Common Statistical Models with R Examples

Anbin Rhee

2025-07-01

Table of contents

1 Simple Linear Regression (SLR)

Simple Linear Regression is a method to model the relationship between **one predictor variable (X)** and **one outcome variable (Y)**, assuming the relationship is roughly linear. Its purpose is to predict the outcome variable based on the predictor and to quantify how much change in the predictor leads to change in the outcome. The SLR equation takes the form

$$Y = \beta_0 + \beta_1 X + \epsilon,$$

where β_0 is the intercept, β_1 is the slope, and ϵ is random error. SLR helps us understand and predict how one variable influences another, under assumptions that the relationship is linear, the errors are normally distributed and have constant variance, and observations are

independent. In practice, SLR is like drawing the best-fitting straight line through a scatter plot to make sense of patterns in the data. For example, we might ask: how much does a student's exam score improve for each extra hour of study?

```
library(ggplot2)
library(stats)
# set your directory
setwd("~/Desktop/RA,TA,SAIG/DSPG/Stat Lectures")
# load the data file
data_SLR <- read.csv(file="exam_data")
# scatter plot
ggplot(data_SLR, aes(x=study_hours, y=exam_score))+
  geom_point()+
  labs(title="Relationship between Study Hours and Exam Scores")+
    theme_minimal()
```

Scatter plots are super important! They help us to see if there is a pattern, a linear trend, or outliers before we fit the model. Here, you see a clear upward trend: more study hours generally means higher exam scores.

Now, let's fit the model: exam_score $= \beta_0 + \beta_1$ study_hours $+ \epsilon_i$.

```
# run SLR models with lm function
SLR <- lm(exam_score~study_hours, data=data_SLR)
# output
summary(SLR)
```

The simple linear regression model examined the relationship between exam score and study hours. The intercept of about 37.6 suggests that a student who studied zero hours would be expected to score roughly 37.6 points on the exam, which represents the model's baseline. The estimated slope of 5.52 means that for each additional hour of study, the expected exam score increases by about 5.5 points. This effect is highly statistically significant, with a p-value less than 2e-16, providing strong evidence that more study hours are associated with higher exam scores. The model's R-squared value is approximately 0.85, indicating that about 85% of the variability in exam scores can be explained by study hours alone, which is quite high and suggests an excellent fit. The residual standard error is about 5.1, meaning the typical difference between the observed scores and the predicted scores is around 5 points. Overall, these results show a strong positive relationship between the number of hours studied and exam performance, with more study time leading to substantially higher predicted scores.

1.1 Model Assumption Check

1.1.1 Independence and Constant Variance

```
data_SLR$residuals <- resid(SLR)
data_SLR$fitted <- fitted(SLR)
ggplot(data_SLR, aes(x = fitted, y = residuals)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "red") +
  labs(
    title = "Homoscedasticity Check",
    x = "Fitted Values",
    y = "Residuals"
  ) +
  theme_minimal()
```

This code checks two important assumptions of simple linear regression: **independence** and **constant variance (homoscedasticity)** of the residuals. After extracting the residuals and fitted values from the regression model, the code plots the residuals against the fitted values using `ggplot2`, with a horizontal red reference line at zero.

This plot helps to assess whether the residuals are randomly scattered around zero without any clear pattern, which supports the **independence** assumption (no time or sequence pattern) as well as **homoscedasticity**, which requires the vertical spread of residuals to remain roughly the same across the range of predicted values.

In the displayed plot, the residuals appear fairly randomly distributed around zero, with no obvious systematic pattern or trend, which suggests that the independence assumption is reasonable. Additionally, the spread of the residuals is relatively even across fitted values, supporting the constant variance assumption. There is some slight variability at higher fitted values, but no severe funnel or curved patterns, indicating the model meets these assumptions well enough for valid inference.

1.1.2 Linearity

```
ggplot(data_SLR, aes(x = study_hours, y = exam_score)) +
  geom_point() +
  geom_smooth(method = "lm", se = TRUE, color = "blue") +
  labs(
    title = "Linearity Check",
    x = "Study Hours",
```

```
    y = "Exam Score"
  ) +
  theme_minimal()
```

This plot is designed to check the **linearity assumption** of simple linear regression, which requires that the relationship between the predictor (study hours) and the response variable (exam score) is approximately linear. The code uses `ggplot2` to plot the raw data points and overlays a linear trend line with a confidence band using `geom_smooth(method = "lm")`. In the resulting plot, the blue regression line captures the overall trend of the data, while the shaded gray area shows the 95% confidence interval for the line. The pattern of data points closely follows a straight line, with no evidence of curves or major deviations, supporting the linearity assumption. This means a linear model is appropriate for describing the relationship between study hours and exam scores.

Visualizing the linear relationship in this way is a very important first step before interpreting the model's coefficients, because if the true relationship were curved or non-linear, the linear regression results could be misleading. This check reassures us that using a straight line to model these data makes sense.

1.1.3 Normality

```
ggplot(data_SLR, aes(x = residuals)) +
  geom_histogram(bins = 15, fill = "lightblue", color = "black") +
  labs(
    title = "Histogram of Residuals",
    x = "Residuals",
    y = "Count"
  ) +
  theme_minimal()
```

This plot is used to check the **normality assumption** of simple linear regression, which says that the residuals should be approximately normally distributed. The code uses `ggplot2` to create a histogram of the residuals from the model, with 15 bins and a light blue fill to make the distribution easy to see.

The histogram shows the counts of residuals across their range. Ideally, for normality, we hope to see a roughly bell-shaped, symmetric pattern centered around zero. In this case, the histogram appears reasonably symmetric, with most residuals clustered around zero and fewer residuals farther from zero on either side. There are no dramatic gaps or clear skew, although there might be a slightly longer right tail, but nothing extreme.

This supports the idea that the residuals are approximately normal, which is important because normal residuals help ensure the reliability of hypothesis tests and confidence intervals in linear

regression. It is always good practice to combine this histogram with a **QQ plot** for a more detailed normality check, which you can do next.

```
ggplot(data_SLR, aes(sample = residuals)) +
  stat_qq() +
  stat_qq_line(color = "red") +
  labs(
    title = "Normal Q-Q Plot of Residuals"
  ) +
  theme_minimal()
```

This QQ plot is another way to check the **normality assumption** of residuals in simple linear regression. The code uses `ggplot2` to plot the quantiles of the model's residuals against the theoretical quantiles from a normal distribution, adding a reference red line with `stat_qq_line()`.

If the residuals are approximately normally distributed, then the points on the QQ plot should fall roughly along the red line. In this output, most points follow the red line quite closely in the middle range, with only a few mild deviations at the tails, especially on the upper right side. This is generally acceptable for regression, as slight deviations in the tails are common and not usually a serious concern.

Overall, this QQ plot supports the histogram finding that the residuals are approximately normal, which validates the regression's inference procedures such as confidence intervals and p-values. Using both a histogram and a QQ plot together is a good practice to build confidence that the normality assumption is reasonable for the data.

2 Multiple Linear Regression (MLR)

Multiple linear regression (MLR) is a statistical technique that models the relationship between one continuous outcome variable and two or more predictor variables. It extends simple linear regression by allowing you to include multiple independent variables simultaneously, so you can examine their combined effects on the outcome while adjusting for one another. The MLR equation takes the form

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon,$$

where 0 is the intercept, β_1 through β_p are the slopes for each predictor, and is random error. MLR helps you understand how each predictor is associated with the outcome while holding the other predictors constant. In practice, MLR is a powerful way to answer questions like "How does salary change depending on experience, education level, and hours worked, all at the same time?"

```r
# load the salary_data file
salary_data <- read.csv("salary_data")
# see the first six data
head(salary_data)
# visualization: experiance vs. salary by education level
ggplot(salary_data, aes(x=experience, y=salary, color=education_level)) +
  geom_point(size=2, alpha=0.7) +
  labs(title="Employee Salaries by Experience and Education Level",
       x="Years of Experience",
       y= "Annual Salary",
       color="Education Level")+
  theme_minimal()

# visualization: hours per week vs. salary by education level
ggplot(salary_data, aes(x=hours_per_week, y=salary, color=education_level)) +
  geom_point(size=2, alpha=0.7) +
  labs(title="Employee Salaries by Weekly Hours and Education Level",
       x="Hours Worked per Week",
       y= "Annual Salary",
       color="Education Level") +
  theme_minimal()
```

The first scatter plot shows how salary increases with experience, with each education level colored differently. You can see that higher education levels tend to cluster higher up, showing higher salaries for the same experience.

The second scatter plot shows how salary changes with hours worked per week, with each education level. There does not appear to be a very strong upward or downward trend along the x-axis, suggesting that hours worked per week alone does *not* have a strong relationship with annual salary. Salaries are spread out across different hours worked, meaning people working 35 hours and those working 45 hours can have similar salaries. Also, Higher education levels tend to sit higher up on the salary scale, regardless of hours worked.

```r
# set the baseline category as High_school
salary_data$education_level <- factor(salary_data$education_level, ordered = FALSE)
salary_data$education_level <- relevel(salary_data$education_level, ref = "High School")

# run multiple linear regression
MLR <- lm(salary ~ experience + education_level + hours_per_week, data = salary_data)
summary(MLR)
```

The multiple linear regression results show that experience, education level, and hours worked per week were used to predict annual salary. The intercept of about $35,480 represents the

expected salary for a person with a high school education (the reference group), zero years of experience, and working zero hours per week, although this is a hypothetical baseline. Each additional year of experience is associated with an increase of about $1,908 in salary, holding other variables constant, and this effect is highly significant. Compared to employees with a high school education, those with a bachelor's degree earn about $5,375 more on average, those with a master's degree earn roughly $10,936 more, and those with a PhD earn about $15,382 more, adjusting for experience and weekly hours; all these differences are statistically significant. On the other hand, the hours worked per week showed an estimated effect of about $116 per additional hour, but this was not statistically significant ($p = 0.22$), suggesting we cannot conclude a real effect of working hours on salary after accounting for education and experience. The model explains approximately 86.5% of the variation in salaries (R-squared = 0.865), indicating an excellent fit, and the overall F-test shows the model is statistically significant as a whole. Overall, this analysis suggests that education level and experience are important predictors of salary, while the number of hours worked per week does not show a clear effect in this data.

3 Two Sample T-Test (Comparing Two Groups)

A two-sample t-test is a statistical method used to compare the means of two independent groups to determine if there is evidence of a significant difference between them. This test is commonly applied when you have one categorical variable with two groups (such as Campaign A vs. Campaign B) and a continuous outcome variable (like purchase amount). The test assumes that observations are independent, the outcome variable is approximately normally distributed in each group, and that the variances are roughly equal (though the Welch version of the test relaxes that last assumption). The null hypothesis states that the two groups have equal population means, while the alternative hypothesis suggests a difference exists.

```
campaign_data <- read.csv("campaign_data")
# check the data
head(campaign_data)

ggplot(campaign_data, aes(x=campaign, y=purchase_amount, fill=campaign)) +
  geom_boxplot() +
  labs(
    title = "Comparison of Purchase Amounts by Campaign",
    x = "Campaign",
    y = "Purchase Amount ($)"
  ) +
  theme_minimal()
```

The boxplot created with `ggplot2` shows the distribution of purchase amounts for each campaign group. The plot indicates that Campaign B tends to have higher purchase amounts than Campaign A: the median (middle line of the box) for Campaign B is higher, and the upper quartile also appears greater. This suggests that Campaign B may have encouraged higher spending. The boxplot also helps us quickly check for outliers and see if the spread of the data looks similar between groups.

```
# two-sample t-test
t.test(purchase_amount ~ campaign, data = campaign_data)
```

This two-sample t-test compares the mean purchase amounts between Campaign A and Campaign B. The test statistic is approximately –4.81, with about 170 degrees of freedom, and a very small p-value of 3.36×10 , which is far below the usual 0.05 threshold. This means we reject the null hypothesis and conclude that there is a statistically significant difference in average purchase amounts between the two campaigns. Specifically, the mean purchase amount for Campaign A was about $50.63, while for Campaign B it was about $60.35, suggesting that Campaign B led to higher spending on average. The 95% confidence interval for the difference in means ranges from –13.7 to –5.7 dollars, which does not include zero, further supporting that the difference is significant. Overall, this provides strong evidence that Campaign B is more effective in increasing customer purchase amounts than Campaign A.

```
summary(lm(purchase_amount ~ campaign, data=campaign_data))
```

4 ANOVA (Comparing Multiple Groups)

Analysis of Variance, or **ANOVA**, is a statistical method used to compare the means of three or more independent groups. It tests whether at least one group mean is significantly different from the others.

The ANOVA equation takes the form

$$Y_{ij} = \mu + \alpha_j + \epsilon_{ij},$$

where μ is the grand mean, α_j effect of being j group, and is random error.

The ANOVA procedure partitions the total variability in the data into between-group differences and within-group differences, then compares these using an F-test. The null hypothesis is that all group means are equal, while the alternative hypothesis is that at least one group differs.

$$H_0 : \alpha_1 = \alpha_2 = \cdots = \alpha_J \quad \text{vs.} \quad H_1 : \text{At least one group is different}$$

ANOVA assumes the data are normally distributed within groups, have equal variances, and are based on independent observations. If the ANOVA test is significant, you can follow up with post-hoc tests (like Tukey HSD) to pinpoint which groups differ from each other. ANOVA is a fundamental tool for analyzing experiments, marketing campaigns, educational methods, and any context involving multiple groups to be compared on a continuous outcome. Here, we will use ANOVA to compare purchase amounts across multiple campaigns (Campaign A, B, or C).

```
# load the multiple campaign data
multiple_campaign_data <- read.csv("multiple_campaign_data")
# see the first six observations
head(multiple_campaign_data)
# visualization : boxplot
ggplot(multiple_campaign_data, aes(x=campaign, y=purchase_amount, fill=campaign)) +
  geom_boxplot() +
  labs(
    title = "Purchase Amount by Campaign",
    x = "Campaign",
    y = "Purchase Amount ($)"
  ) +
  theme_minimal()
```

The boxplots display the distribution of purchase amounts for each campaign group (A, B, and C). For each campaign, the box represents the middle 50% of the data (the interquartile range), with the horizontal line inside the box showing the median purchase amount. The vertical lines (whiskers) extend to show the range of most of the data, and dots outside these whiskers represent outliers.

In this plot, Campaign B's box is higher overall, with a higher median and a wider spread of purchase amounts, suggesting that customers exposed to Campaign B tend to spend more and with greater variability. Campaign A has the lowest median and a tighter box, showing lower and more consistent purchase amounts. Campaign C sits somewhere in between, with a median higher than Campaign A but lower than B, and it shows a few outliers on the higher side.

Overall, these boxplots visually suggest that there are differences in typical spending between the campaigns, with Campaign B likely being the most successful, and provide a good motivation to test these differences formally with ANOVA.

There are two ways to run ANOVA models in R.

```
# ANOVA
# option 1
```

```r
anova_model <- aov(purchase_amount ~ campaign, data=multiple_campaign_data)
summary(anova_model)
# Post-hoc test if needed
TukeyHSD(anova_model)
```

The ANOVA results in the first summary test whether there are any differences in mean purchase amounts across Campaigns A, B, and C. The F-test is highly significant ($F = 25.09$, $p < 0.0000000001$), which means there is strong evidence that at least one campaign differs from the others in average purchase amount. However, ANOVA does not say which pairs of campaigns are different, so a post-hoc test is needed.

The Tukey HSD post-hoc test provides those pairwise comparisons. It shows that Campaign B has an average purchase amount about \$13.12 higher than Campaign A ($p < 0.0001$), Campaign C is about \$6.38 higher than Campaign A ($p = 0.0019$), and Campaign C is about \$6.74 lower than Campaign B ($p = 0.0009$). All of these differences are statistically significant after adjusting for multiple comparisons.

```r
# option 2: preferred
anova_lm <- lm(purchase_amount~campaign, data=multiple_campaign_data)
summary(anova_lm)
```

The alternative approach with `lm()` fits the same ANOVA in a regression framework, coding Campaign A as the baseline category. The coefficients confirm the same story:

1. Campaign B is associated with a \$13.12 higher average purchase than Campaign A ($p < 0.0001$),
2. Campaign C is associated with a \$6.38 higher average purchase than Campaign A ($p = 0.00065$).

This linear model formulation makes it easier to interpret the differences directly against the reference group, Campaign A. The residual standard error and R-squared match the ANOVA approach, showing consistent results.

Overall, these outputs strongly support that Campaign B performed best in encouraging higher spending, followed by Campaign C, with Campaign A performing worst.

5 Logistic Regressions

Logistic regression is a statistical modeling technique used when the outcome variable is binary — that is, it has only two possible values, often coded as 0 or 1. Instead of predicting a continuous outcome like linear regression, logistic regression models the *probability* that an observation belongs to a particular category (usually coded as 1). It does this by estimating

the log-odds of success as a linear combination of predictor variables. Logistic regression is very popular for medical trials, marketing conversion rates, and other situations where the outcome is yes/no, success/failure, or event/no event. Interpreting logistic regression results typically focuses on odds ratios, which describe how the odds of the outcome change with each predictor.

```
# load the treatment data
treatment_data <- read.csv("treatment_data")
# quick check frequency table
table(treatment_data$treatment, treatment_data$success)


ggplot(treatment_data, aes(x=treatment, fill=factor(success))) +
  geom_bar(position="fill") +
  scale_y_continuous(labels=scales::percent_format()) +
  labs(
    title = "Treatment Success Rates",
    y = "Percentage",
    fill = "Success"
  ) +
  theme_minimal()
```

This simple frequency table suggests that the New treatment might be performing somewhat better than the Standard treatment.

Next, you used `ggplot` to create a bar plot showing the proportion of successes within each treatment group. The `position=\"fill\"` argument rescales the bars to 100%, making it easier to compare relative success rates rather than raw counts. According to the bar plot, the New treatment group has a higher proportion of successes (shown in teal) than the Standard group, though both have fairly high failure rates.

```
# logistic regression
model <- glm(success ~ treatment + age + gender, family=binomial, data=treatment_data)
summary(model)
```

This logistic regression model predicts the probability of treatment success using three predictors: treatment group, age, and gender. The intercept represents the log-odds of success for the reference category, which is the new treatment, at age 0, for females. The coefficient for `treatmentStandard` is –0.742, which means that being in the standard treatment group is associated with lower log-odds of success compared to the new treatment, adjusting for age and gender. If we exponentiate this coefficient, `exp(-0.742)  0.48`, suggesting that the odds of success in the standard treatment group are about 52% lower than in the new treatment group. This effect is statistically significant with a p-value of about 0.039, supporting the

conclusion that the new treatment has higher success odds.

The coefficient for age is about –0.034, with a p-value of 0.073, indicating a possible (though not strongly significant) trend where each additional year of age slightly reduces the odds of success; the odds ratio `exp(-0.034)  0.97` means about a 3% decrease in odds per year of age. For gender, males have a coefficient of –0.55 compared to females, corresponding to an odds ratio of `exp(-0.55)  0.58`, suggesting about 42% lower odds of success compared to females, although this effect is not statistically significant ($p = 0.126$).

Overall, the model suggests that treatment group is the most important predictor, with the new treatment performing significantly better than the standard treatment, even after adjusting for age and gender. The model's residual deviance (196 on 196 degrees of freedom) is smaller than the null deviance (205), indicating the model explains some variability in success outcomes.