IMRaD Report: Gene expression analysis

Yingxin Xu

2024-05-29

1. Introduction

Gene expression is a critical process in cellular function, regulating a wide range of biological activities and responses.

This experiment investigates the effect of a new treatment on gene expression, with saline serving as a placebo for comparison. The study is conducted on two distinct cell lines to determine if the response to the treatment is consistent across different cellular environments. Additionally, the impact of varying concentrations of a growth factor on gene expression is examined to understand the interaction between the treatment and the growth factor.

Variables:

1. `cell_line`: The two cell line types used for the experiment, Wild-type and Cell-type 101.
2. `Treatment`: This variable describes whether to add new treatment. Added: Activating factor 42; Not added: Placebo (saline added instead).
3. `name`: Names of different cell lines. There are two cell lines for each identical cell line type and the same treatment. There were 8 cell lines in total.
4. `conc`: different concentration of a growth factor, from 0 to 10.
5. `gene_expression`: Gene expression level.

2. Methods

The analysis is conducted using R-4.3.3 on an Intel® Core™ i5-1035G1 CPU @ 1.00GHz (8 CPUs) with 16GB RAM and a 64-bit Windows 10 operating system. All packages used in this analysis can be found in the appendix. If you need to cite any of them in your paper, please use the "`citation(package name)`" function in RStudio to obtain the corresponding reference information for the R package.

2.1. Load data

2.2. Data cleaning

1. Standardize the case of words, changing all to title case. For example, "placebo" should be standardized to "Placebo", "activating factor 42" to "Activating Factor 42", "CELL-TYPE 101" to "Cell-type 101", and "WILD-TYPE" to "Wild-type".
2. Standardize all cell line names in the "name" variable.

2.3. EDA: look at the whole dataset

1. List the data structure details, including data types, data volume, number of variables, mean and median for numerical data, and number of categories for categorical data, etc.
2. Use a histogram to plot the data distribution of the response variable gene expression.
3. Use box plots and scatter plots to illustrate the relationship between each categorical variable and gene expression.

2.4. Fit models

The purpose of this research is to explore the impact of different treatments on gene expression under varying concentrations of growth factors. Therefore, when modeling, it is important to consider including interaction terms between treatment and concentration `(Treatment * conc)`. Based on this, two models were attempted, a linear model and a linear mixed-effect model.

Linear model

Using a basic linear model, all independent variables are treated as predictors, with the addition of interaction terms between `Treatment` and `conc`. `cell_line` and `name` are included as separate predictors. To maintain research integrity, models with and without added interaction term were both fitted.

Linear mixed-effect model

Including `Treatment`, `conc`, and `cell_line` as predictors, with the addition of interaction terms between `Treatment` and `conc`. `name` is included as a random variable in the model. This is because there are differences between individuals of different cell lines, and these differences are random (similar to the differences between individuals in a human population). To maintain research integrity, models with and without added interaction term were both fitted.

2.5. Model evaluation

Basic methods were employed to compare two linear models with two linear mixed-effects models.

1. ANOVA: Compare models of same type, to decide whether to maintain the interaction term or not.
2. AIC (Akaike Information Criterion), BIC (Bayesian Information Criterion) values: Compare different types of model. Generally, a lower AIC or BIC value indicates a better model.
3. Diagnostic plots

3. Results

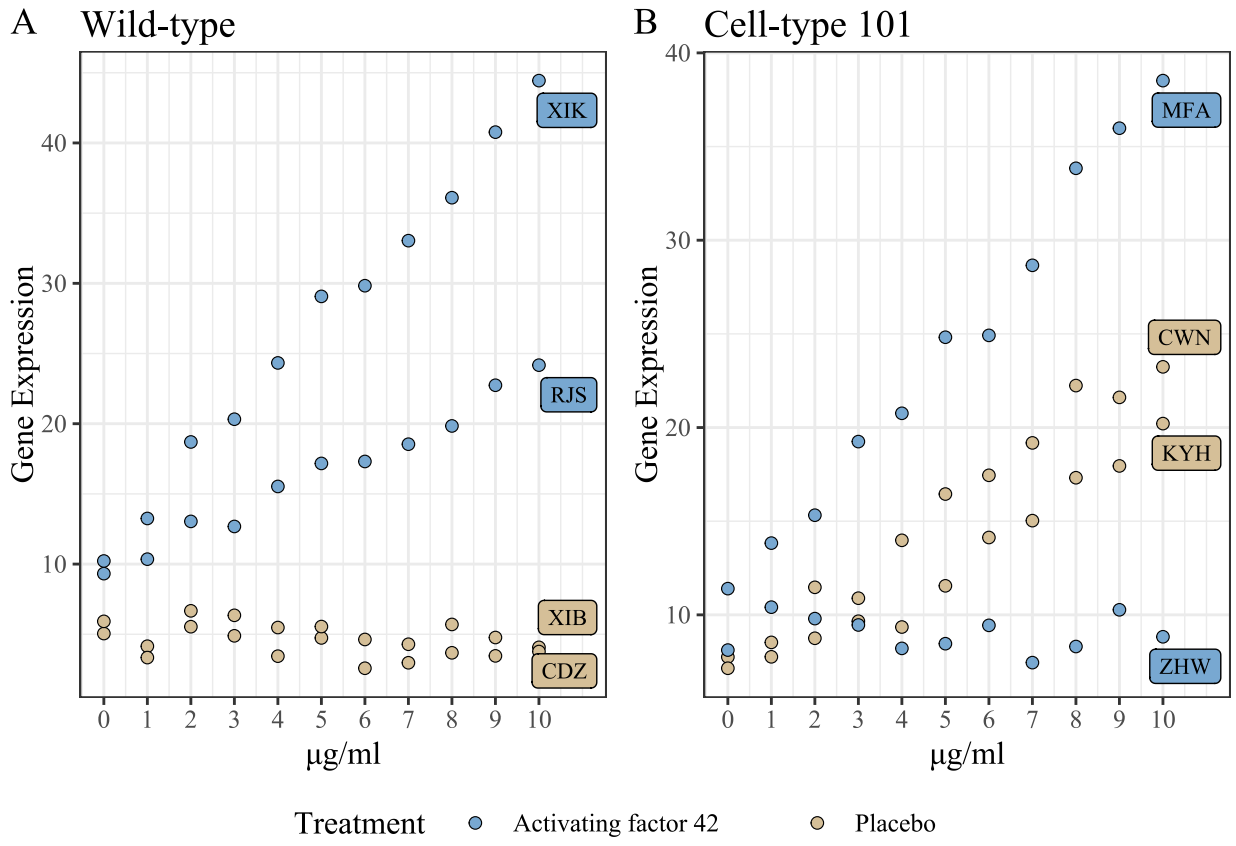
3.1. EDA results

1. Data structure

Taking an overview of the data, there are 3 categorical variables: `cell_line`, `Treatment`, and `name`; and 2 numerical variables: `conc` and `gene_expression`.

2. Gene expression vs. growth factor plot (grouped by cell line type)

Plot the relationship between growth factor concentration and gene expression, categorized by cell line type. Different colors represent different treatments, and individual cell line instances are labeled in the graph.



3.2. Model fitting results

1. Linear models with (`gene_lm2`) and without (`gene_lm1`) interaction term `Treatment:conc`.

   gene_lm1: gene_expression ~ Treatment + conc + cell_line + name

   gene_lm2: gene_expression ~ Treatment * conc + cell_line + name

2. Linear mixed-effect models with(`gene_lmm2`) and without (`gene_lmm1`) interaction term `Treatment:conc`. name serves as a random factor.

   gene_lmm1: gene_expression ~ Treatment + conc + cell_line + (1|name)

   gene_lmm2: gene_expression ~ Treatment * conc + cell_line + (1|name)

3.3. Model evaluation

1. ANOVA

- Linear models

```
## Analysis of Variance Table
##
## Model 1: gene_expression ~ Treatment + conc + cell_line + name
## Model 2: gene_expression ~ Treatment * conc + cell_line + name
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     79 1523.0
## 2     78 1213.4  1    309.56 19.899 2.708e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p-value is smaller than 0.05, indicating that adding the interaction term signaficantly improved the model.

- Linear mixed-effect models

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: gene_fl
## Models:
## gene_lmm1: gene_expression ~ Treatment + conc + cell_line + (1 | name)
## gene_lmm2: gene_expression ~ Treatment * conc + cell_line + (1 | name)
##           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## gene_lmm1    6 545.09 559.96 -266.55   533.09
## gene_lmm2    7 528.91 546.26 -257.46   514.91 18.178  1  2.012e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p-value is smaller than 0.05, indicating that adding the interaction term signaficantly improved the model.

2. AIC and BIC values

The ANOVA results indicate that including the interaction term is necessary, so here we only compare the AIC and BIC values of the two models with the interaction term added (`gene_lm2` and `gene_lmm2`).

```
## AIC of gene_lm2: 502.6321
##  AIC of gene_lmm2: 518.5759
```

```
## BIC of gene_lm2: 529.8829
##  BIC of gene_lmm2: 535.9173
```

3. Diagnostic results

The regression diagnostic plots for `gene_lm2` and `gene_lmm2` were generated. Due to the similarity between the diagnostic plots for both models, they are not listed here. For viewing, please refer to the code in the appendix.

4. Discussion

4.1. Gene expression vs. growth factor plot (grouped by cell line type)

From the gene expression plot, we can observe that the new treatment indeed has a enhancing effect on the gene expression of growth factors.

However, it is worth noting that even for the same cell line type and the same treatment, there are differences in gene expression between two different cell line instances. For example, while "XIK" and "RJS" are both wild type cell lines under the action of Activating factor 42, the gene expression levels differ. The contrast is even more evident between "MFA" and "ZHW". Both are cell-type 101 cell lines under the action of Activating factor 42, but the gene expression in "ZHW" is even lower than that in the saline group.

We can speculate that the differences in response to the new treatment among cell line instances are due to variations between cell lines: some cell lines have increased gene expression with the new treatment, while others have decreased gene expression. This demonstrates the importance of including "name" as a random factor in the modeling process.

4.2. Model evaluation

- As previously mentioned, the ANOVA results for both the linear model and the linear mixed-effects model indicated that adding interaction terms significantly improves the models. Therefore, interaction terms were included in both models.
- Comparing linear model and linear mixed-effect model:

The AIC and BIC values of the models are:

Table 1: Comparison of AIC and BIC values for LM and LMM

Model	AIC	BIC
gene_lm2	502.6321	529.8829
gene_lmm2	518.5759	535.9173

The AIC and BIC values for `gene_lm2` are slightly lower than those for `gene_lmm2`. Since a model with a lower AIC or BIC value is considered better, this implies that, from the perspective of AIC and BIC, the linear model is slightly superior to the linear mixed-effects model. However, this is not an absolute advantage.

The difference between the two models is less than 20 for both AIC and BIC, indicating that there is no big advantage of one model over the other. Combined with the similar diagnostic plots, we can conclude that the two models are very similar.

Therefore, the difference between these two models lies in whether "name" is treated as a predictor or a random factor. Our goal is to fit a predictive model for gene expression, which may be applied to predict the gene expression of many other cell line instances (not present in this dataset). Thus, we should not treat "name" as a fixed predictor but rather as a random factor.

Therefore, the final decision is to use the linear mixed-effects model with interaction terms, i.e.

```
gene_lmm2: gene_expression ~ Treatment * conc + cell_line + (1|name).
```

Appendix

```
knitr::opts_chunk$set(echo = TRUE)
pacman::p_load(readxl, regclass, car, vip, tidymodels, lme4, showtext, gghighlight, ggpubr,
               patchwork, ggrepel, gglm, multilevelmod, knitr)
gene_path = paste(here::here(), "/raw-data/2024-03-04-WIF-tis4d.xlsx", sep = "")
gene_fl <- read_excel(gene_path, sheet = 1, na = "NA")
gene_fl <-
  gene_fl |>
  mutate(
    treatment = case_when(
      treatment == "placebo" ~ "Placebo",
      treatment == "activating factor 42" ~ "Activating factor 42",
      TRUE ~ treatment
    )
  )

gene_fl <-
  gene_fl |>
  mutate(
    cell_line = case_when(
      cell_line == "CELL-TYPE 101" ~ "Cell-type 101",
      cell_line == "WILD-TYPE" ~ "Wild-type",
      TRUE ~ cell_line
    )
  )

gene_fl <-
  gene_fl |>
  mutate(
    name = case_when(
      name == "GL-cDZ" ~ "GL-CDZ",
      name == "Gl-Cwn" ~ "GL-CWN",
      name == "GL-cwN" ~ "GL-CWN",
      name == "GL-kYH" ~ "GL-KYH",
      name == "Gl-Rjs" ~ "GL-RJS",
      name == "GL-rjS" ~ "GL-RJS",
      name == "Gl-Xib" ~ "GL-XIB",
      name == "GL-XIb" ~ "GL-XIB",
      name == "GL-Xik" ~ "GL-XIK",
      name == "Gl-Zhw" ~ "GL-ZHW",
      name == "GL-ZHw" ~ "GL-ZHW",
      TRUE ~ name
    )
  )

gene_fl <- na.omit(gene_fl)

colnames(gene_fl)[2] <- "Treatment"

# look at the whole dataset
gene_tab <- skimr::skim_without_charts(gene_fl)
## Add font
```

```r
font_add(family = "times",
         regular = here::here("font/Times New Roman.ttf"))
# points

set_geom_fonts <- function(family = NULL, ggrepel = FALSE) {
  if (is.null(family)) {
    family <- ggplot2::theme_get()$text$family
  }

  update_geom_defaults("text", list(family = family))
  update_geom_defaults("label", list(family = family))

  # Optional defaults for the ggrepel geoms
  if (ggrepel) {
    update_geom_defaults(ggrepel::GeomTextRepel, list(family = family))
    update_geom_defaults(ggrepel::GeomLabelRepel, list(family = family))
  }
}

showtext_auto()

gene_fl_A <- gene_fl[which(gene_fl$cell_line == "Wild-type"), ]
gene_fl_B <- gene_fl[which(gene_fl$cell_line == "Cell-type 101"), ]


label_data <- c(rep(NA, 20), "XIB", "CDZ", rep(NA, 20), "RJS", "XIK")

pA <- gene_fl_A |>
  ggplot(aes(x = conc, y = gene_expression, fill = Treatment)) +
  scale_fill_manual(values = c("#78A8D1", "#D5BF98")) +
  geom_point(colour = "black", size = 2, shape = 21, stroke = 0.3) +
  labs(x = expression(paste(mu, "g/ml")), y = "Gene Expression") +
  ggtitle("Wild-type") +
  geom_label_repel(label = label_data, nudge_x = 1, nudge_y = 0, size = 3, show.legend = FALSE) +
  scale_x_continuous(limits = c(0, 11), breaks = seq(0, 10, 1)) +
  theme_bw() +
  scale_shape_manual() +
  set_geom_fonts(family = "times", ggrepel = TRUE) +
  theme(text = element_text(family = "times", size = 12))


label_data <- c(rep(NA, 20), "CWN", "KYH", rep(NA, 20), "ZHW", "MFA")

pB <- gene_fl_B |>
  ggplot(aes(x = conc, y = gene_expression, fill = Treatment)) +
  scale_fill_manual(values = c("#78A8D1", "#D5BF98")) +
  geom_point(colour = "black", size = 2, shape = 21, stroke = 0.3) +
  labs(x = expression(paste(mu, "g/ml")), y = "Gene Expression") +
  ggtitle("Cell-type 101") +
  geom_label_repel(label = label_data, nudge_x = 1, nudge_y = 0, size = 3, show.legend = FALSE) +
  scale_x_continuous(limits = c(0, 11), breaks = seq(0, 10, 1)) +
  theme_bw() +
  scale_shape_manual() +
```

```
  set_geom_fonts(family = "times", ggrepel = TRUE) +
  theme(text = element_text(family = "times", size = 12))

p <- ggpubr::ggarrange(pA, pB, labels = "AUTO", common.legend = T, legend = "bottom",
                       font.label = list(size = 14, color = "black",family = "times"),
                       align = "hv", nrow = 1)
p

# linear model without interaction term
gene_lm1 <- lm(gene_expression ~ Treatment + conc + cell_line + name, data = gene_fl)
# linear model with interaction term
gene_lm2 <- lm(gene_expression ~ Treatment * conc + cell_line + name, data = gene_fl)

# linear mixed-effect model
# without interaction term
gene_lmm1 <- lmer(gene_expression ~ Treatment + conc + cell_line + (1|name), data = gene_fl)
# with interaction term
gene_lmm2 <- lmer(gene_expression ~ Treatment * conc + cell_line + (1|name), data = gene_fl)
anova(gene_lm1, gene_lm2)
anova(gene_lmm1, gene_lmm2)
# Compare models with interaction term
# AIC
aic_model1 <- AIC(gene_lm2)
aic_model2 <- AIC(gene_lmm2)
cat("AIC of gene_lm2:", aic_model1, "\n", "AIC of gene_lmm2:", aic_model2, "\n")

# BIC
bic_model1 <- BIC(gene_lm2)
bic_model2 <- BIC(gene_lmm2)
cat("BIC of gene_lm2:", bic_model1, "\n", "BIC of gene_lmm2:", bic_model2, "\n")
# Uncomment the following codes to produce diagnostic plots

# # Diagnostic plots
# # Linear model
# gglm(gene_lm2)
#
# # Linear mixed-effect model
# # Residual vs fitted
# plot(gene_lmm2, xlab="Fitted values", ylab = "Residuals", main="Residual vs Fitted")
# # qqplot
# qqnorm(residuals(gene_lmm2))
# # Residual vs leverage
# ggplot(
#   data.frame(
#     lev = hatvalues(gene_lmm2),
#     pearson = residuals(gene_lmm2, type = "pearson")),
#   aes(x = lev, y = pearson)) +
#   labs(x = "Leverage", y="Standardized Residuals", title= "Residual vs Leverage")+
#   geom_point() +
#   geom_smooth()+
#   theme_bw()

comparison <- data.frame(
```

```
  Model = c("gene_lm2", "gene_lmm2"),
  AIC = c(502.6321, 518.5759),
  BIC = c(529.8829, 535.9173)
)

kable(comparison, caption = "Comparison of AIC and BIC values for LM and LMM")
```