Confidence Intervals for the Mean: Known vs Unknown Standard Deviation

Levi Lopes de Lima

2025-12-02

Explaning the lab

In this lab we construct two types of confidence intervals for the mean of a normal population:

1. A z-interval assuming the population standard deviation is **known**.
2. A t-interval assuming the population standard deviation is **unknown**.

In order to make the codes reusable, we wrap them in certain R functions depending on the relevant parameters (`ci_mean_known_sigma` and `ci_mean_unknown_sigma_2`), which are then applied to a couple of problems appearing in (Dekking 2005), to which we refer for the theoretical details; see also (Lima 2025) for further discussions on the subject.

A glimpse at the theory I

Recall that a random variable X is said to follow a normal (notation: $X \sim \mathcal{N}(\mu, \sigma^2)$, where $\mu \in \mathbb{R}$ and $\sigma^2 > 0$) if it is continuous (in the sense that its distribution P_X is absolutely continuous with respect to Lebesgue measure: $dP_X = \psi_X dx$) and its **probability density function** $\psi_X = \psi_{(\mu,\sigma^2)}$ is given by

$$\psi_{(\mu,\sigma^2)}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad x \in \mathbb{R}.$$

We recall that $\mathbb{E}(X) = \mu$ and $\mathrm{var}(X) = \sigma^2$ so the shape of $\psi_{(\mu,\sigma^2)}$, as member of this family of densities, is completely determined by these population parameters.

We will be interested in estimating the population mean μ starting from a random sample $\{X_j\}_{j=1}^{+\infty}$ drawn from a normal population as above. As usual, we take the **sample mean**,

$$\overline{X}_n = \frac{X_1 + \cdots + X_n}{n},$$

as the corresponding estimator, and will make use of the **pivotal method**, which in this case relies on the following well-known fact:

- If, for $j = 1, \ldots, k$, $Y_j \sim \mathcal{N}(\mu_j, \sigma_j^2)$ are independent, $a_j \in \mathbb{R}$ and $b \in \mathbb{R}$ then

$$\sum_j a_j Y_j + b \sim \mathcal{N}\left(\sum_j a_j \mu_j + b, \sum_j a_j^2 \sigma_j^2\right).$$

In words, any affinne combination of independent normal random variables follows a normal with the anticipated expectation and variance.

Using this we easily check that that the **standardization**

$$\overline{Z}_n = \frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}}$$

of $\overline{X}_n$ is a **pivotal quantity** in the sense that its distribution does *not* depend on the estimated parameter μ, as it follows a standard normal variation:

$$\overline{Z}_n \sim \mathcal{N}(0, 1).$$

Hence, for any $0 < \alpha < 1$ there holds

$$P\left(-z_{\alpha/2} \leq \overline{Z}_n \leq z_{\alpha/2}\right) = 1 - \alpha,$$

where

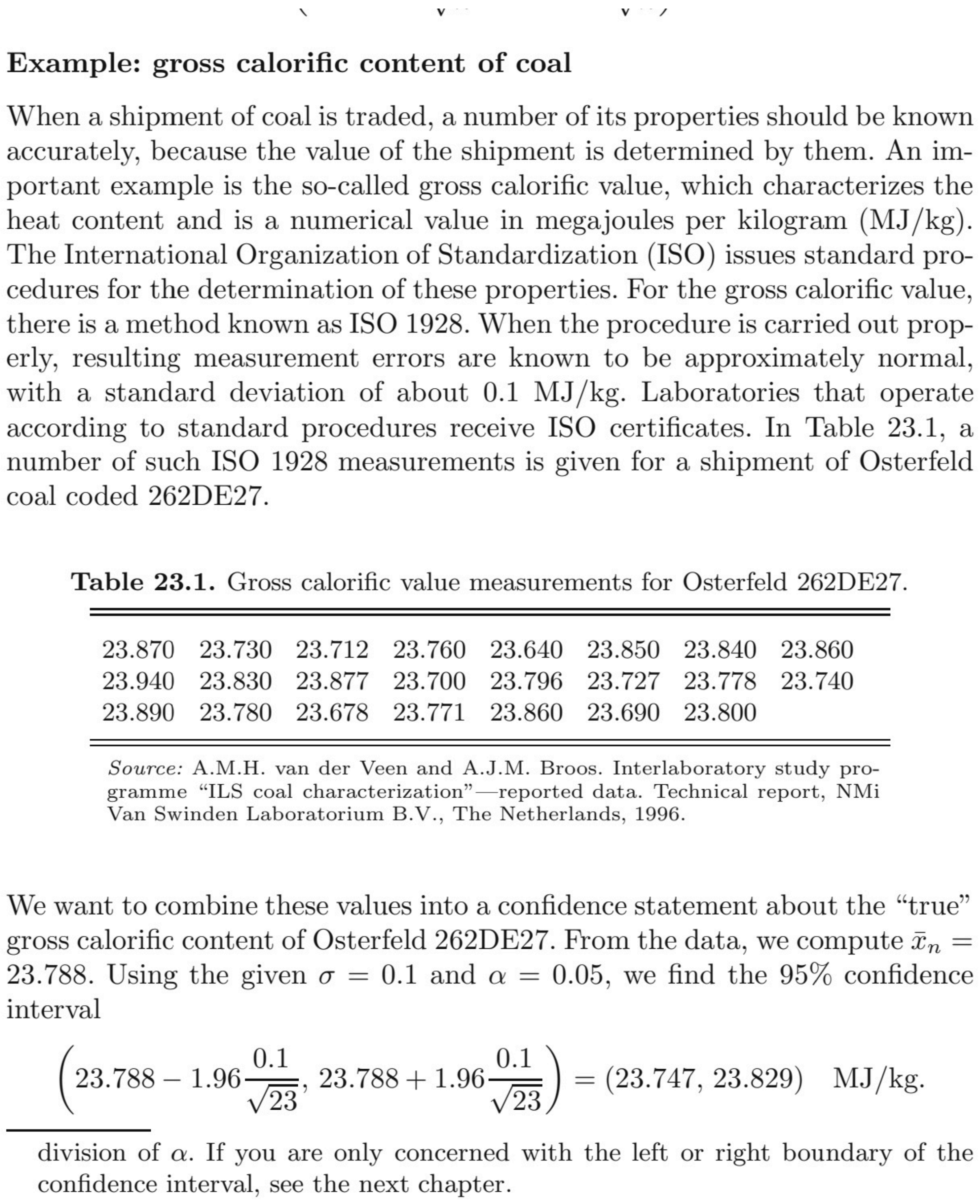
$$z_{\alpha/2} = \Phi^{-1}\left(1 - \frac{\alpha}{2}\right), \quad \Phi(x) = \int_{-\infty}^{x} \psi_{(0,1)}(t)dt$$

is the **normal quantile** associated to the **confidence level** $1-\alpha$. Of course, we may rewrite this as a confidence interval (CI) for μ:

$$\mu \in \left[\overline{X}_n \mp z_{\delta/2}\frac{\sigma}{\sqrt{n}}\right] \quad \text{with prob.} = 1-\alpha.$$

We insist that this **z-interval** is an exact (not approximate) CI which holds true for samples of any size n, its only drawback being that its size depends on the (generally unknown!) standard deviation σ.

The first problem (as taken from (Dekking 2005), page 347)



Example: gross calorific content of coal

When a shipment of coal is traded, a number of its properties should be known accurately, because the value of the shipment is determined by them. An important example is the so-called gross calorific value, which characterizes the heat content and is a numerical value in megajoules per kilogram (MJ/kg). The International Organization of Standardization (ISO) issues standard procedures for the determination of these properties. For the gross calorific value, there is a method known as ISO 1928. When the procedure is carried out properly, resulting measurement errors are known to be approximately normal, with a standard deviation of about 0.1 MJ/kg. Laboratories that operate according to standard procedures receive ISO certificates. In Table 23.1, a number of such ISO 1928 measurements is given for a shipment of Osterfeld coal coded 262DE27.

Table 23.1. Gross calorific value measurements for Osterfeld 262DE27.

23.870	23.730	23.712	23.760	23.640	23.850	23.840	23.860
23.940	23.830	23.877	23.700	23.796	23.727	23.778	23.740
23.890	23.780	23.678	23.771	23.860	23.690	23.800	

Source: A.M.H. van der Veen and A.J.M. Broos. Interlaboratory study programme "ILS coal characterization"—reported data. Technical report, NMi Van Swinden Laboratorium B.V., The Netherlands, 1996.

We want to combine these values into a confidence statement about the "true" gross calorific content of Osterfeld 262DE27. From the data, we compute $\bar{x}_n = 23.788$. Using the given $\sigma = 0.1$ and $\alpha = 0.05$, we find the 95% confidence interval

$$\left(23.788 - 1.96\frac{0.1}{\sqrt{23}},\, 23.788 + 1.96\frac{0.1}{\sqrt{23}}\right) = (23.747,\, 23.829) \quad \text{MJ/kg.}$$

division of α. If you are only concerned with the left or right boundary of the confidence interval, see the next chapter.

Since $\sigma = 0.1$ is known, we may use the z-interval above to check the computation above.

Reading the data (coal)

After extracting the data set in table above to a csv file…

```
dados <- read.csv("coal.csv")
```

…we may view its structure…

```
str(dados)
```

```
## 'data.frame':    23 obs. of  2 variables:
##  $ id         : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ measurement: num  23.9 23.7 23.7 23.8 23.6 ...
```

…and then print it for comparison:

```
print(dados)
```

```
##    id measurement
## 1   1      23.870
## 2   2      23.730
## 3   3      23.712
## 4   4      23.760
## 5   5      23.640
## 6   6      23.850
## 7   7      23.840
## 8   8      23.860
## 9   9      23.940
## 10 10      23.830
## 11 11      23.877
## 12 12      23.700
## 13 13      23.796
## 14 14      23.727
## 15 15      23.778
## 16 16      23.740
## 17 17      23.890
## 18 18      23.780
## 19 19      23.678
## 20 20      23.771
## 21 21      23.860
## 22 22      23.690
## 23 23      23.800
```

It is convenient to store the data set as a vector for further use:

```
x <- dados$measurement
length(x)
```

```
## [1] 23
```

```
summary(x)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   23.64   23.73   23.78   23.79   23.86   23.94
```

Constructing a z-interval for the coal dataset (known σ)

As already remarked, the code will be wrapped in a R function with three parameters: the data set vector, the (known) standard deviation (σ) and $\alpha = 1 - \text{confidence level.}$

```
#building a function for computing a z-interval (known sigma)
ci_mean_known_sigma <- function(x, sigma, alpha = 0.05) {
  # sample size and mean
  n     <- length(x)
  meanx <- mean(x)
  # standard error under known sigma
  se <- sigma / sqrt(n)
  # normal critical value
  zcrit <- qnorm(1 - alpha/2)
  # margin of error
  margin <- zcrit * se
  # interval endpoints
  lower <- meanx - margin
  upper <- meanx + margin
  # return as a list
  return(list(
    n        = n,
    mean     = meanx,
    se       = se,
    z_crit   = zcrit,
    lower    = lower,
    upper    = upper
  ))
}
```

We now save the lists of the function evaluated at $\sigma = 0.1$ with three choices for the confidence level: 92%, 95% and 98%.

```
# saving the evaluated function
ci_known_coal_90 <- ci_mean_known_sigma(x,0.1,0.1) # 90%
ci_known_coal <- ci_mean_known_sigma(x,0.1,0.05) # 95% (the standard choice)
ci_known_coal_98 <- ci_mean_known_sigma(x,0.1,0.02) # 98%
str(ci_known_coal)
```

```
## List of 6
##  $ n     : int 23
##  $ mean  : num 23.8
##  $ se    : num 0.0209
##  $ z_crit: num 1.96
##  $ lower : num 23.7
##  $ upper : num 23.8
```

The result may be displayed in a table (with some of the items in the list omitted for convenience):

```
ci_table_coal <- data.frame(
  Method        = c("z-interval (known sigma at 90%)","z-interval (known sigma at 95%)","z-interval (known sigma at 98%)"),
 # SD_or_sigma   = round(c(sigma_known, sd_hat), 4),
  #SE            = round(c(se_known, se_hat), 4),
  #CriticalValue = round(c(z_crit, t_crit), 4),
  CI_Lower      = round(c(ci_known_coal_90$lower,ci_known_coal$lower,ci_known_coal_98$lower), 4),
  Mean          = round(c(ci_known_coal_90$mean,ci_known_coal$mean,ci_known_coal_98$mean), 4),
  CI_Upper      = round(c(ci_known_coal_90$upper,ci_known_coal$upper,ci_known_coal_98$upper), 4)
)
knitr::kable(ci_table_coal, caption = "Confidence Interval for the Coal Problem (known sigma)")
```

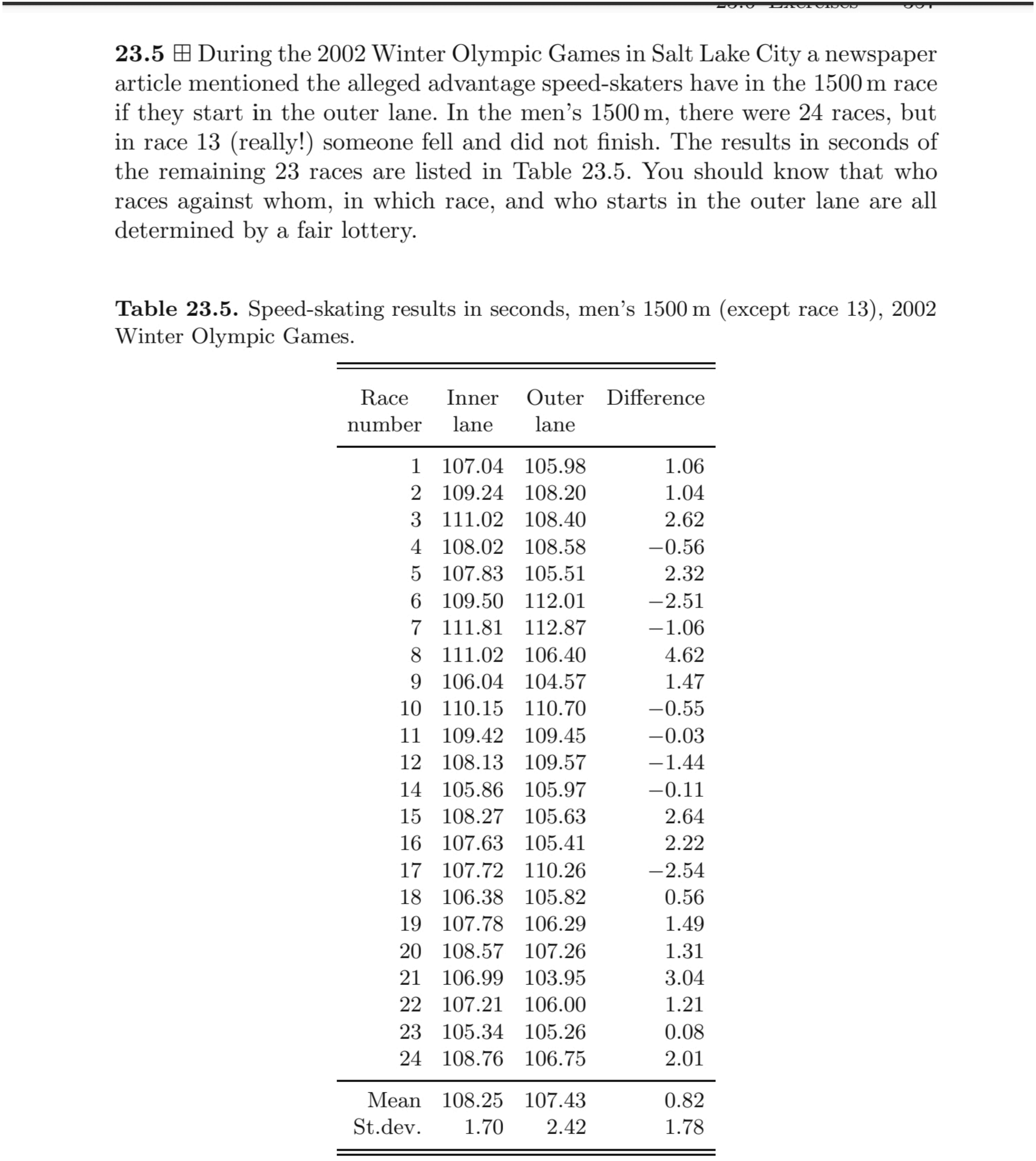
Confidence Interval for the Coal Problem (known sigma)

Method	CI_Lower	Mean	CI_Upper
z-interval (known sigma at 90%)	23.7535	23.7878	23.8221
z-interval (known sigma at 95%)	23.7469	23.7878	23.8287
z-interval (known sigma at 98%)	23.7393	23.7878	23.8363

As expected, the CIs get larger with the confidence level. Also, the result for 95% matches the computation in the image above.

The problem of the races (taken from (Dekking 2005), page 357)

We now turn to another (proposed) problem in (Dekking 2005).



23.5 ⊞ During the 2002 Winter Olympic Games in Salt Lake City a newspaper article mentioned the alleged advantage speed-skaters have in the 1500 m race if they start in the outer lane. In the men's 1500 m, there were 24 races, but in race 13 (really!) someone fell and did not finish. The results in seconds of the remaining 23 races are listed in Table 23.5. You should know that who races against whom, in which race, and who starts in the outer lane are all determined by a fair lottery.

Table 23.5. Speed-skating results in seconds, men's 1500 m (except race 13), 2002 Winter Olympic Games.

Race number	Inner lane	Outer lane	Difference
1	107.04	105.98	1.06
2	109.24	108.20	1.04
3	111.02	108.40	2.62
4	108.02	108.58	−0.56
5	107.83	105.51	2.32
6	109.50	112.01	−2.51
7	111.81	112.87	−1.06
8	111.02	106.40	4.62
9	106.04	104.57	1.47
10	110.15	110.70	−0.55
11	109.42	109.45	−0.03
12	108.13	109.57	−1.44
14	105.86	105.97	−0.11
15	108.27	105.63	2.64
16	107.63	105.41	2.22
17	107.72	110.26	−2.54
18	106.38	105.82	0.56
19	107.78	106.29	1.49
20	108.57	107.26	1.31
21	106.99	103.95	3.04
22	107.21	106.00	1.21
23	105.34	105.26	0.08
24	108.76	106.75	2.01
Mean	108.25	107.43	0.82
St.dev.	1.70	2.42	1.78

We quote from the original source:

- As a consequence of the lottery and the fact that many different factors contribute to the actual time difference "inner lane minus outer lane" the assumption of a normal distribution for the difference is warranted. The numbers in the last column can be seen as realizations from an $\mathcal{N}(\mu, \sigma^2)$. distribution, where μ is the expected outer lane advantage. *Construct a 95% confidence interval for* μ. N.B. $n = 23$, not 24!

A glimpse at the theory II

Since now the standard deviation σ is unknown, one must somehow estimate it. From the Law of Large Numbers and the fundamental identity

$$\sum_j (X_j-\mu)^2 = \sum_j (X_j-\overline{X}_n)^2 + n(\overline{X}_n-\mu)^2$$

we easily check that the **sample variance**

$$S_n^2 = \frac{1}{n-1}\sum_j (X_j-\overline{X}_n)^2$$

satisfies:

- $\mathbb{E}(S_n^2)=\sigma^2$ (which means that S_n^2 is **unbiased** for σ^2);
- $S_n^2 \overset{p}{\to} \sigma^2$ (which means that S_n^2 is **consistent** for σ^2).

In particular, $S_n \overset{p}{\to} \sigma$ and S_n is **consistent** for the standard deviation σ. It follows that the **studentization**

$$T_{n-1} := \frac{\overline{X}_n-\mu}{S_n/\sqrt{n}},$$

which is obtained from the padronization $\overline{Z}_n$ of $\overline{X}_n$ by the replacement $\sigma \rightsquigarrow S_n$, satisfies

$$T_{n-1} = \frac{\sigma}{S_n}\overline{Z}_n \overset{d}{\to} \mathcal{N}(0,1),$$

which immediately yields an approximate **large sample** CI for μ:

$$\mu \in \left[\overline{X}_n \mp z_{\delta/2}\frac{S_n}{\sqrt{n}}\right] \quad \text{with prob.} \approx 1-\alpha,$$

but of course this does **not** apply to the problem at hand (because a sample with $n=23$ fails to qualify as being large!). Taking into account that

$$T_{n-1} = \frac{\frac{\overline{X}_n-\mu}{\sigma/\sqrt{n}}}{\sqrt{\frac{\frac{n-1}{\sigma^2}S_n^2}{n-1}}},$$

the explicit determination of the distribution of T_{n-1} depends on the following facts (both relying heavily on the assumption that the underlying population is normal):

- $\frac{\overline{X}_n-\mu}{\sigma/\sqrt{n}} \sim \mathcal{N}(0,1)$;
- $\frac{n-1}{\sigma^2}S_n^2 \sim \chi^2_{n-1}$, the **chi-square** distribution of $n-1$ degrees of freedom.

The key step now is to combine this with a fundamental result due to W. S. Gosset:

- For any $k\geq 1$, if $Z\sim\mathcal{N}(0,1)$ and $W\sim\chi^2_k$ are independent then

  $$T_k := \frac{Z}{\sqrt{W/k}} = t_k dx,$$

  where

  $$t_k(x) = \frac{\Gamma(\frac{k+1}{2})}{\sqrt{\pi k}\Gamma(\frac{k}{2})}\left(1+k^{-1}x^2\right)^{-\frac{k+1}{2}}.$$

  In words, T_k follows a **Student distribution** with k degrees of freedom[1].

Leading this to the above expression for T_{n-1}, we immediately obtain a CI for μ which applies to the cases in which σ is unknown:

$$\mu \in \left[\overline{X}_n \mp t_{n-1,\alpha/2}\frac{S_n}{\sqrt{n}}\right] \quad \text{with prob.} = 1-\alpha,$$

where $t_{n-1,\delta/2}$ is the **Student quantil** associated to the confidence level $1-\alpha$. Note that this **t-interval** is not only exact but also applies to samples of **any** size. As we shall see below, it applies to the “races problem” mentioned above.

Reading the data (races)

As usual, we start by reading the data from a csv file…

```
dados_2 <- read.csv("races.csv")
str(dados_2)
```

```
## 'data.frame':    23 obs. of  4 variables:
##  $ race      : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ inner_lane: num  107 109 111 108 108 ...
##  $ outer_lane: num  106 108 108 109 106 ...
##  $ difference: num  1.06 1.04 2.62 -0.56 2.32 -2.51 -1.06 4.62 1.47 -0.55 ...
```

```
head(dados_2)
```

```
##   race inner_lane outer_lane difference
## 1    1     107.04     105.98       1.06
## 2    2     109.24     108.20       1.04
## 3    3     111.02     108.40       2.62
## 4    4     108.02     108.58      -0.56
## 5    5     107.83     105.51       2.32
## 6    6     109.50     112.01      -2.51
```

…and saving the results in the last column (`difference`) as a vector for further use:

```
y <- dados_2$difference
length(y)
```

```
## [1] 23
```

```
summary(y)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -2.5400 -0.3300  1.0600  0.8213  2.1150  4.6200
```

Constructing t-interval for the races data set (unknown σ)

We now proceed to the relevant code, as always wrapped in a function depending on two (adjustable) parameters: the data set vector and $\alpha = 1 - \text{confidence level}$:

```
#building a function for computing a t-interval
ci_mean_unknown_sigma_2 <- function(y, alpha = 0.05) {
  # sample size and mean
  m     <- length(y)
  meany <- mean(y)
  # sample standard deviation
  sd_hat_2 <- sd(y)
  # standard error
  se_2 <- sd_hat_2 / sqrt(m)
  # Student t critical value
  tcrit <- qt(1 - alpha/2, df = m - 1)
  # margin of error
  margin <- tcrit * se_2
  # interval endpoints
  lower <- meany - margin
  upper <- meany + margin
  # return as a list
  return(list(
    m        = m,
    mean     = meany,
    sd_hat   = sd_hat_2,
    se       = se_2,
    t_crit   = tcrit,
    lower    = lower,
    upper    = upper
  ))
}
```

We now evaluate the function at two confidence levels, save the resulting lists as vectors…

```
#saving the evaluated function
ci_unknown_2_95 <- ci_mean_unknown_sigma_2(y,0.05) # 95% (the standard choice)
ci_unknown_2_98 <- ci_mean_unknown_sigma_2(y,0.02) # 98%
```

…which are then displayed in a table:

```
ci_table_skates <- data.frame(
  Method        = c("t-interval for races (unknown sigma at 95%)", "t-interval for races (unknown sigma at 98%)"),
 SD_or_sigma   = round(c(ci_unknown_2_95$sd_hat,ci_unknown_2_98$sd_hat), 4),
  #SE            = round(c(se_known, se_hat), 4),
  #CriticalValue = round(c(z_crit, t_crit), 4),
  CI_Lower      = round(c(ci_unknown_2_95$lower,ci_unknown_2_98$lower), 4),
 Mean          = round(c(ci_unknown_2_95$mean,ci_unknown_2_98$mean), 4),
  CI_Upper      = round(c(ci_unknown_2_95$upper,ci_unknown_2_98$upper), 4)
)
knitr::kable(ci_table_skates, caption = "Confidence Interval for races (unknown sigma)")
```

Confidence Interval for races (unknown sigma)

Method	SD_or_sigma	CI_Lower	Mean	CI_Upper
t-interval for races (unknown sigma at 95%)	1.7828	0.0504	0.8213	1.5922
t-interval for races (unknown sigma at 98%)	1.7828	-0.1111	0.8213	1.7537

Since the t-interval in the first row fails to include 0, we may conclude that, at the 95% confidence level, the speed skaters in the outer lanes appear to have a real advantage. Note, however, that according to the t-interval in the second row, this statistical evidence disappears at the 98% confidence level. Thus, we may obtain seemingly conflicting results depending on the chosen confidence level, but there is no contradiction here, since this choice is inherently subjective.

A final comparison

Here we pretend that the estimated value for the standard deviation $\sigma \approx 1.78$ above is the true population value σ and use it to compute the corresponding z-interval by means of the function `ci_mean_known_sigma()` previously constructed.

```
ci_est<-ci_mean_known_sigma(y,1.78,0.05)
print(ci_est)
```

```
## $n
## [1] 23
##
## $mean
## [1] 0.8213043
##
## $se
## [1] 0.3711557
##
## $z_crit
## [1] 1.959964
##
## $lower
## [1] 0.09385263
##
## $upper
## [1] 1.548756
```

… so we may now compare the intervals (both at a 95% level of confidence)…

```
ci_table_comp <- data.frame(
  Method        = c("t-interval (unknown sigma)",
                    "z-interval (estimated sigma)"),
 # SD_or_sigma   = round(c(ci_unknown_2$sd_hat, ci_est$sd_hat), 4),
  #SE            = round(c(ci_unknown_2$se, ci_est$se_hat), 4),
  #CriticalValue = round(c(z_crit, t_crit), 4),
  CI_Lower      = round(c(ci_unknown_2_95$lower, ci_est$lower), 4),
  Mean          = round(c(ci_unknown_2_95$mean, ci_est$mean), 4),
  CI_Upper      = round(c(ci_unknown_2_95$upper, ci_est$upper), 4)
)

knitr::kable(ci_table_comp, caption = "Comparison of Confidence Intervals for races (unknown and estimated sigma)")
```

Comparison of Confidence Intervals for races (unknown and estimated sigma)

Method	CI_Lower	Mean	CI_Upper
t-interval (unknown sigma)	0.0504	0.8213	1.5922
z-interval (estimated sigma)	0.0939	0.8213	1.5488

… to check that the z-interval is a bit smaller. This only confirms that the Student’s distribution has a slightly heavier tail than the normal, which implies that $t_{n-1,\delta/2} > z_{\delta/2}$ for any choice of δ, with $t_{n-1,\delta/2} \to z_{\delta/2}$ as $n \to +\infty$.

Dekking, Frederik Michel. 2005. *A Modern Introduction to Probability and Statistics: Understanding Why and How*. Springer Science & Business Media.

Lima, Levi Lopes de. 2025. “Topics in Probability, Parametric Estimation and Stochastic Calculus.” https://arxiv.org/abs/2510.20163.

1. The name “Student” is a pseudonym used by William Sealy Gosset, a noted British statistician, in his scientific paper publications during his work at the Guinness Brewery in Dublin, Ireland.↩