Probability and Statistics

Exercise sheet 11

Exercise 11.1 Suppose that $X_1, \dots, X_n$ form a random sample from an absolutely continuous distribution for which the probability density function $f_\theta(x)$ under $\mathbb{P}_\theta$ is given by

$$f_\theta(x) = \begin{cases} \theta x^{\theta-1} & \text{for } 0 < x < 1, \\ 0 & \text{otherwise.} \end{cases}$$

Also, suppose that the value of $\theta > 0$ is unknown. Find the MLE (maximum likelihood estimator) for θ.

Solution 11.1 Let

$$L(\theta; x_1, \dots, x_n) := \prod_{i=1}^n \theta x_i^{\theta-1} = \theta^n \left(\prod_{i=1}^n x_i \right)^{\theta-1}.$$

The derivative of $\log L$ with respect to θ is

$$\frac{\partial}{\partial\theta} \left(n \log\theta + (\theta - 1) \log \left(\prod_{i=1}^n x_i \right) \right) = \frac{n}{\theta} + \log \left(\prod_{i=1}^n x_i \right).$$

Setting this to 0 for the sample $X_1, \dots, X_n$ gives

$$\hat\theta = -\frac{n}{\log \left(\prod_{i=1}^n X_i \right)} = -\frac{1}{\left(\overline{\log X}\right)_n},$$

where $\left(\overline{\log X}\right)_n = \frac{1}{n} \sum_{i=1}^n \log X_i$ is a critical point of L. For $\theta < \hat\theta$, $\log L$ is increasing and for $\theta > \hat\theta$, $\log L$ is decreasing. Thus $\hat\theta$ is the global maximum and is the MLE for θ.

Exercise 11.2 Suppose that $X_1, \dots, X_n$ form a random sample from a normal distribution for which both the mean μ and the variance $\sigma^2 > 0$ are unknown.

(a) Find the MLE for $\theta = (\mu, \sigma^2)$.

(b) Find the MLE for $\tilde\theta = (\mu, \sigma)$.

Solution 11.2 The density of X_i is

$$f_{\mu,\sigma^2}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2} \right).$$

The logarithm of the likelihood function is

$$\log L(\mu, \sigma^2; x_1, \dots, x_n) = -\frac{n}{2} \log(2\pi\sigma^2) - \sum_{i=1}^n \frac{(x_i - \mu)^2}{2\sigma^2}.$$

(a) We look for μ and σ^2 which maximize L, which is equivalent to maximizing $\log L$. We compute the partial derivatives and set them to 0; this gives

$$\frac{\partial \log L}{\partial \mu} = \sum_{i=1}^n \frac{X_i - \mu}{\sigma^2} = 0,$$

$$\frac{\partial \log L}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \left(\sum_{i=1}^{n} \frac{(X_i-\mu)^2}{2}\right)\frac{1}{(\sigma^2)^2} = 0.$$

So for any σ^2, $\mu \mapsto \log L(\mu, \sigma^2)$ is maximized for the sample $X_1, \dots, X_n$ when

$$\hat{\mu} = \overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i,$$

which is the sample average of the X_i. For $\mu = \overline{X}_n$, $\sigma^2 \mapsto \log L(\mu, \sigma^2)$ is maximized for the sample $X_1, \dots, X_n$ when

$$\widehat{\sigma^2} = \sum_{i=1}^{n} \frac{(X_i - \overline{X}_n)^2}{n} = \left(\overline{(X_i - \overline{X}_n)^2}\right)_n,$$

which is $\frac{n-1}{n}$ times the sample variance.

(b) The estimator for the first coordinate of $\tilde{\theta}$ is the same $\hat{\mu}$ as in (a). We use the chain rule to calculate

$$\frac{\partial \log L}{\partial \sigma} = \frac{\partial \log L}{\partial \sigma^2}\frac{\partial \sigma^2}{\partial \sigma} = \frac{\partial \log L}{\partial \sigma^2} 2\sigma.$$

This is 0 if and only if $\hat{\sigma}^2 = \widehat{\sigma^2}$. If we were to optimize σ over $\mathbb{R}$, we should obtain two solutions $\hat{\sigma} = \pm\sqrt{\widehat{\sigma^2}}$, but by definition σ is always positive and therefore we obtain $\hat{\sigma} = \sqrt{\widehat{\sigma^2}} = \sqrt{\left(\overline{(X_i - \overline{X}_n)^2}\right)_n}$.

More generally, we can easily show that MLE is invariant under reparametrization. Let $g : \Theta \to \tilde{\Theta}$ be surjective and X_i have the density $f_{i,\theta} = \tilde{f}_{i,g(\theta)}$. Then

$$\tilde{L}(\tilde{\theta}; x_1, \dots, x_n) = \prod \tilde{f}_{i,\tilde{\theta}}(x_i) = \prod \tilde{f}_{i,g(\theta)}(x_i) = \prod f_{i,\theta}(x_i) = L(\theta; x_1, \dots, x_n).$$

This implies that

$$\arg\min_{\tilde{\theta}\in\tilde{\Theta}} \tilde{L}(\tilde{\theta}) = g\left(\arg\min_{\theta\in\Theta} L(\theta)\right).$$

(To be precise, we use the set-valued definition of the $\arg\min_\Theta L := \{\theta \in \Theta : L(\theta) = \min L\}$ as the set of all minimizers. In the case of a bijective g and a uniquely solvable L, one could also use the point-valued definition of the arg min as the unique optimizer.)

Exercise 11.3 We toss a coin 100 times and we get 60 heads. We want to do a test to know whether the coin is fair. We can use the central limit theorem to approximate binomial distributions.

(a) Test the hypothesis with an approximate 0.01 level of significance. Should this test be one or two-sided (in other words, $\Theta_A = \left\{p : p > \frac{1}{2}\right\}$ or $\Theta_A = \left\{p : p \neq \frac{1}{2}\right\}$)? Compute the realized value of the P-value.

(b) What is the largest number of heads we should have in 100 tosses so that we cannot reject $\tilde{H}_0$:= "The coin is not biased towards head". Compute the realized value of the P-value.

Solution 11.3 Let $(X_i)_{i=1}^n$ be an i.i.d. family of Bernoulli(p) random variables. Let $X = \sum_{i=1}^{100} X_i$; then $X \sim \text{Bin}(100, p)$.

(a) We want to know whether the coin is fair or not, so our hypotheses are

$$H_0 : \text{The coin is fair,}$$
$$H_A : \text{The coin is not fair.}$$

That is to say,

$$H_0 : \Theta_0 = \left\{ \frac{1}{2} \right\},$$

$$H_A : \Theta_A = \left\{ p : p \neq \frac{1}{2} \right\}.$$

Then we should use a two-sided test.

Under H_0, we have that $\mathbb{E}_{\frac{1}{2}}[X] = 50$ and $\mathrm{Var}_{\frac{1}{2}}[X] = n\frac{1}{2}(1 - \frac{1}{2}) = 25$. By the central limit theorem, the distribution of $\frac{X-50}{5}$ can be approximated by the standard normal distribution. Let Φ be the c.d.f. of the standard normal distribution. Now, we want to find c_1 and c_2 such that

$$\begin{aligned} 0.01 &\geq \mathbb{P}_{\frac{1}{2}}[X \notin (c_1, c_2)] \\ &= 1 - \mathbb{P}_{\frac{1}{2}}\left[\frac{c_1 - 50}{5} < \frac{X - 50}{5} < \frac{c_2 - 50}{5} \right] \\ &\approx 1 - \Phi\left(\frac{c_2 - 50}{5} \right) + \Phi\left(\frac{c_1 - 50}{5} \right). \end{aligned}$$

We should like to make the rejection zone as large as possible to maximize the power of the test. Given that Φ is symmetric, we choose

$$\frac{c_1 - 50}{5} = -\frac{c_2 - 50}{5},$$

which means that $c_1 = 100 - c_2$. Finally, we use the same approximation and now solve

$$0.01 \geq 1 - \Phi\left(\frac{c_2 - 50}{5} \right) + 1 - \Phi\left(\frac{c_2 - 50}{5} \right)$$

to get

$$\Phi\left(\frac{c_2 - 50}{5} \right) \geq 0.995 = \Phi^{\leftarrow}(0.995)$$

or

$$\frac{c_2 - 50}{5} \geq 2.5758$$

to find

$$c_2 \geq 62.9.$$

So the rejection region at the (approximate) significance level $\alpha = 0.01$ is

$$K_{1\%} = [0; 37] \cup [63; 100].$$

Given that $60 \notin K_{1\%}$, we cannot reject H_0 with 0.01 level of significance.

The approximate realized value of the P-value is

$$\pi(60) \approx 1 - \Phi\left(\frac{60 - 50}{5} \right) + 1 - \Phi\left(\frac{60 - 50}{5} \right) = 0.0455.$$

This means that we could reject H_0 at the level $\alpha = 5\%$.

(b) We have to take our hypotheses

$$\tilde{H}_0 : \tilde{\Theta}_0 = \left\{ p : p \leq \frac{1}{2} \right\},$$

$$\tilde{H}_A : \tilde{\Theta}_A = \left\{ p : p > \frac{1}{2} \right\}.$$

Now we have to find c such that

$$0.01 \geq \mathbb{P}_{\frac{1}{2}}[X > c] \approx 1 - \Phi\left(\frac{c-50}{5}\right).$$

Then we have to choose $c \geq 61.6$, from which we get $K_{1\%} = [62; 100]$. So we reject H_0 if we have 62 or more heads. The approximate realized value of the P-value is

$$\tilde{\pi}(60) \approx 1 - \Phi\left(\frac{60-50}{5}\right) = 0.02275.$$

Exercise 11.4 We want to study the `starsCYG`-dataset. Run the following **R**-code. (You can either install **R** and **R**-Studio or another **R**-environment or run it in your browser `https://www.kaggle.com/jakobheiss/probstat2020-ex11-4/edit` using a free kaggle-account or alternatively create a free RStudio Cloud account). (You could also use Anaconda to install R and RStudio.).

```
library(ggplot2);
library(reshape2);
library(robustbase);
library(MASS);
library(quantreg);


data(starsCYG);

data <- data.frame(log.Te = starsCYG$log.Te,
         LS = lm(log.light ~ ., data = starsCYG)$fitted.values,
         L1 = rq(log.light ~ ., data = starsCYG)$fitted.values,
         #M = rlm(log.light ~ ., data = starsCYG, method = "M")$fitted.values,
         S = lqs(log.light ~ ., data = starsCYG, method = "S")$fitted.values,
         MM = lmrob(log.light ~ ., data = starsCYG)$fitted.values,
         LMS = lqs(log.light ~ ., data = starsCYG, method = "lms")$fitted.values,
         LTS = ltsReg(log.light ~ ., data = starsCYG)$fitted.values
)

ggplot(melt(data, id.vars = "log.Te", variable.name = "Regression", value.name = "log.light"),
        aes(x = log.Te, y = log.light, color = Regression)) +
  geom_point(data = starsCYG, color = "#222222") +
  geom_line();
```

Each observation corresponds to a star. On the x-axis, the log of the temperature is plotted, and the y-axis corresponds to the brightness of the stars (log of the light-intensity). We want to find a (affine) linear regression line (see eq. (5.2.1) in the lecture notes) that helps us to understand the relationship between these two quantities. In the given code, there are 6 different methods applied to obtain a regression line. `LS` (least squares; see Example 5.2.3) and `L1` (L_1; see Example 6.2.3 in the lecture notes) have already been defined in the lecture. The other four methods `S`, `MM`, `LMS` and `LTS` are more advanced and you don't have to understand them in detail. These four algorithm all follow the goal to find a line that nicely fits through the majority of points instead of trying to fit all data points. The main idea of `LMS` is to minimize the median of the squared vertical distances between the line and the data points. Instead of the standard least squares method, which minimizes the sum of squared residuals over n points, the `LTS` (least trimmed squares) method attempts to minimize the sum of squared residuals over a subset consisting of h points. The unused $n-h$ points do not influence the fit. The main idea of `LTS` is quite well explained at `https://en.wikipedia.org/wiki/Least_trimmed_squares` (the **R**-implementation of `LTS` is a bit more advanced than that. If you want to test the basic version of LTS, you have to extract `raw.coefficients` from `LTS` and look at `h.alpha.n`). The focus of this exercise lies on having an interesting discussion in the exercise class and understanding intuitively the concept of outliers and breakpoints. You don't have to write down a perfect solution.

(a) Which of those stars are special (in other words, which follow a different pattern than the majority, or which are outliers)?

(b) In this example, one could solve (a) simply by looking at this two-dimensional plot. In high-dimensional situations, one cannot visualize the data so easily. Therefore, one needs algorithms that automatically detect (potential) outliers. One could have the idea to suspect those points with high vertical distance to the LS-regression-line to be the outliers. Do you think this is a good approach (in this example)? Do you have a better idea?

(c) Which of these algorithms have a breakpoint high enough for this problem?

Solution 11.4

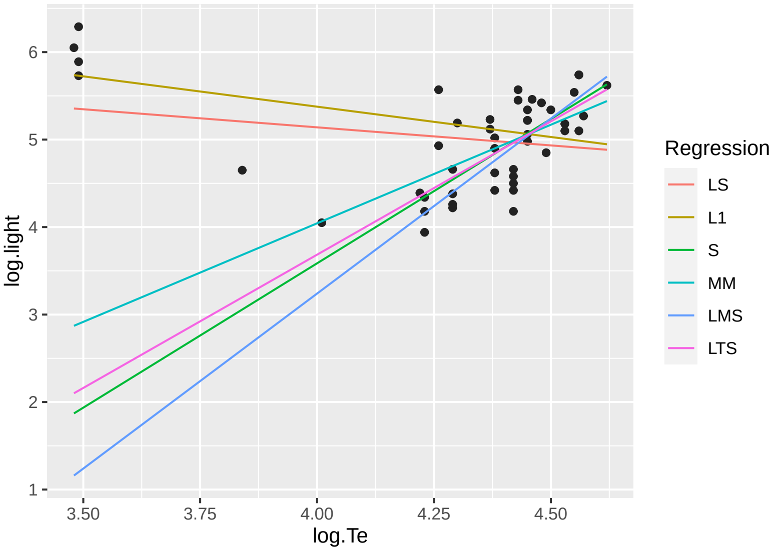

Figure 1: The plot outputted by the given code.

(a) The four data points in the upper left corner are outliers. These stars are called *giants*.

(b) No, this is a very bad idea (in this example, and it is usually not optimal). It is much more reasonable to use a robust fit (e.g. LTS) and suspect those points as outliers that are vertically especially far away from the robust regression line. This would for example in the case LTS find the correct outliers in this example.

(c) LS and L1 both have the breakpoint $\varepsilon^* = 0$. This means that it would be possible to place a single data point at a particular position to disturb the estimator arbitrarily much. So in this example, the four outliers can easily destroy these estimators. (Note that the L_1-estimator is robust with respect to y-outliers, but not with respect to leverage points. In the starsCYG-dataset, the four outliers are leverage points, because they are far away from the center of the other points on the x-axis.)

The field of statistics dealing with outliers and breakpoint-robust estimators is called *robust statistics*. S, MM, LMS and LTS are more advanced algorithms that are particuarly desigend to be breakpoint-robust. You can see in this example that they all succeed in fitting the trend of the majority of the data without fitting the outliers.

If you have feedback regarding the exercise sheets, please send a mail to Jakob Heiss.