Lab-2: Multiple linear regression

Youssouph Cissokho

2023-10-02 14:42:02

1 Preliminaries

In this presentation, we will use the data set named `p2.13` from the **MPV** package, which is a default `R` dataset. Thus, we don't need to import it; it is immediately available in R.

Goal : how the number of days the ozone levels exceeded 0.20 ppm depends on the seasonal meteorological index, which is the seasonal average 850-millibar temperature.

Let's first load few libraries :

NOTE: If you don't have the package, you need to install it first by doing the following in the commande line:

1. install.package("MPV")
2. library(MPV)

```
library(ggplot2)  # load the library ggplot2 for visualization load the library ggplot2 for vi
# install.packages('MPV')
library(MPV)  # load the library
```

```
## Loading required package: lattice

## Loading required package: KernSmooth

## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009

## Loading required package: randomForest

## randomForest 4.7-1.1

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
# Most of this package consists of data sets from the
# textbook Introduction to Linear Regression Analysis, by
# Montgomery, Peck and Vining.
```

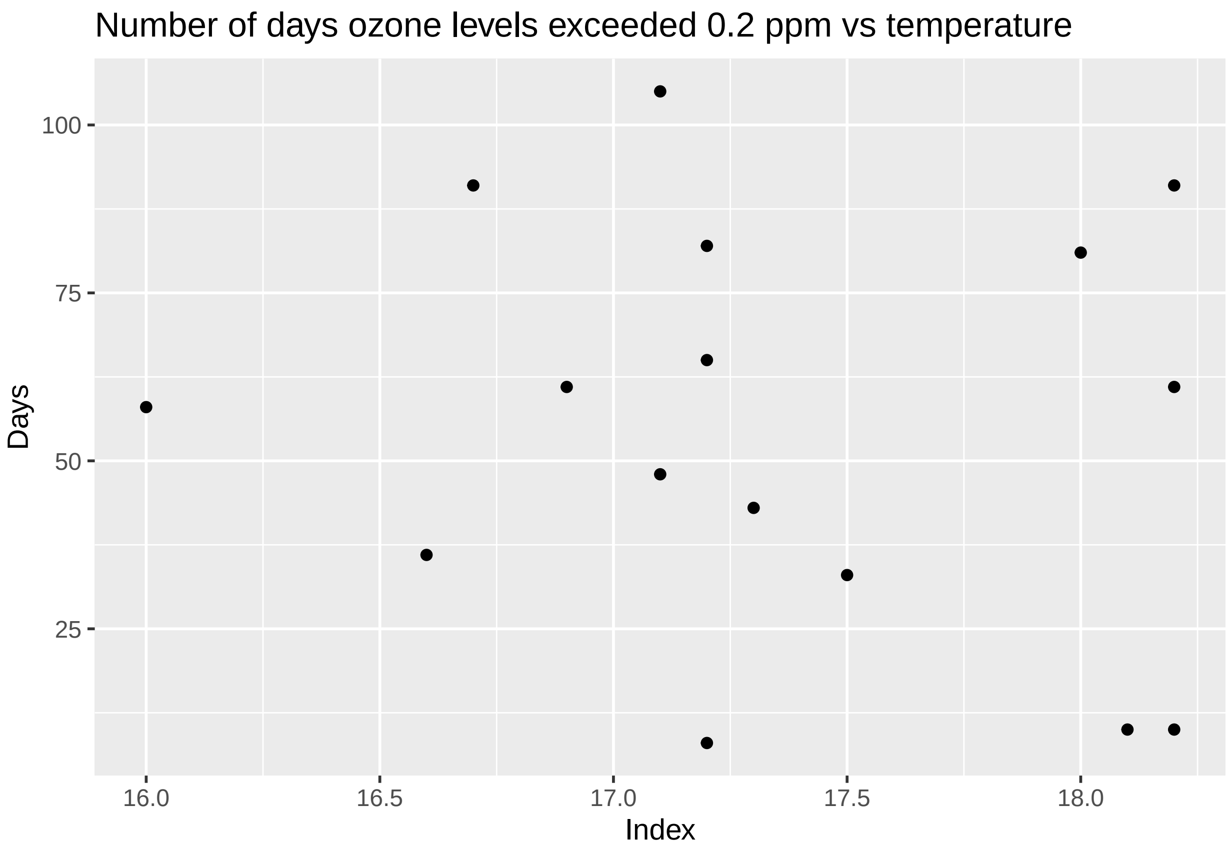
1.1 View structure of the dataset

Now let's examine the `data` in details:

1.2 Visualization of the dataset

We can use `ggplot2` library as discussed in **Lab_1** to plot the data. For more details please visit ggplot2.

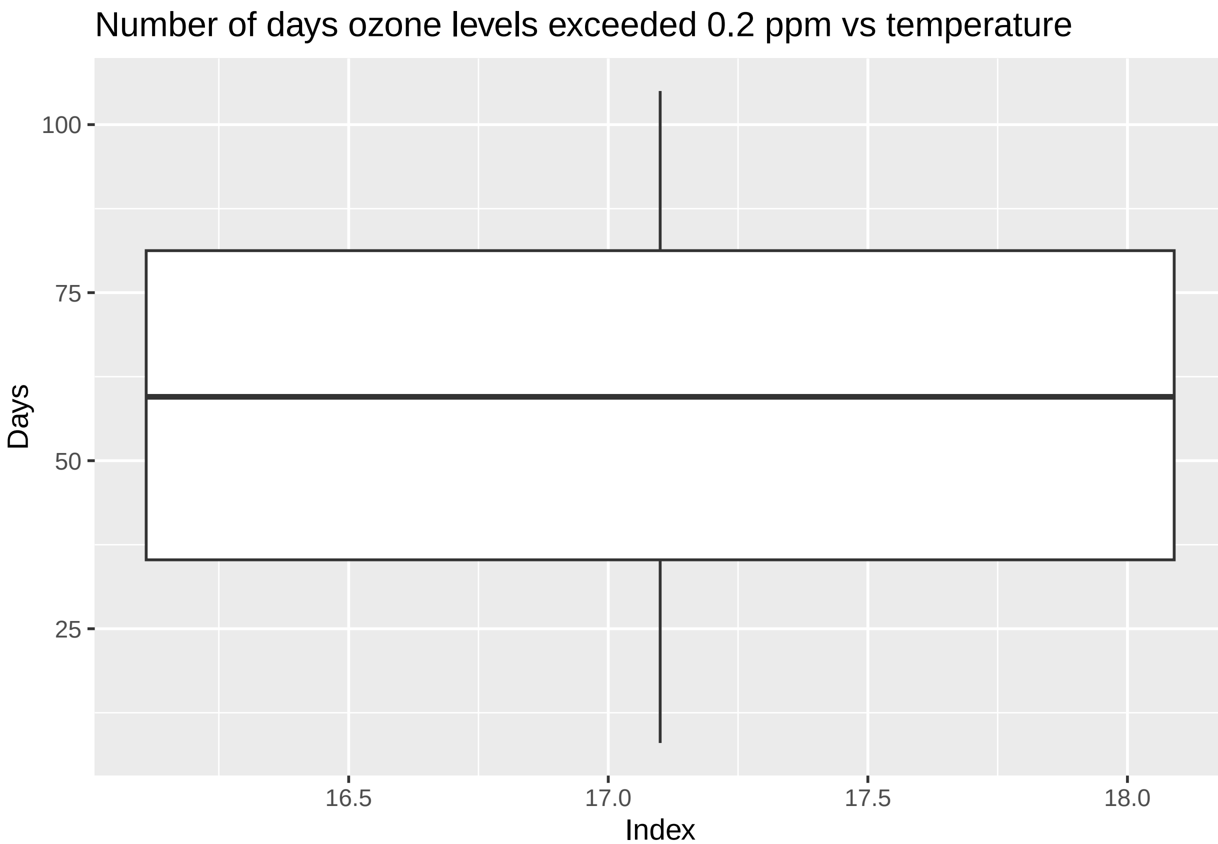
```
ggplot(p2.13, aes(x = index, y = days)) + geom_point() + labs(title = "Number of days ozone lev
    x = "Index", y = "Days")
```



1.3 Checking for outliers

An **Outlier** is a data point that differs significantly from other observations, be may be due to a variability in the measurement, a result of experimental error, etc. Outliers greatly affect the linear regression modelling (Std. error, R^2). Moreover, the F and t tests are not reliable when outliers are present.

```
ggplot(p2.13, aes(x = index, y = days)) + geom_boxplot() + labs(title = "Number of days ozone l
    x = "Index", y = "Days")
```



We do not have any outlier here.

2 Linear regression

Regression models are used for several purposes, including the following:

1. Data description;
2. Parameter estimation;
3. Prediction and estimation;
4. Control.

This simple linear regression model is

$$y = \beta_0 + \beta_1 x + \varepsilon$$

where

- β_0 is intercept;
- β_1 is the slope;
- ε are

$$\epsilon \sim N(0, \sigma^2).$$

ie they are *independent and identically distributed* (iid) normal random variables with mean 0 and variance σ^2.

Goal to estimate β_0 and β_1.

2.1 The `lm()` Function

The `lm()` command is used to fit linear models which actually account for a broader class of models than simple linear regression. The structure of the `lm()` function is as follows:

```
lm(formula, data, subset, weights, na.action, method = "qr",
    model = TRUE, x = FALSE, y = FALSE, qr = TRUE, singular.ok = TRUE,
    contrasts = NULL, offset)
```

where :

- formula has the form **response** ~ **terms** where
  - **response** is the (numeric) response vector
  - **terms** is a series of terms which specifies a linear predictor for response.
- data is the data set, etc. For more use the help in R (`?lm()`).

2.2 Estimation of β_0 and β_1 using lm(...) function in R

```
# returns a linear model object, which is saved in
# `model_fit`
model_fit = lm(formula = days ~ index, data = p2.13)
model_fit  # print the coefficients
```

```
##
## Call:
## lm(formula = days ~ index, data = p2.13)
##
## Coefficients:
## (Intercept)        index
##     183.596       -7.404
```

Printing the `linear-model object` simply shows the estimated regression coefficients. A more complete report is obtained by the summary function.

2.3 Display the summary of the regression

```
# provide alternative summaries of a regression fit
summary(model_fit)
```

```
##
## Call:
## lm(formula = days ~ index, data = p2.13)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -48.252 -21.947  -2.305  26.979  48.008
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  183.596    214.359   0.856    0.406
## index         -7.404     12.351  -0.599    0.558
##
## Residual standard error: 31.2 on 14 degrees of freedom
## Multiple R-squared:  0.02502,    Adjusted R-squared:  -0.04462
## F-statistic: 0.3593 on 1 and 14 DF,  p-value: 0.5585
```

Some comments about this output:

- Rsidual: is the difference between the **observed value** and the corresponding **fitted value**;
- Coefficients: gives a p x 4 matrix with columns for the **estimated coefficient, its standard error, t-statistic and corresponding (two-sided) p-value**
  - **Estimates** gives the estimated regression coefficients;
  - **Std. Error** gives the standard errors;
  - **t value** gives the ratio of the estimate to its std error and is a **Wald test** of the hypothesis that the coresponding coefficient is zero(0);
  - **Pr(>|t|)** gives a two-sided p-value assuming that the t-distribution is appropriate;
- Signif. codes: The number of asterisks (under p-values' column) corresponds to the significance of the coefficient. The more asterisks, the more significant.
- Residual standard error is an estimate of σ. It measures how well the model fits the data (a small residual standard error is preferred);
- Multiple R-squared is the square of the correlation between the response and the fitted values i.e. the 'fraction of variance explained by the model (**Adjusted R-squared** is R^2 adjusted i.e. it deals with an increase in R^2 spuriously due to adding features);
- F-statistic: tests the hypothesis that all the regression coef. are 0 vs at least one of them is non-0, which follows an F-distribution (if errors are normal or large n);

2.4 Construct the analysis-of-variance table and test for significance of regression.

```r
# provide an analysis-of-variance table
anova(model_fit)
```

```
## Analysis of Variance Table
##
## Response: days
##           Df  Sum Sq Mean Sq F value Pr(>F)
## index      1   349.7  349.69  0.3593 0.5585
## Residuals 14 13624.7  973.20
```

The F statistic tells you whether the model is significant or insignificant. The model is significant if any of the coefficients are nonzero (i.e., if $\beta_i \neq 0$ for some i). It is insignificant if all coefficients are zero ($\beta_1 = \beta_2 = \ldots = \beta_n = 0$).

- P-value < 0.05 indicates that the model is significant (at least one β_i is nonzero);
- P-value > 0.05 indicates that the model is not significant.

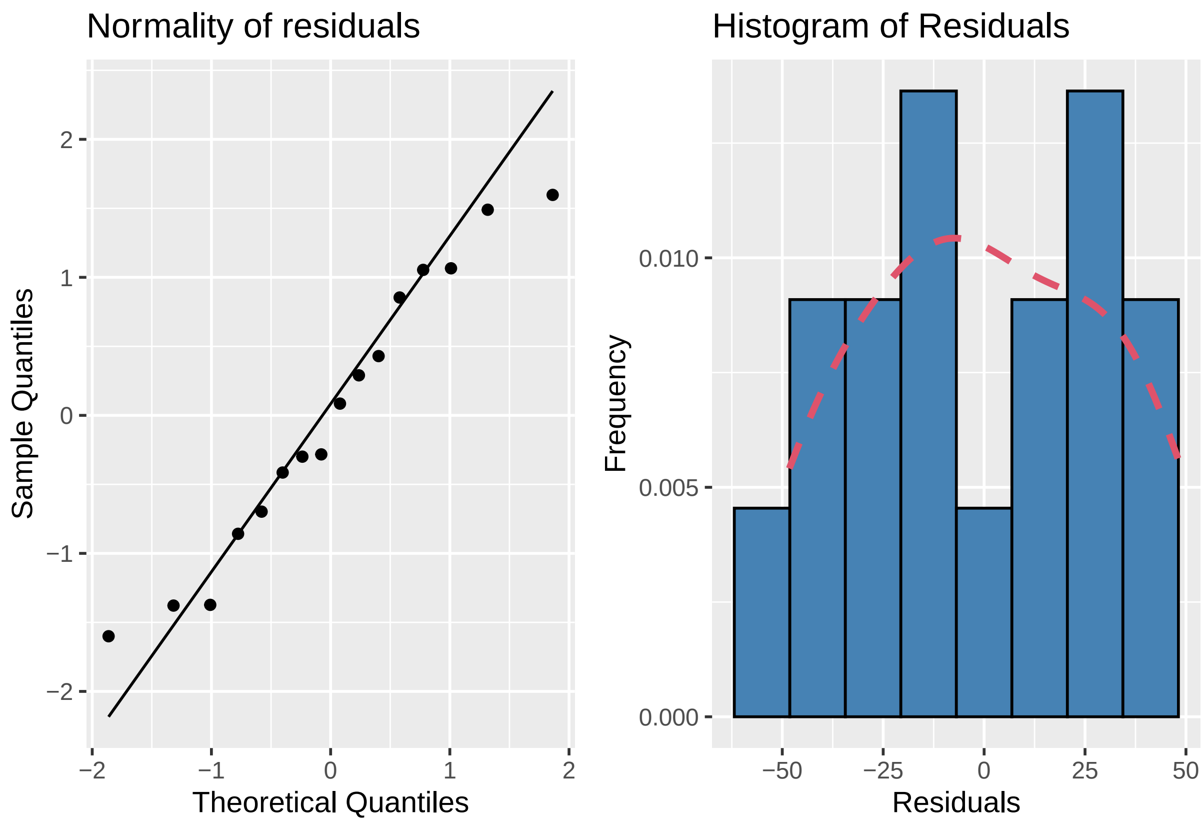
The computed value of F_0 is 0.3598 and the corresponding P-value for this test is 0.5585. Consequently, this model is likely not significant.

2.5 Checking the Normality of Residuals

```r
# this package provides rich support for complex layouts
library(patchwork)
p1 <- ggplot(model_fit, aes(sample = rstandard(model_fit))) +
    geom_qq() + stat_qq_line() + labs(title = "Normality of residuals",
    x = "Theoretical Quantiles", y = "Sample Quantiles")

p2 <- ggplot(p2.13, aes(x = model_fit$residuals)) + geom_histogram(aes(y = ..density..),
    bins = 8, fill = "steelblue", color = "black") + labs(title = "Histogram of Residuals",
    x = "Residuals", y = "Frequency") + geom_density(lwd = 1.2,
    linetype = 2, colour = 2)
p1 | p2
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```



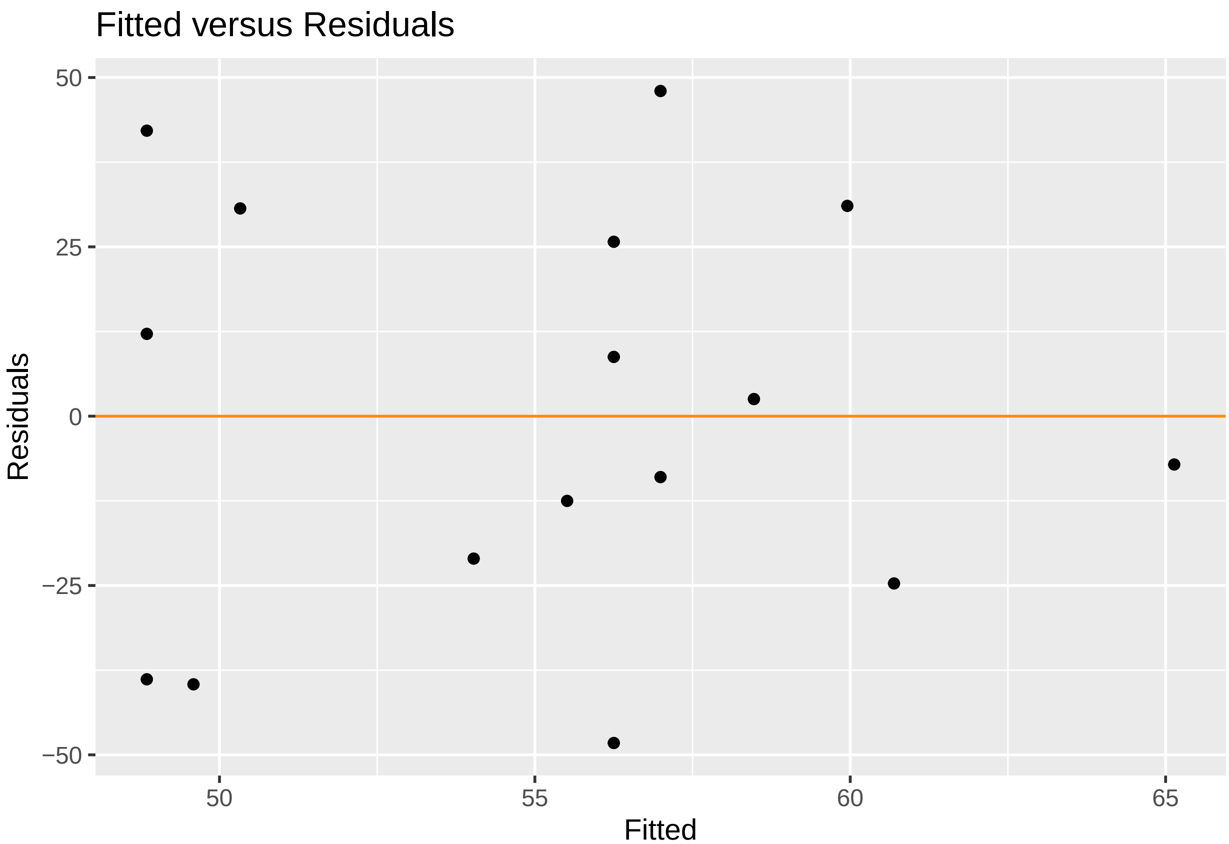
*

The normal probability plot (qqplot) of the residuals is approximately linear hence the assumption that the error terms are normally distributed is reasonable.

2.6 Fitted versus Residuals plot

The residual vs fitted plot is used to detect non-linearity, unequal error variances, and outliers.

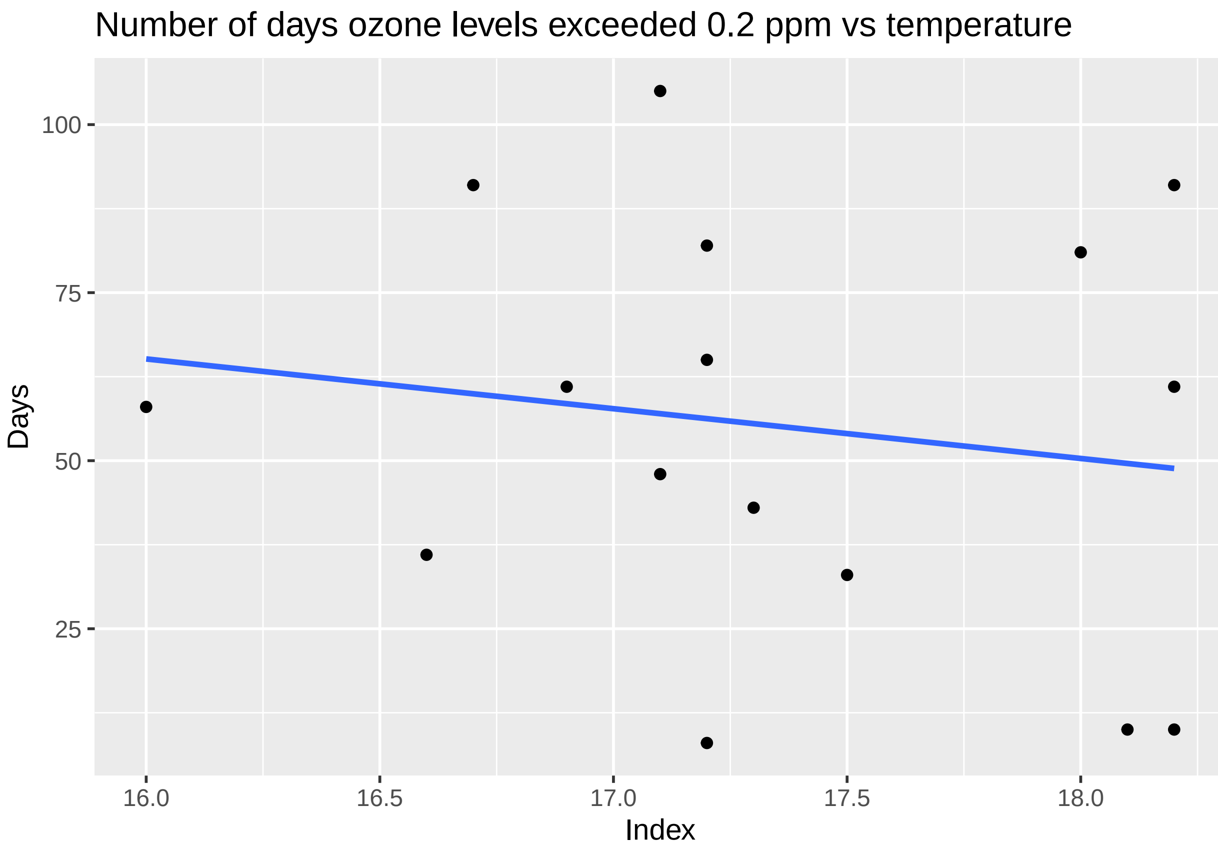
```r
ggplot(model_fit, aes(.fitted, .resid)) + geom_point() + geom_hline(yintercept = 0,
    color = "darkorange") + labs(title = "Fitted versus Residuals",
    x = "Fitted", y = "Residuals")
```



- The residuals vs fitted plot "bounce randomly" around the 0 line. This suggests that the assumption that the relationship is linear is reasonable. Moreover, the residuals roughly form a "horizontal band" around the 0 line. This suggests that the homoscedasticity hypothesis holds (i.e. variances of the error terms are equal).

2.7 Adding a regression line (line of Best-Fit) to the plot

```
ggplot(p2.13, aes(x = index, y = days)) + geom_point() + stat_smooth(method = lm,
    se = FALSE) + labs(title = "Number of days ozone levels exceeded 0.2 ppm vs temperature",
    x = "Index", y = "Days")
```

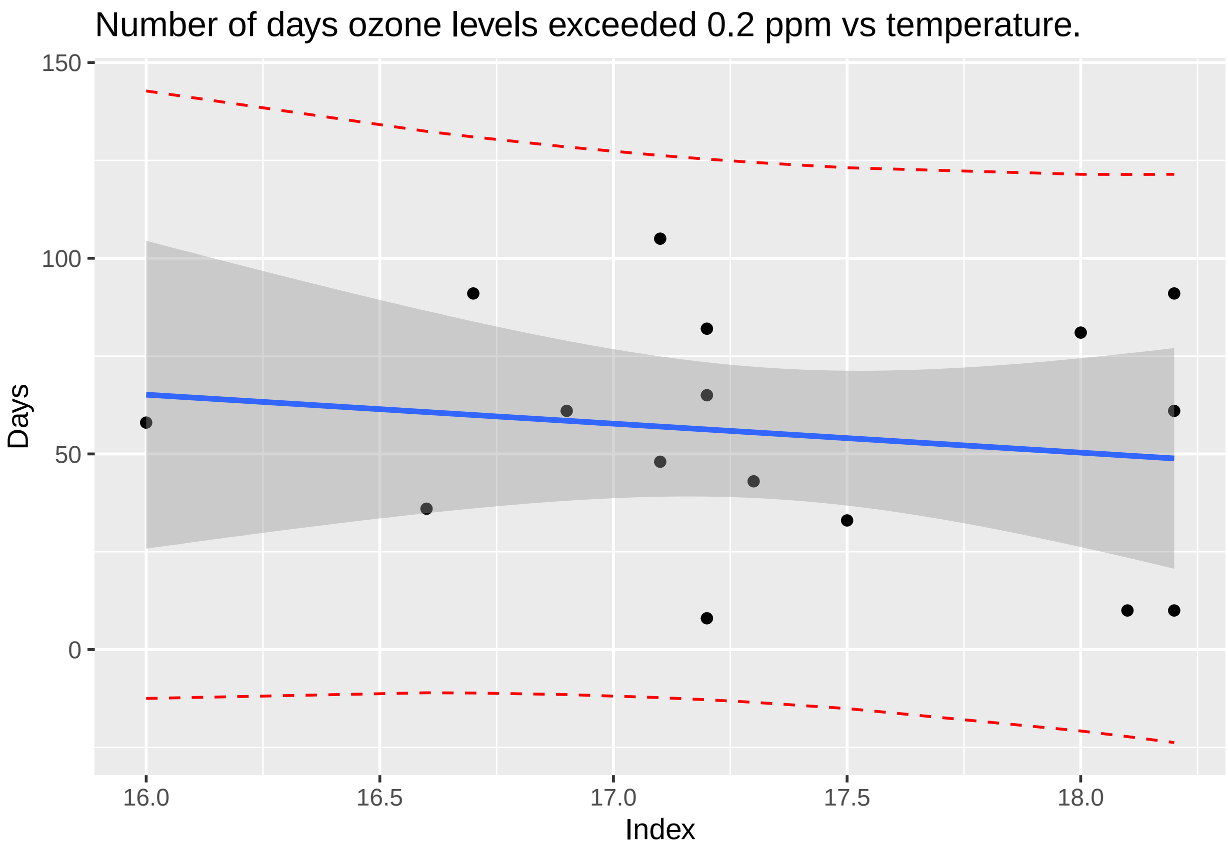


2.8 Plot of 95% confidence and prediction intervals

```
# 95\% prediction intervals
predictions_CI = predict(model_fit, interval = "prediction",
    level = 0.95)

# combining the 2 data sets
New_data = cbind(p2.13, predictions_CI)

ggplot(New_data, aes(index, days)) + geom_point() + geom_line(aes(y = lwr),
    color = "red", linetype = "dashed") + geom_line(aes(y = upr),
    color = "red", linetype = "dashed") + geom_smooth(method = lm,
    se = TRUE) + labs(title = "Number of days ozone levels exceeded 0.2 ppm vs temperature.",
    x = "Index", y = "Days")
```



2.9 The corresponding linear regression model is

```
Days = 183.6 - 7.4 * Index.
```

2.10 Perform prediction (CI)

When estimating the confidence interval (also called the mean interval), the question one is trying to answer is typically as mentioned above:

```
new <- data.frame(index = 28)  # create a new data

# this provides a 95% confidence interval
predict(model_fit, new, interval = "confidence")
```

```
##         fit       lwr      upr
## 1 -23.70863 -306.4965 259.0792
```

```
# this provides a 95% prediction confidence interval
predict(model_fit, newdata = new, interval = "prediction")
```

```
##         fit       lwr     upr
## 1 -23.70863 -314.3042 266.887
```

2.11 Confidence intervals for the parameters

percent confidence interval (CI) on the slope β_1 is given by

```r
confint(model_fit, level = 0.95)
```

```
##                  2.5 %    97.5 %
## (Intercept) -276.15827 643.35059
## index        -33.89456  19.08707
```

2.12 Problems

- Do p2.11 (page 60)
- Do p2.14 (page 62)

3 Summary

In this presentation, we discussed about

- A pre-scanning of the data set in section Preliminaries;
    a. View structure of the dataset
    b. Visualization of the dataset
    c. Checking for outliers
- step by step guide on how to perform a simple linear regression in R in section linear regression
    a. Regression analysis using the command lm() in R
    b. Analysis-of-variance table
    c. Checking the Normality of Residuals
    d. Fitted versus Residuals plot
    e. 95% confidence and prediction intervals
    f. Perform prediction
    g. Confidence intervals for the parameters