
## AT1G01070 1628.6849      1.1583778 0.2567314  4.512022 6.421263e-06
## AT1G01110 1014.5105     -0.5181897 0.1279471 -4.050031 5.121086e-05
## AT1G01120 9312.6118      2.3131944 0.2115199 10.936062 7.749291e-28
## AT1G01180  309.4875     -0.6278540 0.2104442 -2.983471 2.849994e-03
## AT1G01210 1083.7434     -0.7285457 0.1255663 -5.802081 6.549708e-09
##                   padj
## AT1G01060 1.094629e-10
## AT1G01070 3.463210e-05
## AT1G01110 2.343705e-04
## AT1G01120 2.856286e-26
## AT1G01180 8.820534e-03
## AT1G01210 5.483610e-08
```

Number of diff genes and max log2FC

Q10 (0.5 point): how many genes are differentially expressed (adjusted p-value < 0.01)? How many genes are positively regulated in response to the hormone treatment?
How many genes are negatively regulated in response to the hormone treatment?

```
n_diff <- diff_genes %>% nrow()
pos <- diff_genes %>% filter(log2FoldChange > 0) %>% nrow()
neg <- diff_genes %>% filter(log2FoldChange < 0) %>% nrow()
```

- Total number of genes diff. regulated (padj < 0.01) is 7902 genes.

- Total number of genes *positively* & diff. regulated (padj < 0.01) is 4141 genes.

- Total number of genes *negatively* & diff. regulated (padj < 0.01) is 3761 genes.

Q11 (0.5 point) : display a table of the top 20 positively differentially expressed genes based on their log2 fold change.
What is the maximum positive log2fold change? Convert this log2 fold change to the untransformed fold change value = revert the log2 transformation.

```
top20 <-
  diff_genes %>%
  arrange(desc(log2FoldChange)) %>%
  mutate(fold_change = 2^log2FoldChange) %>%
  relocate(fold_change, .before = lfcSE) %>%
  head(n = 20)
top20
##              baseMean log2FoldChange fold_change     lfcSE      stat
## AT1G27461   450.50451      12.490991   5756.5584 1.2250299 10.196478
## AT1G04560  1406.28788      12.291266   5012.3297 1.0533910 11.668285
## AT4G31830   319.18754      11.993477   4077.5216 1.1994373  9.999253
## AT5G62800   223.29870      11.478503   2853.4725 1.2022971  9.547144
## AT3G17520  8951.16589      11.049334   2119.2446 0.5834087 18.939270
## AT2G33380  3963.24362      10.695898   1658.7695 0.4578457 23.361358
## AT5G59330   250.22952      10.680743   1641.4368 1.4078212  7.586718
## AT3G03341  2207.12938      10.345668   1301.2366 0.5105503 20.263759
## AT5G52300  6377.64985      10.104950   1101.2682 0.3052127 33.107896
## AT3G51810   152.25783       9.964051    998.7996 1.1972125  8.322709
## AT1G52690 37053.70206       9.915419    965.6920 0.1956953 50.667634
## AT1G76290   233.70093       9.701184    832.4292 1.0597912  9.153864
## AT3G02480 52069.14644       9.532293    740.4675 0.2088601 45.639609
## AT5G50360  1866.35112       9.375026    663.9940 0.3497922 26.801706
## AT3G15670  4859.21357       9.279350    621.3878 0.3018093 30.745739
## AT5G45950  1435.79619       9.237277    603.5279 0.3996510 23.113358
## AT2G42540  2809.94955       9.207797    591.3207 0.3592415 25.631216
## AT5G43840    43.90151       9.129367    560.0326 1.2507993  7.298827
## AT5G40790    36.87936       8.879640    471.0185 1.2444369  7.135468
## AT5G06760 23304.52862       8.743607    428.6352 0.2660167 32.868634
##                  pvalue          padj
## AT1G27461  2.055923e-24  6.252456e-23
## AT1G04560  1.851179e-31  8.241010e-30
## AT4G31830  1.535507e-23  4.487759e-22
## AT5G62800  1.333211e-21  3.477218e-20
## AT3G17520  5.414248e-80  1.261310e-77
## AT2G33380 1.056538e-120 4.527077e-118
## AT5G59330  3.281089e-14  4.827084e-13
## AT3G03341  2.686481e-91  7.242933e-89
## AT5G52300 2.287693e-240 3.920942e-237
## AT3G51810  8.597519e-17  1.588125e-15
## AT1G52690  0.000000e+00  0.000000e+00
## AT1G76290  5.493293e-20  1.271086e-18
## AT3G02480  0.000000e+00  0.000000e+00
## AT5G50360 3.086634e-158 2.116108e-155
## AT3G15670 1.394127e-207 1.592957e-204
## AT5G45950 3.398444e-118 1.405960e-115
## AT2G42540 6.849027e-145 4.213908e-142
## AT5G43840  2.902880e-13  3.921994e-12
## AT5G40790  9.645811e-13  1.237707e-11
## AT5G06760 6.171087e-237 8.710308e-234
```

Max log2 fold change is equal to 8.6. This corresponds to a fold change of 2^8.6 = 388. Therefore the AT3G23635 gene is 388 more expressed in auxin-treated seedlings than in control conditions.

Select genes positively and differentially regulated for downstream analysis (over-representation analysis for instance.)

```
results(dds, contrast = c("condition", "ABA", "control")) %>%
  as.data.frame() %>%
  filter(padj < 0.01) %>%
  filter(log2FoldChange > 0) %>%
  write.csv(file = "diff_genes_ABA_deforges.csv")
```

Exercise 3: volcano plot

Shrinkage

Q12 (0.5 point): Shrink the log2 fold changes in order to shrink high log2 fold changes from lowly expressed genes. Use the related DESeq2 function that we have seen in the tutorial: https://scienceparkstudygroup.github.io/rna-seq-lesson/06-differential-analysis/index.html#3-volcano-plot
- First, extract the results completely with "results(dds, . . .)" - Then shrink the log2 fold changes with the "lfcShrink()" function and the "normal" shrinkage estimator. Call this new object res_shrink.

```
all_gene_results <- results(dds, contrast = c("condition", "ABA", "control"))

resLFC <- lfcShrink(dds = dds,
                    res = all_gene_results,
                    type = "normal",
                    coef = 2)
```

Volcano plot

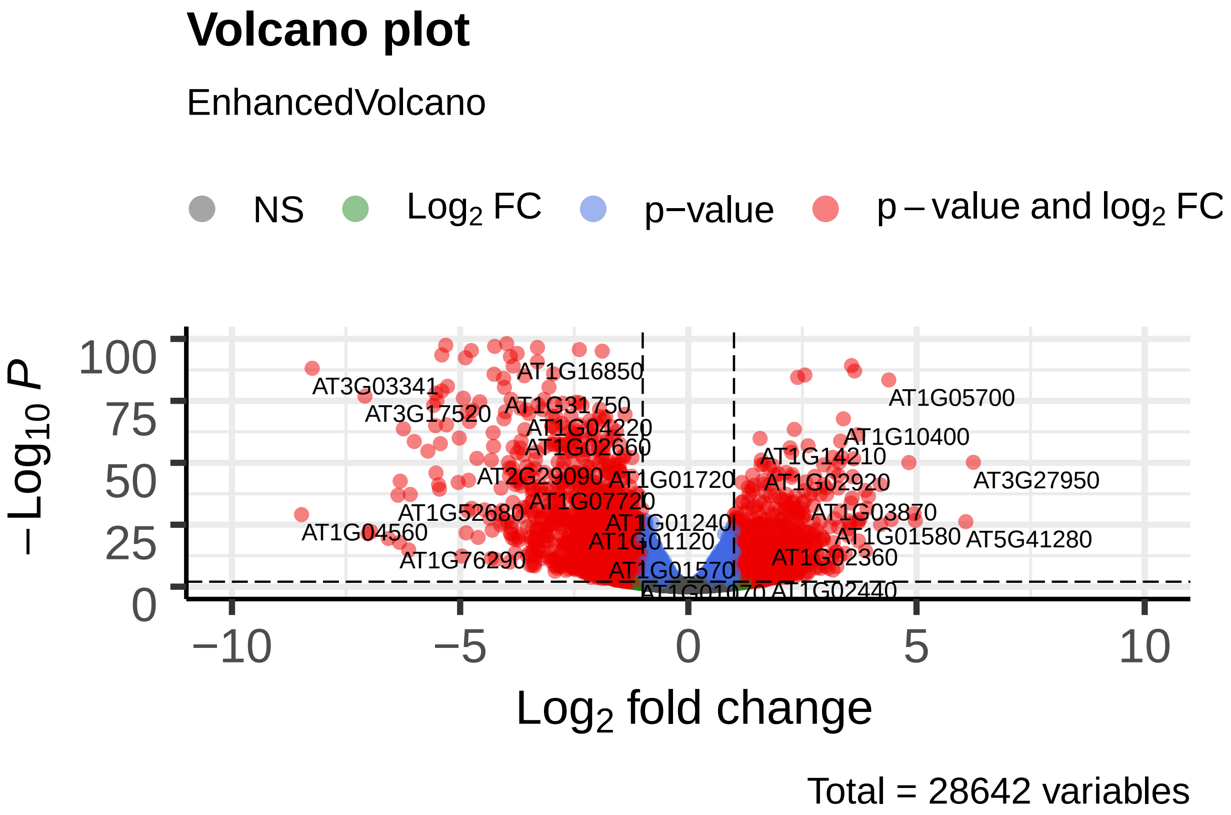
Q13 (1 point): make a volcano plot, a type of scatterplot that shows, for each gene, the magnitude of the shrinked log2 fold change (x-axis) versus statistical significance (adjusted p-value) as seen in the tutorial. Use the "res_shrink" object that you have built in the previous task. Hint: make sure you change the default "FCcutoff" and "pCutoff" values so that they better reflect your min/max log2 fold changes and adjusted p-values. Do not keep the default values for these two parameters.

What are the values to be set for the limits of x and y?

```
min(resLFC$log2FoldChange, na.rm = T)
## [1] -9.461703
```

```
max(resLFC$log2FoldChange, na.rm = T)
## [1] 6.747823
```

```
library("EnhancedVolcano")
EnhancedVolcano(toptable = resLFC,
                x = "log2FoldChange",
                y = "padj",
                lab = rownames(resLFC),
                xlim = c(-10, +10),
                ylim = c(0,100),
                pCutoff = 0.01,
                transcriptPointSize = 2.0,
                FCcutoff = 1,
                title = "Volcano plot",
                legend=c(
                  'Not significant',
                  'Log2 fold-change (but do not pass p-value cutoff)',
                  'Pass p-value cutoff',
                  'Pass both p-value & Log2 fold change')) +
  guides(legend = NULL)
```



Q14: - Most up-regulated genes: top right of the volcano plot.
- Most down-regulated genes: top left of the volcano plot.
- Most statistically significant genes are the highest on the y-axis.

Exercise 4: find annotation of the 5 selected candidate genes

Q15 (1 point): for each of your 5 selected genes, make a plot showing their expression in control and hormone-treated seedlings.

Select based on:
1. Highest log2 fold change, then
2. Highest baseMean

```
diff_genes <- results(dds, contrast = c("condition", "ABA", "control"))
diff_genes %>%
  as.data.frame() %>%
  filter(log2FoldChange > 0) %>%
  arrange(desc(log2FoldChange)) %>%
  head(n = 20) %>%
  arrange(desc(baseMean))
##              baseMean log2FoldChange     lfcSE      stat        pvalue
## AT3G02480 52069.14644       9.532293 0.2088601 45.639609  0.000000e+00
## AT1G52690 37053.70206       9.915419 0.1956953 50.667634  0.000000e+00
## AT5G06760 23304.52862       8.743607 0.2660167 32.868634 6.171087e-237
## AT3G17520  8951.16589      11.049334 0.5834087 18.939270  5.414248e-80
## AT5G52300  6377.64985      10.104950 0.3052127 33.107896 2.287693e-240
## AT3G15670  4859.21357       9.279350 0.3018093 30.745739 1.394127e-207
## AT2G33380  3963.24362      10.695898 0.4578457 23.361358 1.056538e-120
## AT2G42540  2809.94955       9.207797 0.3592415 25.631216 6.849027e-145
## AT3G03341  2207.12938      10.345668 0.5105503 20.263759  2.686481e-91
## AT5G50360  1866.35112       9.375026 0.3497922 26.801706 3.086634e-158
## AT5G45950  1435.79619       9.237277 0.3996510 23.113358 3.398444e-118
## AT1G04560  1406.28788      12.291266 1.0533910 11.668285  1.851179e-31
## AT1G27461   450.50451      12.490991 1.2250299 10.196478  2.055923e-24
## AT4G31830   319.18754      11.993477 1.1994373  9.999253  1.535507e-23
## AT5G59330   250.22952      10.680743 1.4078212  7.586718  3.281089e-14
## AT1G76290   233.70093       9.701184 1.0597912  9.153864  5.493293e-20
## AT5G62800   223.29870      11.478503 1.2022971  9.547144  1.333211e-21
## AT3G51810   152.25783       9.964051 1.1972125  8.322709  8.597519e-17
## AT5G43840    43.90151       9.129367 1.2507993  7.298827  2.902880e-13
## AT5G40790    36.87936       8.879640 1.2444369  7.135468  9.645811e-13
##                    padj
## AT3G02480  0.000000e+00
## AT1G52690  0.000000e+00
## AT5G06760 8.710308e-234
## AT3G17520  1.261310e-77
## AT5G52300 3.920942e-237
## AT3G15670 1.592957e-204
## AT2G33380 4.527077e-118
## AT2G42540 4.213908e-142
## AT3G03341  7.242933e-89
## AT5G50360 2.116108e-155
## AT5G45950 1.405960e-115
## AT1G04560  8.241010e-30
## AT1G27461  6.252456e-23
## AT4G31830  4.487759e-22
## AT5G59330  4.827084e-13
## AT1G76290  1.271086e-18
## AT5G62800  3.477218e-20
## AT3G51810  1.588125e-15
## AT5G43840  3.921994e-12
## AT5G40790  1.237707e-11
```

AT2G23170 = GH3.3
AT4G37390 = AUR3 Auxin upregulated 3 GH3-2
AT3G58190 = SYMMETRIC LEAVES 2-LIKE 16
AT3G10870 = ARABIDOPSIS THALIANA METHYL ESTERASE 17
AT2G45420 = LOB DOMAIN-CONTAINING PROTEIN 18

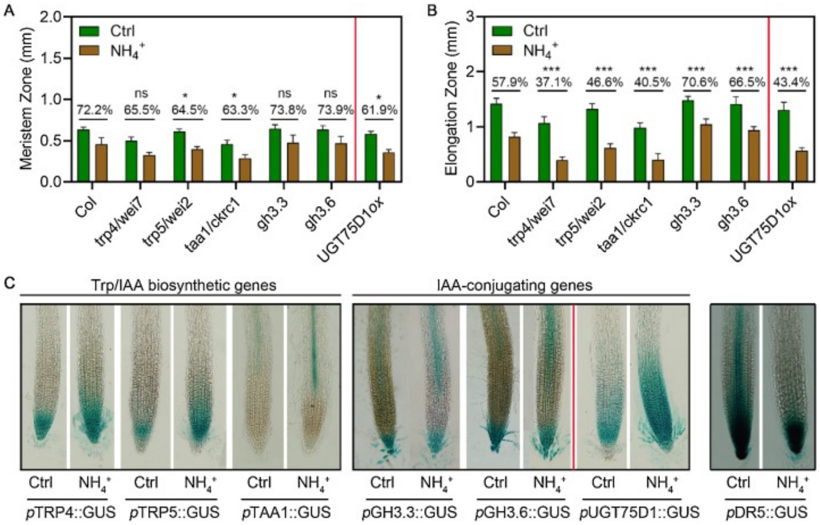
AT2G23170

Names: Auxin-responsive GRETCHEN HAGEN3.3 (GH3.3) protein
Description: conjugates amino acids to auxin and regulates its homeostasis.

> “The production of amide-linked IAA-amino acid conjugates is catalysed by Group II GRETCHEN HAGEN3 (GH3) acyl amido synthetases”

Papers: - Paper 1: Di DW, Li G, Sun L, Wu J, Wang M, Kronzucker HJ, Fang S, Chu J, Shi W. High ammonium inhibits root growth in Arabidopsis thaliana by promoting auxin conjugation rather than inhibiting auxin biosynthesis. J Plant Physiol. 2021 Apr 18;261:153415. doi: 10.1016/j.jplph.2021.153415. Epub ahead of print. PMID: 33894579. - Paper 2: Staswick PE, Serban B, Rowe M, Tiryaki I, Maldonado MT, Maldonado MC, Suza W. Characterization of an Arabidopsis enzyme family that conjugates amino acids to indole-3-acetic acid. Plant Cell. 2005 Feb;17(2):616-27. doi: 10.1105/tpc.104.026690. Epub 2005 Jan 19. PMID: 15659623; PMCID: PMC548830.

Images



This figure shows that ammonium (NH4+) induces the expression of GH3.3 in the elongation zone. This in turn reduces the amount of biologically active auxin.

AT3G58190

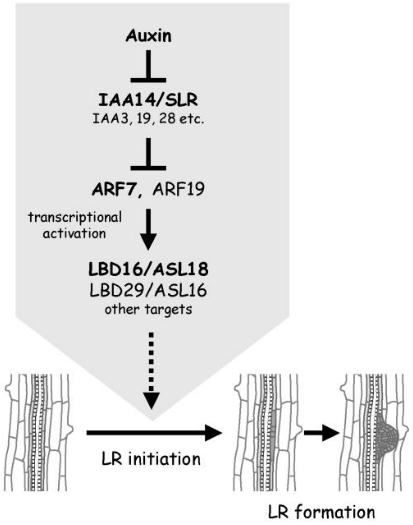
Names: * ASYMMETRIC LEAVES 2-LIKE 16 * LATERAL ORGAN BOUNDARIES DOMAIN 29 (LBD29)

Papers: - Paper 1: Zhang F, Tao W, Sun R, Wang J, Li C, et al. (2020) PRH1 mediates ARF7-LBD dependent auxin signaling to regulate lateral root development in Arabidopsis thaliana. PLOS Genetics 16(2): e1008044. https://doi.org/10.1371/journal.pgen.1008044. - Paper 2: Okushima Y, Fukaki H, Onoda M, Theologis A, Tasaka M. ARF7 and ARF19 regulate lateral root formation via direct activation of LBD/ASL genes in Arabidopsis. Plant Cell. 2007;19(1):118-130. doi:10.1105/tpc.106.047761. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1820965/

Quotes:

> “The development of lateral roots in Arabidopsis thaliana is strongly dependent on signaling directed by the AUXIN RESPONSE FACTOR7 (ARF7), which in turn activates LATERAL ORGAN BOUNDARIES DOMAIN (LBD) transcription factors (LBD16, LBD18 and LBD29)”

Images


Auxin
IAA14/SLR
IAA3, 19, 28 etc.
ARF7, ARF19
transcriptional activation
LBD16/ASL18
LBD29/ASL16
other targets
LR initiation
LR formation