AN INTRODUCTION TO R

DEEPAYAN SARKAR

Introduction to statistical inference

The goal of *statistical inference* is to study data to infer knowledge that goes beyond the immediate scope of the data. One usually focuses on two kinds of inference: *estimation* and *testing*. We study various *methods* to do inference. These can be "intuitive" or "common-sense" methods, or they can be rigorously derived based on some criterion. Statisticians always like to study various optimality properties of the methods they study. To make concrete statements about such optimality properties, we usually need to make model assumptions about the data.

Let us start with an example we have already seen: three sets of observations on ten patients at an asylum. The observations recorded are the average increase in the hours of sleep given three sleep-inducing drugs, compared to a control average where no medication was given.

```
> extra.hyoscyamine <- c(0.7, -1.6, -0.2, -1.2, -0.1, 3.4, 3.7, 0.8, 0.0, 2.0)
> extra.laevorotatory <- c(1.9, 0.8, 1.1, 0.1, -0.1, 4.4, 5.5, 1.6, 4.6, 3.4)
> extra.racemic <- c(1.5, 1.4, 0.0, -0.7, 0.5, 5.1, 5.7, 1.5, 4.7, 3.5)
```

First question. Consider the `extra.hyoscyamine` observations, which represents the extra hours of sleep on average when given the hyoscyamine drug. The mean increase is

```
> mean(extra.hyoscyamine)
[1] 0.75
```

However, we are not really interested in these 10 particular patients, but in the effect of the drug in the "general population". To make any statements about that, we need to first link this particular sample to the population.

This is done by hypothesizing a statistical model. In this simple case, our model needs just one univariate distribution; we will pretend that we have planned the experiment but not yet observed the data, and denote the $n = 10$ observations we will see by the symbols $X_1, X_2, \ldots, X_n$. We will assume that these quantities are independent random variables coming from a common but unknown distribution F. The only unknown component of our model is F, which we refer to as the *parameter* of our model. The link between the model and the actual observed data is completed by assuming that the observed data is one realization of these random variables. If we repeated the experiment with n other patients, the observed numbers would be different, but they would be realizations from the same distribution F.

It is difficult to infer much about the unknown distribution F in this model. However, some simple things are possible; for example, we may be interested in the expected value of the X_i-s, given by

$$\mu = E(X_i) = \int_{-\infty}^{\infty} x f(x) \mathrm{d}x$$

Date: October 2011.

where f is the density function of the distribution F. We call μ the *population mean*. A common-sense estimator of μ is the sample mean 0.75 seen above. It also happens to be the *least squares* estimator, in the sense that it is the value of a that minimizes the squared-error loss

$$\sum_{i=1}^{n}(X_i - a)^2.$$

This estimator also has nice optimality properties: it is *unbiased* and has minimum variance among all linear unbiased estimators. What does that mean? Notice that the sample mean

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

is itself a random variable: it would have taken a different value if a different set of 10 patients had been selected for the experiment. The probablity distribution of $\bar{X}$ (which depends on F) has its own mean and variance. $\bar{X}$ is unbiased because $E(\bar{X}) = \mu$, and it also has lower variance than any other (linear) unbiased estimator of μ.

There may be other features of F we may be interested in such as the median and variance; these also have estimators with various optimality properties.

Second question. In our observed sample, $\bar{X}$ has the value 0.75. Does this mean μ is exactly equal to 0.75? Of course not. Can we at least say that μ is positive (that is, giving hyoscyamine is better than giving no drug at all)? Even that is not necessarily true.

To see why, let us do some simulation. It is expensive to perform experiments in real life, but it is cheap to do them in a computer. Consider our model. We do not know the parameter F, but let us suppose for a moment that F was the standard Normal distribution $N(0, 1)$. Here is the mean of one sample of size 10 from the $N(0, 1)$ distribution.

```
> mean(rnorm(10))
[1] -0.4310653
```

We can repeat this experiment several times to get

```
> replicate(20, mean(rnorm(10)))
 [1]  0.353412449  0.066056723 -0.469304540 -0.477413625  0.070744760
 [6] -0.194377583 -0.040043189 -0.529158686  0.199731778  0.385597958
[11]  0.696666885 -0.537843399 -0.367312810  0.001679262 -0.206758857
[16] -0.063791765  0.136722404 -0.422392426 -0.155119988  0.245495107
```

As we can see, $\bar{X}$ is sometimes positive even when the true F has $\mu = 0$. Thus, the fact that $\bar{X}$ is positive in our experiment does not imply that $\mu > 0$. There is not much more we can say about this problem unless we make further model assumptions.

A more specific model. Our previous model made very few assumptions: only that all observations are independent and come from the same distribution. Not much inference can be done in such a general setup. We will now make a much more specific model: we will assume that F is a Normal distribution, with mean μ and variance σ^2; that is,

$$X_i \sim N(\mu, \sigma^2),\ i = 1, 2, \ldots, n$$

The question we are interested in is still whether μ is positive. The intuitive idea is that we would be more inclined to believe this conjecture if in addition to $\bar{X}$ being positive, the variance is also small. The familiar

unbiased estimator of variance is

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2.$$

Let us try our simulation approach again, assuming that $\mu = 0$. But looking at a very specific F as we did before (the standard Normal) will not allow us to make any general conclusions, so we would like to allow σ^2 to be any positive quantity. However, this seems pointless, as we cannot simulate from $N(0, \sigma^2)$ if we do not know σ^2. What can we do then?

The trick is to notice that if we scale $\bar{X}$ to obtain a new quantity

$$U = \frac{\bar{X}}{\sqrt{s^2}}$$

then the distribution of U does not depend on σ^2.

Exercise 1. *Let $X_1, \ldots, X_n \sim N(0, \sigma^2)$ independently. Let $Z_i = \frac{X_i}{\sigma}, i = 1, \ldots, n$. Then by definition, $Z_i \sim N(0, 1)$. Let*

$$\begin{aligned} U_X &= \frac{\bar{X}}{\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2}}, \text{ and} \\ U_Z &= \frac{\bar{Z}}{\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(Z_i - \bar{Z})^2}} \end{aligned}$$

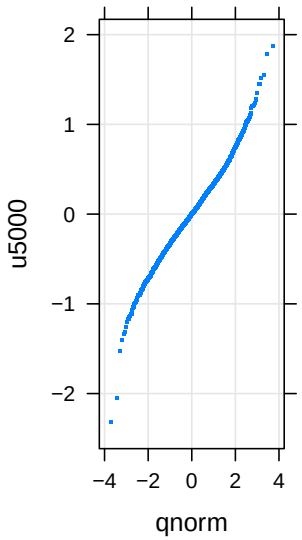
Prove that $U_X = U_Z$. As the distribution of U_Z clearly does not depend on σ^2, the distribution of U_X must also not depend on σ^2.

This means that whatever value of σ^2 we use to simulate, the computed value of $U = \frac{\bar{X}}{s}$ will have the same distribution; so we may as well use $\sigma^2 = 1$. Let us compute U from 5000 simulations of our model.

```
> u <- function() {
      x <- rnorm(10, mean = 0, sd = 1)
      mean(x) / sd(x)
  }
> u5000 <- replicate(5000, u())
> summary(u5000)
      Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
-2.3200000 -0.2205000  0.0007736  0.0023330  0.2306000  1.8770000
```

What is the distribution of this U? Is it Normal? We can check using a Normal Q-Q plot.

```
> library(lattice)
> qqmath(~u5000, distribution = qnorm, type = c("p", "g"),
         aspect = "xy", pch = ".", cex = 2)
```



Exercise 2. *Does this look like a straight line? Compare with the Normal Q-Q plot of values directly simulated from the Normal distribution.*

The systematic curvature is indicative of tails wider than the Normal distribution.

Simulation p-value. We now have 5000 simulated values of U under the assumption that $\mu = 0$ (in the model that F is a Normal distribution). What is the value of U in our actual experiment?

```
> print(U <- mean(extra.hyoscyamine) / sd(extra.hyoscyamine))
[1] 0.4192264
```

Let us compute the proportion of cases in which our simulated U is larger than our observed U.

```
> sum(u5000 > U) / length(u5000)
[1] 0.1058
```

Thus under our model, there is a roughly 10% chance that even when $\mu = 0$, we see a value of U at least as large as what we have actually seen. Of course, our observed proportion is itself an estimate of this chance, but we can compute it more accurately by simulating more samples.

```
> sum(replicate(100000, u())  > U) / 100000
[1] 0.10795
```

We have just performed a hypothesis test, where we tested the *null hypothesis*

$$H_0 : \mu = 0$$

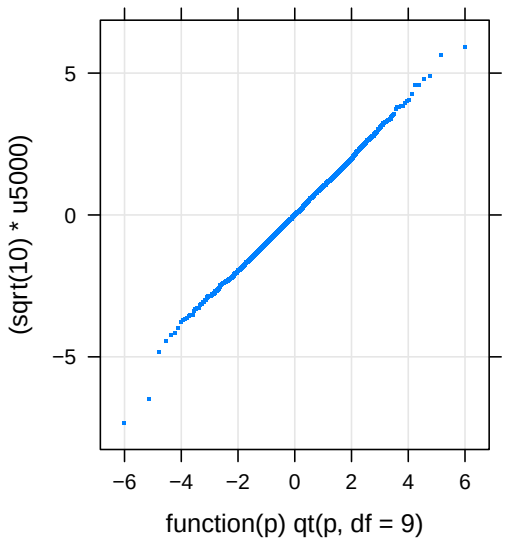
against the alternative hypothesis

$$H_1 : \mu > 0.$$

The proportion computed is the p-value $P(U > u_{\text{observed}})$, which represents the "surprise factor": how unlikely is our observed sample given our null hypothesis?

The true p-value. Our Q-Q plots suggest that the distibution of U when $\mu = 0$ is not Normal, but does not tell us what it is. It so happens that the distribution of U can be derived analytically, and belongs to the family of t-distributions. In fact, $\sqrt{n}\ U \sim t_{n-1}$ (the t-distribution with $n-1$ degrees of freedom), as can be graphically verified by a Q-Q plot against the t_{n-1} distribution. $\sqrt{n}\ U$ is referred to as the *test statistic*, as it is the quantity (statistic) on which the test is based.

```
> qqmath(~ (sqrt(10) * u5000), distribution = function(p) qt(p, df = 9),
         type = c("p", "g"), aspect = "iso", pch = ".", cex = 2)
```



We do not need to depend on simulation to compute the actual p-value; it can be computed using

```
> pt(sqrt(10) * U, df = 9, lower.tail = FALSE)
[1] 0.1087989
```

A p-value of this magnitude is not considered strong enough evidence against our conjecture (null hypothesis) that $\mu = 0$; in other words, we would not conclude in this case that that hyoscyamine is an effective sleep-inducing drug. The usual practice is to consider a p-value less that 0.05 to be "significant", or strong enough evidence (if we stick by such a rule whenever we test, we would erroneously reject the null hypothesis 5% of the time).

Of course, it is possible that our lack of evidence is simply due to the fact that we do not have enough data. Not having enough evidence "against the null" does not mean that we have evidence against the alternative.

Exercise 3. *Repeat the procedure above to compute p-values for testing whether the two other drugs (laevorotatory and racemic) significantly increase the average duration of sleep.*

The R function `t.test()` will also perform the test described above, which is also known as a one-sample t-test.

```
> t.test(extra.hyoscyamine, alternative = "greater")

        One Sample t-test

data:  extra.hyoscyamine
t = 1.3257, df = 9, p-value = 0.1088
alternative hypothesis: true mean is greater than 0
95 percent confidence interval:
 -0.2870553        Inf
sample estimates:
mean of x
     0.75
```

Third question. Let us now ask a slightly different question: is the drug laevorotatory any better than hyoscyamine? Since the observations are paired (measured on the same patients), this is the same as asking if the mean of their difference is positive. R can perform a paired t-test using

```
> t.test(extra.laevorotatory, extra.hyoscyamine,
         alternative = "greater", paired = TRUE)
```

Exercise 4. *Confirm that the paired t-test above gives the same results as*

```
> t.test(extra.laevorotatory - extra.hyoscyamine, alternative = "greater")
```

A non-parametric test. Let us now go back to a more general model for our data, with F only assumed to be symmetric around the mean μ. As before, we would like to test

$$H_0 : \mu = 0 \; vs \; H_1 : \mu > 0$$

The Wilcoxon signed-rank test works as follows: rank the observations according to their numeric value (ignoring their sign), and then compute the sum of ranks for only the positive observations. The idea is that with only the symmetry assumption, the test statistic corresponds to selecting each number from 1 to n with probablity 0.5 and taking their sum.

It is clear that as before, the distribution of the test statistic will not depend on F (as long as $\mu = 0$ and F is symmetric). We can use simulation from any symmetric distribution with mean 0 to estimate the p-value, or use a theoretical calculation. The distribution of the test statistic is not a standard distribution, but is available in R (see `?pwilcox`). The `wilcox.text()` function will perform the directly given the data.

```
> wilcox.test(extra.hyoscyamine, alternative = "greater")

        Wilcoxon signed rank test with continuity correction

data:  extra.hyoscyamine
V = 31, p-value = 0.1716
alternative hypothesis: true location is greater than 0
```

Tests such as these are called non-parametric tests because they do not need strict parametric model assumptions such as Normality. The t-test actually works quite well even under mild departures from Normality, and has better optimality properties (it is more likely to correctly detect situations where H_0 is not true) when the data is actually close to Normal. For this reason, the t-test is preferred when the assumption of Normality is justifiable, even though the Wilcoxon signed-rank test works under more general assumptions.

The two-sample t-test. Consider a more complicated situation where we have independent samples from two different populations: $X_1, X_2, \ldots, X_n \sim F_1$, and $Y_1, Y_2, \ldots, Y_m \sim F_2$. Let the population means of F_1 and F_2 be μ_1 and μ_2. We want to test $H_0 \colon \mu_1 = \mu_2$.

As before, the typical parametric approach is to assume that each $X_i \sim N(\mu_1, \sigma_1^2)$, and each $Y_j \sim N(\mu_2, \sigma_2^2)$. There are two variations of the next step. If we further assume that $\sigma_1^2 = \sigma_2^2$ and use a pooled estimate of the variance, we get the so-called two-sample t-test with equal variance, also called the classical t-test. If we allow $\sigma_1^2 \neq \sigma_2^2$, we can use an alternative procedure due to Welch that produces a test-statistic whose null distribution does not exactly follow a t-distribution, but is well-approximated by it. Both tests can be performed using the `t.test()` function.

Using the energy expenditure data seen before, we can do

```
> data(energy, package = "ISwR")
> s <- with(energy, split(expend, stature))
> t.test(s$lean, s$obese, var.equal = TRUE)

        Two Sample t-test

data:  s$lean and s$obese
t = -3.9456, df = 20, p-value = 0.000799
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -3.411451 -1.051796
sample estimates:
mean of x mean of y
 8.066154 10.297778

> t.test(expend ~ stature, energy, var.equal = FALSE)

        Welch Two Sample t-test

data:  expend by stature
t = -3.8555, df = 15.919, p-value = 0.001411
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -3.459167 -1.004081
sample estimates:
 mean in group lean mean in group obese
           8.066154           10.297778
```

Both these tests have alternative hypothesis $H_1 \colon \mu_1 \neq \mu_2$. The second example uses a formula interface, similar to the one we have seen in lattice graphics. The formula interface is used extensively for modeling in R.

Two-sample Wilcoxon test. As with the one-sample problem, we can compute a two-sample Wilcoxon test statistic, which computes the rank of all observations taken together (without regard to grouping) and then sums the ranks of observations in the first group. The test for $H_1 : \mu_1 < \mu_2$ can be performed using `wilcox.test()` as follows.

```
> wilcox.test(expend ~ stature, energy, alternative = "less")

        Wilcoxon rank sum test with continuity correction

data:  expend by stature
W = 12, p-value = 0.001061
alternative hypothesis: true location shift is less than 0
```

Exercise 5. *Using the ideas described above, compute the p-value for this test using simulation. Use 50000 replications. Is your computed p-value close to the p-value reported by* `wilcox.test()`*?*

Linear models

A wide range of useful statistical models can be viewed as special cases of a class of models called linear models. The general model may be written as

$$y_i = x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{ip}\beta_p + \varepsilon_i, \;\; i = 1, 2, \ldots, n$$

where each i represents one observation in the sample, y_i-s are some response variable that we want to model, and x_{ij}-s are various known predictor or "design" terms. The statistical component in the model comes through the random variables ε_i, which are at a minimum assumed to be independent with mean 0 and constant variance σ^2. Often the more specific model $\varepsilon_i \sim N(0, \sigma^2)$ is assumed.

Estimation. The parameters in a linear model are the *coefficients* $\beta_j, \;\; j = 1, 2, \ldots, p$, and the variance parameter σ^2. The best estimators of the β_j-s turn out to be those that minimise the squared error loss. These estimators have nice analytical solutions, and they can be computed by the R function `lm()`.

Testing. Hypothesis tests under the linear model are conjectures that put *linear* constraints on the β_j coefficients; typically these are of the form $H_0 : \beta_1 = 0$, $H_0 : \beta_1 = \beta_2 = \beta_3 = \beta_4$, etc. Such hypotheses can be tested using a generalization of the t-test known as the F-test, provided we assume the more specific Normal error model.

Specific models. We will not discuss the general linear model theory, but instead just mention some common special cases.

The most familiar example of a linear model is simple linear regression. We will re-use the following artificial example to illustrate regression.

```
> x <- runif(100, min = 1, max = 5)
> mydf <- data.frame(x = x, y = x^2 + 2 * runif(100))
```

A simple linear regression model for this data may be

$$y_i = \beta_1 + x_i\beta_2 + \varepsilon_i, \;\; i = 1, 2, \ldots, n = 100$$

We fit this model in R as

```
> fm1 <- lm(y ~ 1 + x, data = mydf)
> fm1

Call:
lm(formula = y ~ 1 + x, data = mydf)

Coefficients:
(Intercept)            x
     -5.573        5.705
```

We of course know this model to be incorrect, as $E(y_i)$ cannot be expressed as a linear function of x_i. We can fit the correct model by adding a quadratic term.

$$y_i = \beta_1 + x_i\beta_2 + x_i^2\beta_3 + \varepsilon_i,\ i = 1, 2, \ldots, n = 100$$

which is fit by

```
> fm2 <- lm(y ~ 1 + x + I(x^2), data = mydf)
> fm2

Call:
lm(formula = y ~ 1 + x + I(x^2), data = mydf)

Coefficients:
(Intercept)            x       I(x^2)
    1.22003     -0.06778      0.99664
```

In practice we would not know the correct model, so we may need to perform a hypothesis test to decide which model is correct. This is done using

```
> anova(fm1, fm2)

Analysis of Variance Table

Model 1: y ~ 1 + x
Model 2: y ~ 1 + x + I(x^2)
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     98 171.978
2     97  34.123  1    137.86 391.88 < 2.2e-16
```

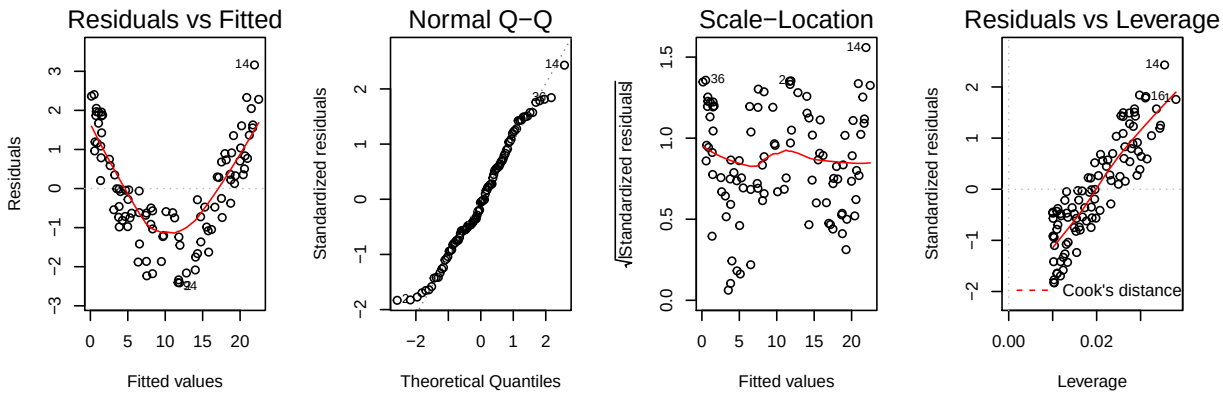
The very small p-value (the column with heading `Pr(>F)`) indicates that `fm2` is a much better fit for the data. Compare this with a test for a model with a (redundant) cubic term:

```
> fm3 <- lm(y ~ 1 + x + I(x^2) + I(x^3), data = mydf)
> anova(fm2, fm3)

Analysis of Variance Table

Model 1: y ~ 1 + x + I(x^2)
Model 2: y ~ 1 + x + I(x^2) + I(x^3)
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
```

```
1     97 34.123
2     96 33.795  1   0.32815 0.9322 0.3367
```

There is not always obvious additional terms to try out to test the goodness of a model fit this way. As an alternative, it is always a good idea to look at graphical diagnostics for a fit. For example, R shows the following plots for `fm1`, the model with only the linear term. The first and last plots are clear indicators of systematic lack-of-fit.

```
> par(mfrow = c(1, 4))
> plot(fm1)
```



Exercise 6. *Look at the corresponding diagnostic plots for* `fm2`*. Do they still indicate lack-of-fit?*

Exercise 7. *Change the call that creates* `mydf` *to*

```
> mydf <- data.frame(x = x, y = x^2 + 2 * rnorm(100))
```

so that the errors have a Normal distribution, rather than the uniform distribution as before. Re-fit `fm2` *and look at its diagnostic plots. Do you see any qualitative difference?*

Exercise 8. *For the* `airquality` *dataset, fit a suitable regression model for* `Ozone` *as response and* `Temp` *as predictor. Comment on goodness-of-fit.*

Another important class of linear models is factorial models. The most simple factorial models are one-way classification models, where the predictor is a categorical variable, and the response is allowed to have a different mean (but the same variance) for each group. One such model with a two-group categorical variable is

```
> fm4 <- lm(expend ~ 1 + stature, energy)
> anova(fm4)
Analysis of Variance Table

Response: expend
          Df Sum Sq Mean Sq F value   Pr(>F)
stature    1 26.485 26.4853  15.568 0.000799
Residuals 20 34.026  1.7013
```

The `anova()` call with a single argument tests submodels obtained by sequentially removing terms from the model, starting with the last one. In this case, the only submodel is equivalent to the hypothesis that both groups have the same mean.

Exercise 9. *In a one-way classification model with two categories, testing equality of means across groups is equivalent to a two-sample t-test with equal variance. Confirm this for the above example by comparing p-values for the appropriate tests.*

An example of a one-way classification model with more than two groups is

```
> fm5 <- lm(Ozone ~ 1 + factor(Month), data = airquality)
> fm5

Call:
lm(formula = Ozone ~ 1 + factor(Month), data = airquality)

Coefficients:
   (Intercept)  factor(Month)6  factor(Month)7  factor(Month)8  factor(Month)9
        23.615           5.829          35.500          36.346           7.833
```

Exercise 10. *Look at the diagnostic plots produced by `plot(fm5)`. Is there any reason to suspect lack-of-fit?*

Exercise 11. *Create a box-and-whisker plot of `Ozone` by `Month`. One-way classification models assume equality of response variance for each group. Is the true for `fm5`? Fit and assess a new model with `log(Ozone)` as the response.*

Any number of continuous (regression-type) and categorical terms can be used in a linear model. Here are a few more complex models for `Ozone`:

```
> airquality <- transform(airquality, fmonth = factor(Month), log.Ozone = log(Ozone))
> fm6 <- lm(log.Ozone ~ 1 + Temp, airquality)
> fm7 <- lm(log.Ozone ~ 1 + fmonth, airquality)
> fm8 <- lm(log.Ozone ~ 1 + Temp + fmonth, airquality)
> fm9 <- lm(log.Ozone ~ 1 + Temp * fmonth, airquality)
```

`fm8` allows a different regression line for each month, but with the same slope. `fm9` allows a different regression line for each month, with possibly different slopes.

Exercise 12. *Use `anova()` to compare all pairs of models for which the comparison makes sense. Which model would you suggest as the most appropriate for the data?*

Further reading

In this tutorial, we have only outlined some very basic ideas of modeling in R. Fitting linear models in R is by itself an extensive topic. `?lm` is a good place to start learning more, especially about the formula language, and many resources on the web have more details. There are also many other kinds of models one can fit in R, which you should explore once you are more comfortable with the basic models.