

Gene Expression - Karl 1

Taryn H

2023-03-01

#Inserting Code

```
data<-read_csv(here::here("Cleaned Data", "Gene_Expression.csv"))
```

```
## Rows: 88 Columns: 6
## -- Column specification -----
## Delimiter: ","
## chr (4): Cell_Line, Treatment, Cell_Line_Treatment, Gene_Line
## dbl (2): Concentration, Gene_Expression
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
data
```

```
## # A tibble: 88 x 6
##   Cell_Line Concentration Treatment Gene_Expression `Cell_Line_Treatment`
##   <chr>          <dbl> <chr>          <dbl> <chr>
## 1 Wild Type      0 Placebo      5.51 Wild TypePlacebo
## 2 Wild Type      1 Placebo      6.41 Wild TypePlacebo
## 3 Wild Type      2 Placebo      5.71 Wild TypePlacebo
## 4 Wild Type      3 Placebo      7.94 Wild TypePlacebo
## 5 Wild Type      4 Placebo      6.87 Wild TypePlacebo
## 6 Wild Type      5 Placebo      7.29 Wild TypePlacebo
## 7 Wild Type      6 Placebo     10.0 Wild TypePlacebo
## 8 Wild Type      7 Placebo      8.85 Wild TypePlacebo
## 9 Wild Type      8 Placebo      8.91 Wild TypePlacebo
## 10 Wild Type     9 Placebo      9.68 Wild TypePlacebo
## # i 78 more rows
## # i 1 more variable: `Gene_Line` <chr>
```

#Clean

```
skimr::skim_without_charts(data)
```

Table 1: Data summary

Name	data
Number of rows	88
Number of columns	6
Column type frequency:	
character	4
numeric	2

Table 1: Data summary

Group variables	None
-----------------	------

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Cell_Line	0	1	9	13	0	2	0
Treatment	0	1	7	20	0	2	0
Cell_Line_Treatment	0	1	16	33	0	4	0
Gene_Line	0	1	6	6	0	8	0

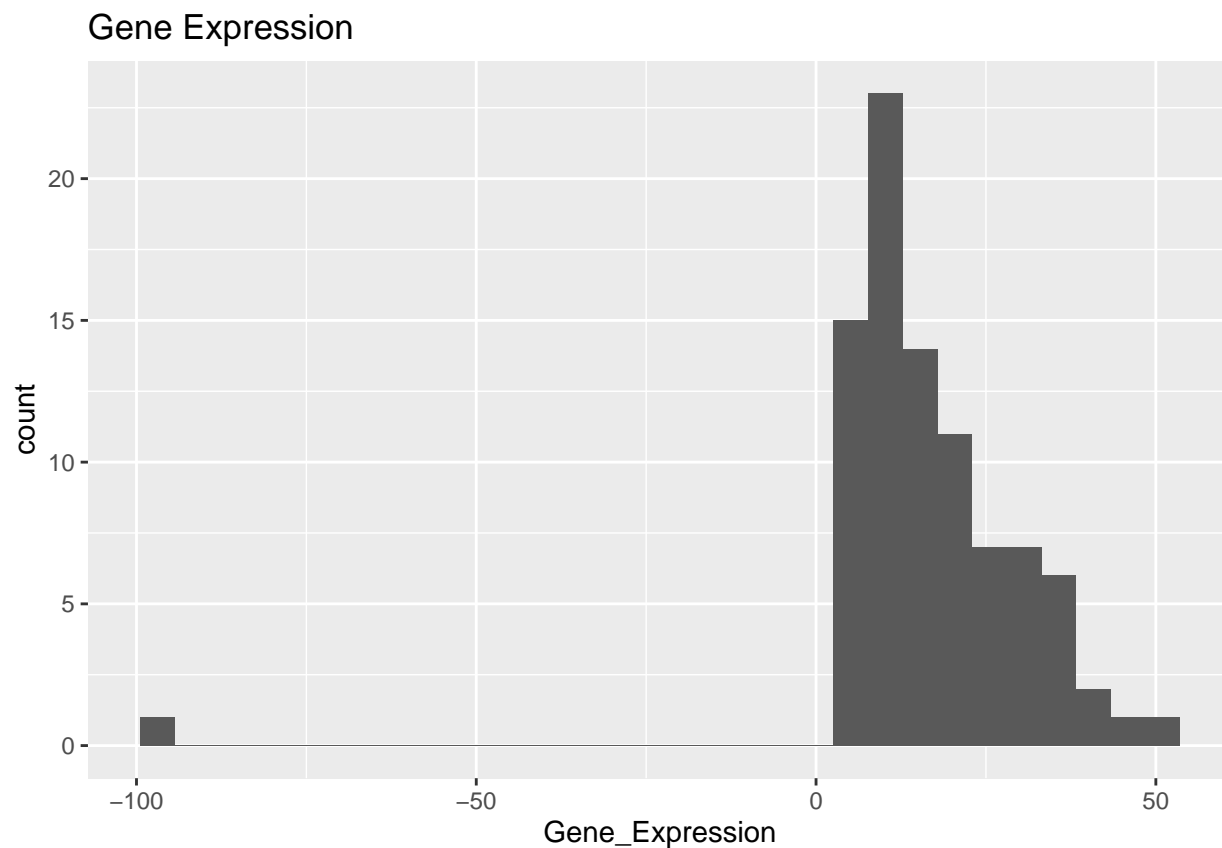
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Concentration	0	1	5.00	3.18	0	2.00	5.00	8.00	10.00
Gene_Expression	0	1	16.14	16.45	-99	8.96	14.82	24.28	48.96

```
#Investigate gene expression
```

```
data |> ggplot(aes(x=Gene_Expression)) + geom_histogram() + ggtitle("Gene Expression")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



#We can see there is a value of -99 which has been entered incorrectly - we will set this to NA

```
data <- data |> filter(Gene_Expression>0)
```

```
five<- fivenum(data$Gene_Expression)
```

```
data |> ggplot(aes(x=Gene_Expression)) +  
  geom_histogram(fill="mediumaquamarine", col="black") +  
  ggtitle("Gene Expression") +  
  geom_vline(aes(xintercept=mean(Gene_Expression)),col="darkblue",linetype="dashed",size=1) +  
  scale_color_brewer(palette="Dark2") +  
  geom_vline(aes(xintercept=five[1]),col="darkorange3",linetype="dashed",size=1) +  
  geom_vline(aes(xintercept=five[2]),col="darkorange3",linetype="dashed",size=1) +  
  geom_vline(aes(xintercept=five[3]),col="darkorange3",linetype="dashed",size=1) +  
  geom_vline(aes(xintercept=five[4]),col="darkorange3",linetype="dashed",size=1) +  
  geom_vline(aes(xintercept=five[5]),col="darkorange3",linetype="dashed",size=1)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
```

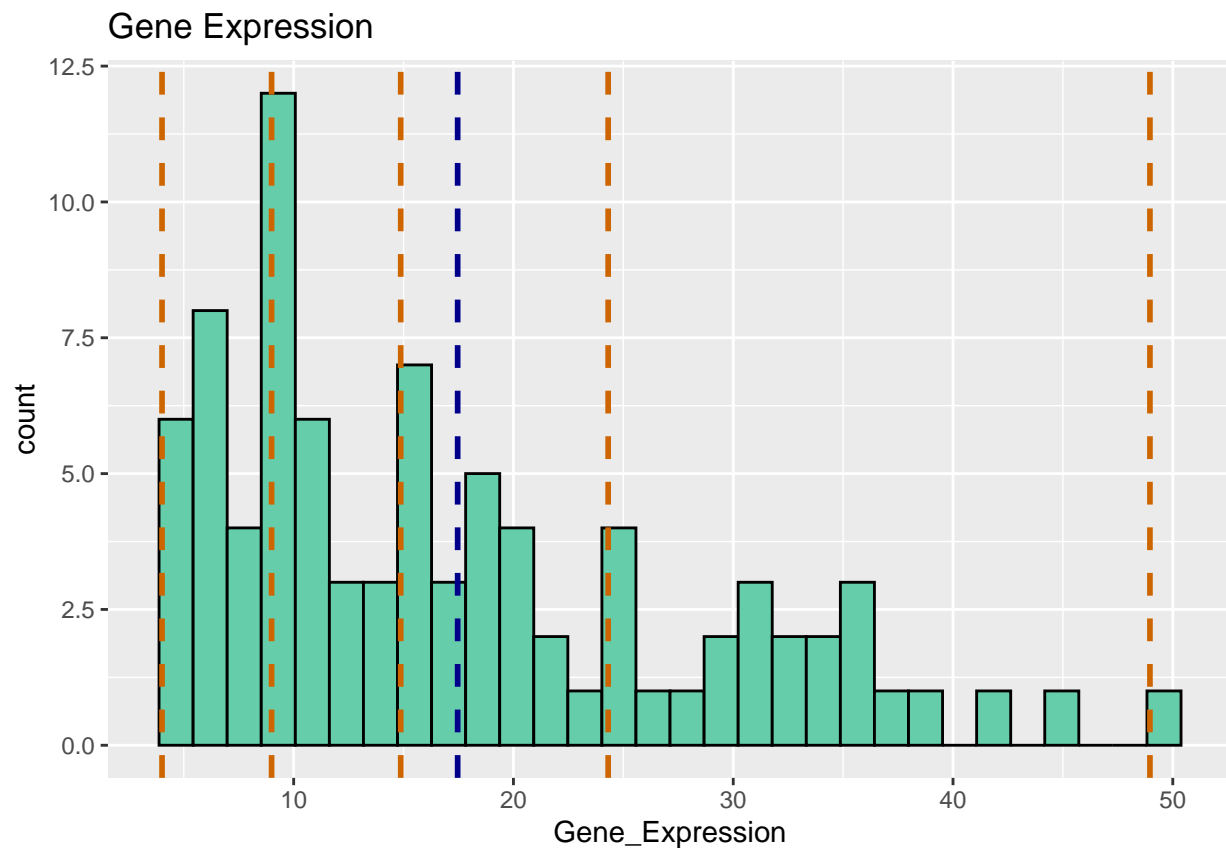
```
## i Please use `linewidth` instead.
```

```
## This warning is displayed once every 8 hours.
```

```
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
```

```
## generated.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



#Investigate cell line

```

class(data$Cell_Line)

## [1] "character"
table(data$Cell_Line)

##
## Cell Type 101      Wild Type
##           44         43
#Investigate Concentration
class(data$Concentration)

## [1] "numeric"
table(data$Concentration)

##
##  0  1  2  3  4  5  6  7  8  9 10
##  8  8  8  8  8  7  8  8  8  8  8
#Investigate Treatment
class(data$Treatment)

## [1] "character"
table(data$Treatment)

##
## Activating Factor 42      Placebo
##           43             44
#Gene Expression box plot + points
data |> ggplot(aes(x=Gene_Expression, fill=Treatment)) +
  geom_dotplot(dotsize=1.2) +
  scale_fill_manual(values=c("Placebo" = "deepskyblue1",
                             "Activating Factor 42"="magenta1")) +
  ggtitle("Dot Plot of Gene Expression coloured by Treatment")

## Bin width defaults to 1/30 of the range of the data. Pick better value with
## `binwidth`.

```



```
data |> ggplot(aes(x=Gene_Expression, fill=Cell_Line)) +
  geom_dotplot(dotsize=1.2) +
  scale_fill_manual(values=c("Wild Type" = "springgreen3",
                             "Cell Type 101"="darkorchid2")) +
  ggtitle("Dot Plot of Gene Expression coloured by Cell Line")
```

```
## Bin width defaults to 1/30 of the range of the data. Pick better value with
## `binwidth`.
```



#Plots

```
sum <- data |> group_by(Cell_Line) |> summarize(min=fivenum(Gene_Expression)[1],
                                                Q1=fivenum(Gene_Expression)[2],
                                                Med=fivenum(Gene_Expression)[3],
                                                Q3=fivenum(Gene_Expression)[4],
                                                max=fivenum(Gene_Expression)[5])
```

sum

A tibble: 2 x 6

Cell_Line	min	Q1	Med	Q3	max
Cell Type 101	4.01	11.6	17.0	24.4	49.0
Wild Type	4.3	7.62	10.0	24.0	41.7

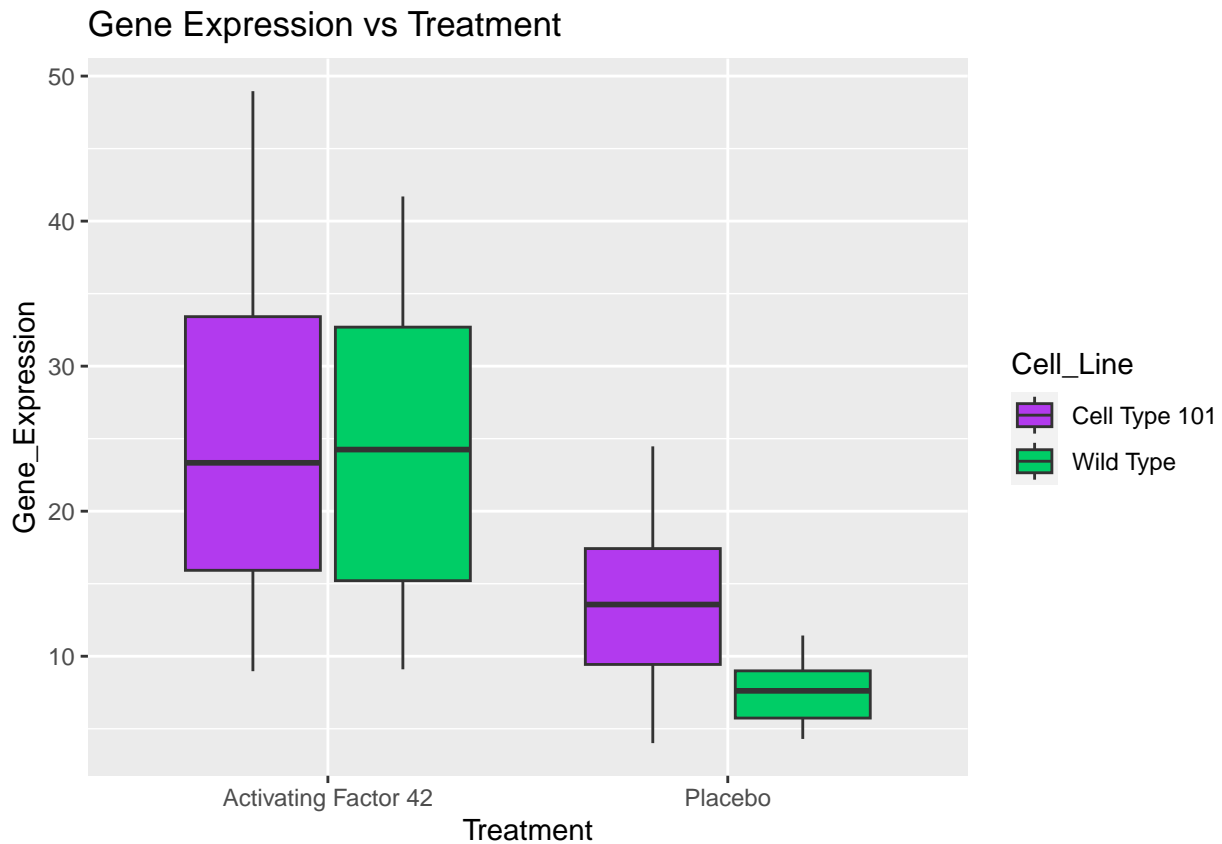
```
sum2 <- data |> group_by(Treatment) |> summarize(min=fivenum(Gene_Expression)[1],
                                                  Q1=fivenum(Gene_Expression)[2],
                                                  Med=fivenum(Gene_Expression)[3],
                                                  Q3=fivenum(Gene_Expression)[4],
                                                  max=fivenum(Gene_Expression)[5])
```

sum2

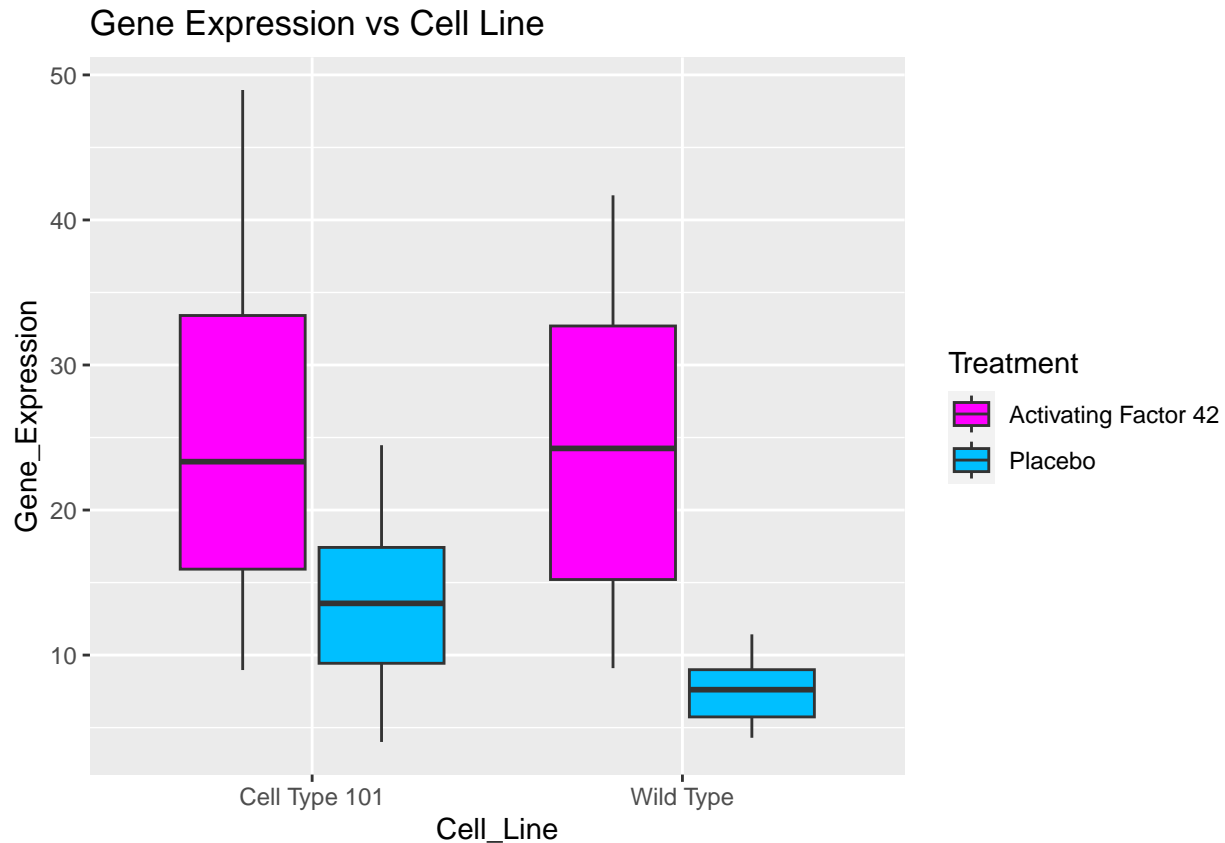
A tibble: 2 x 6

Treatment	min	Q1	Med	Q3	max
Activating Factor 42	8.97	15.3	24.2	32.7	49.0
Placebo	4.01	6.14	9.15	13.6	24.5

```
data |> ggplot(aes(x=Treatment, y=Gene_Expression, fill=Cell_Line)) +
  geom_boxplot() +
  scale_fill_manual(values=c("Wild Type" = "springgreen3",
                             "Cell Type 101"="darkorchid2")) + ggtitle("Gene Expression vs Treatment")
```



```
data |> ggplot(aes(x=Cell_Line, y=Gene_Expression, fill=Treatment)) +
  geom_boxplot() +
  scale_fill_manual(values=c("Placebo" = "deepskyblue1",
                             "Activating Factor 42"="magenta1")) +
  ggtitle("Gene Expression vs Cell Line")
```



```
sum3 <- data |> group_by(Treatment, Cell_Line) |> summarize(min=fivenum(Gene_Expression)[1],
                                                           Q1=fivenum(Gene_Expression)[2],
                                                           Med=fivenum(Gene_Expression)[3],
                                                           Q3=fivenum(Gene_Expression)[4],
                                                           max=fivenum(Gene_Expression)[5])
```

`summarise()` has grouped output by 'Treatment'. You can override using the
`.groups` argument.

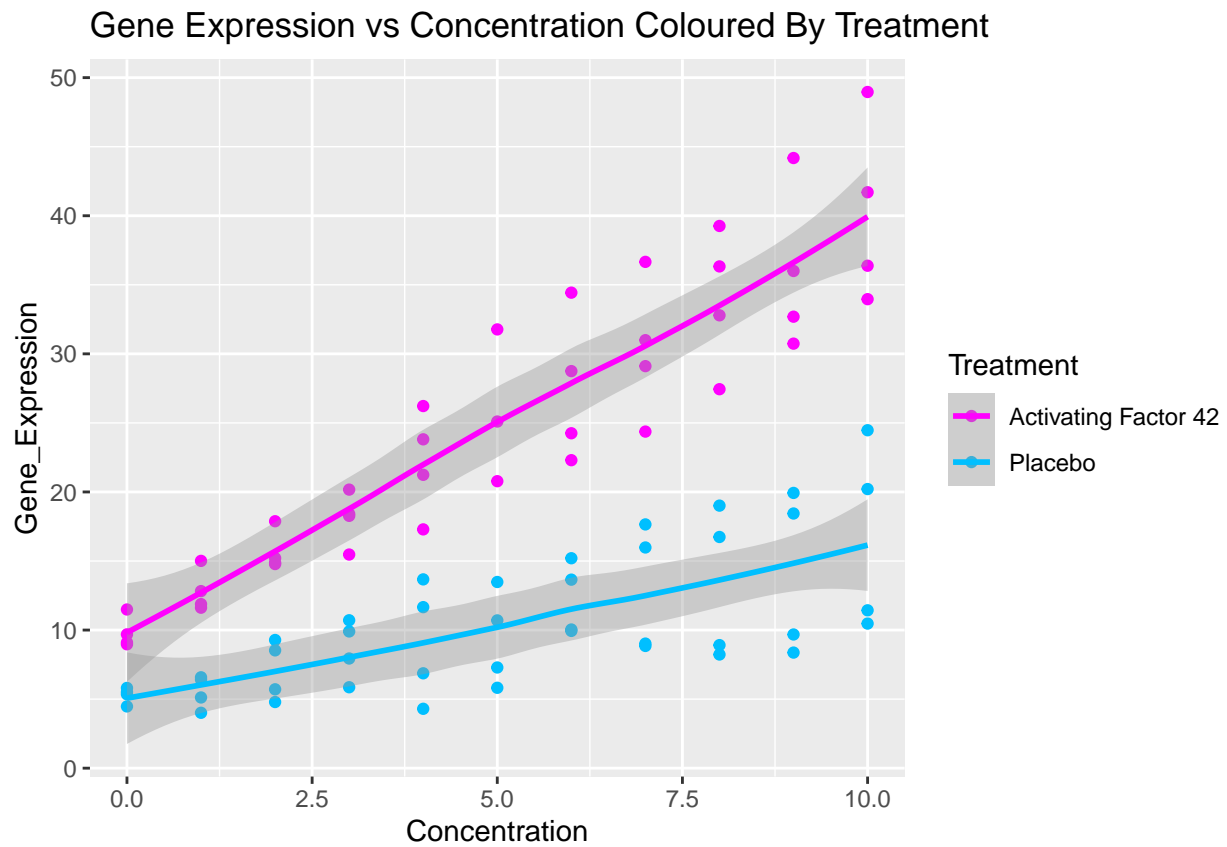
```
sum3
```

```
## # A tibble: 4 x 7
## # Groups:   Treatment [2]
##   Treatment      Cell_Line      min    Q1  Med   Q3   max
##   <chr>          <chr>    <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 Activating Factor 42 Cell Type 101  8.97 15.5 23.3 34.0 49.0
## 2 Activating Factor 42 Wild Type      9.1 15.2 24.2 32.7 41.7
## 3 Placebo        Cell Type 101  4.01  9.28 13.6 17.6 24.5
## 4 Placebo        Wild Type      4.3  5.71  7.62  9.02 11.4
```

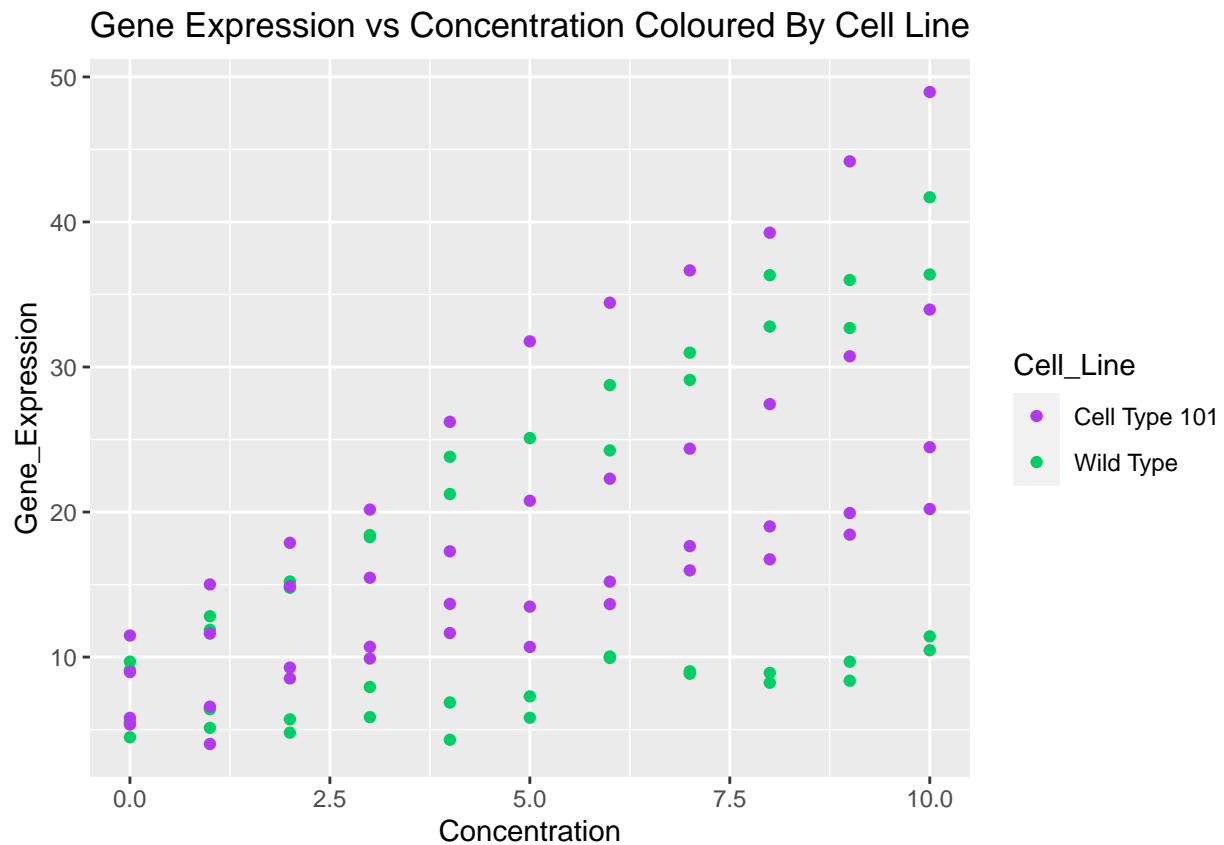
```
data |> ggplot(aes(x=Concentration, y=Gene_Expression, col=Treatment)) +
  geom_point(line="black") + geom_smooth() +
  scale_colour_manual(values=c("Placebo" = "deepskyblue1",
                              "Activating Factor 42"="magenta1")) +
  ggtitle("Gene Expression vs Concentration Coloured By Treatment")
```

Warning in geom_point(line = "black"): Ignoring unknown parameters: `line`

`geom_smooth()` using method = 'loess' and formula = 'y ~ x'



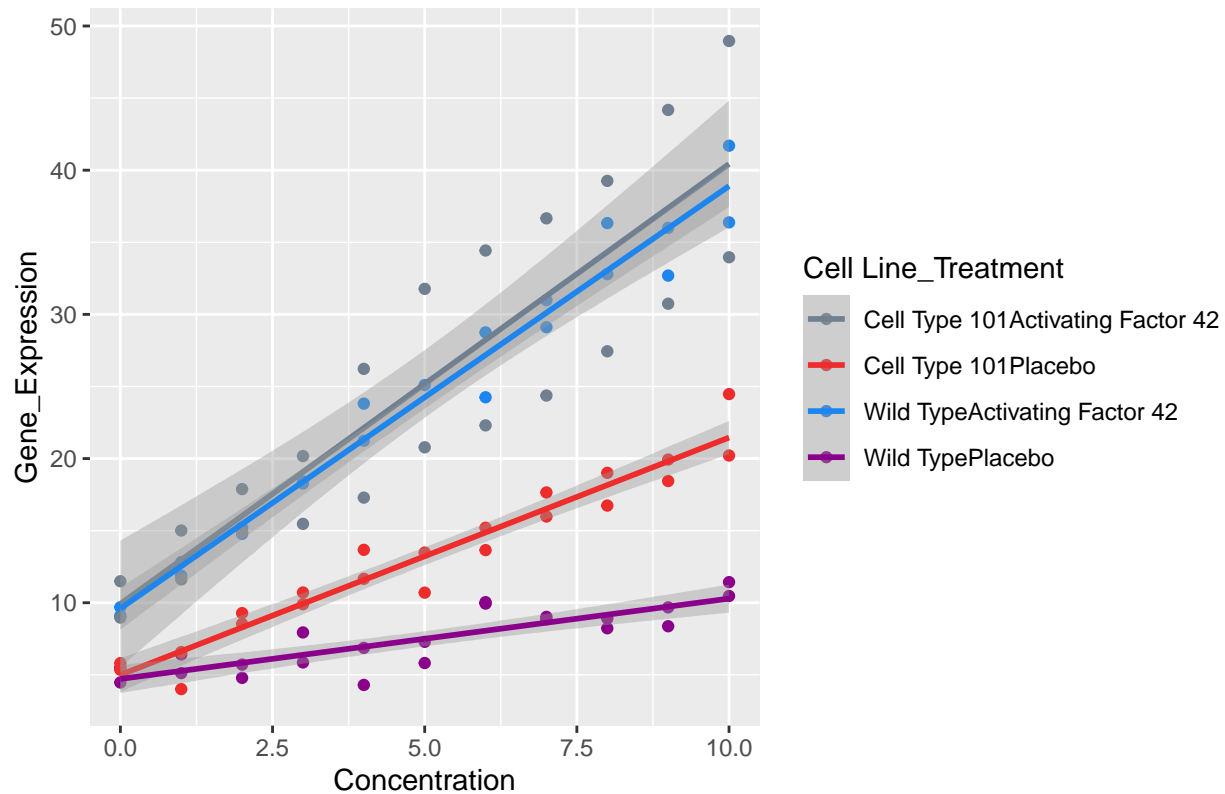
```
data |> ggplot(aes(x=Concentration, y=Gene_Expression, col=Cell_Line)) +
  geom_point() +
  scale_colour_manual(values=c("Wild Type" = "springgreen3",
                              "Cell Type 101"="darkorchid2")) +ggtitle("Gene Expression vs Concentration")
```



```
data |> ggplot(aes(x=Concentration, y=Gene_Expression, col=`Cell Line_Treatment`)) +
  geom_point() + geom_smooth(method=lm) +
  ggtitle("Gene Expression vs Concentration Coloured By Cell Line Concatenated with Treatment") +
  scale_colour_manual(values=c("Wild TypeActivating Factor 42"="dodgerblue2",
                                "Cell Type 101Activating Factor 42"="slategray",
                                "Wild TypePlacebo"="darkmagenta",
                                "Cell Type 101Placebo"="firebrick2"))

## `geom_smooth()` using formula = 'y ~ x'
```

Gene Expression vs Concentration Coloured By Cell Line Concatenated wit



#Make a table

```
Cell1<-data |>
  group_by(Cell_Line) |>
  summarise(Mean=mean(Gene_Expression), SD=sd(Gene_Expression))

gt_data1<-
  gt(Cell1) |>
  tab_header(title="Table 1: Mean of Gene Expression for each Cell Line")
gt_data1 |>
  cols_label(Cell_Line="Cell Line",Mean="Mean", SD="Standard Deviation")
```

Table 1: Mean of Gene Expression for each Cell Line

Cell Line	Mean	Standard Deviation
Cell Type 101	19.20000	10.57852
Wild Type	15.68209	10.97189

#Make a table 2

```
Treat1<-data |>
  group_by(Treatment) |>
  summarise(Mean=mean(Gene_Expression), SD=sd(Gene_Expression))

gt_data2<-
  gt(Treat1) |>
  tab_header(title="Table 2: Mean of Gene Expression for each Treatment")
```

```
gt_data2 |>
  cols_label(Mean="Mean", SD="Standard Deviation")
```

Table 2: Mean of Gene Expression for each Treatment

Treatment	Mean	Standard Deviation
Activating Factor 42	24.72419	10.421329
Placebo	10.36341	5.036591

#Make a table 3

```
five_num<-data |>
  group_by(Treatment) |>
  summarise(fivenum=fivenum(Gene_Expression))
```

```
## Warning: Returning more (or less) than 1 row per `summarise()` group was deprecated in
## dplyr 1.1.0.
## i Please use `reframe()` instead.
## i When switching from `summarise()` to `reframe()`, remember that `reframe()`
## always returns an ungrouped data frame and adjust accordingly.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

## `summarise()` has grouped output by 'Treatment'. You can override using the
## `.groups` argument.
```

```
five_num
```

```
## # A tibble: 10 x 2
## # Groups:   Treatment [2]
##   Treatment      fivenum
##   <chr>         <dbl>
## 1 Activating Factor 42    8.97
## 2 Activating Factor 42   15.3
## 3 Activating Factor 42   24.2
## 4 Activating Factor 42   32.7
## 5 Activating Factor 42   49.0
## 6 Placebo           4.01
## 7 Placebo           6.14
## 8 Placebo           9.15
## 9 Placebo          13.6
## 10 Placebo          24.5
```