Statistical programming

Inferential statistics

Marc Comas-Cufí

Today’s session

- Overview of probability and statistics
  - Chi-squared test: goodness of fit, independence
  - Student’s t-test: one sample, paired samples, independent two-sample
  - One-way ANOVA test
  - Normality test
  - Homocedasticity test

Statistical inference

Objective

Given a sample $X = \{x_1, \ldots, x_N\}$, we are interested in analysing a model $\text{Model}(\theta)$, $\theta \in \Theta$, under the assumption that x_i's are independent and identically distributed, i.e. $x_i \sim \text{Model}(\theta)$.

Examples

- Given $X = \{x_1, \ldots, x_N\}$ numeric, ... *study* $E[x_i]$*, assuming* x_i*'s follow the same numerical distribution* $f(\theta)$.
- Given $X = \{x_1, \ldots, x_N\}$ binary 0/1, ... *study* $E[x_i]$*, assuming* x_i*'s follow a* $Bin(n = 1, \pi)$.

Types of inference

- **Confidence intervals**. Find θ_- and θ_+ such that certain $\mathrm{P}(\theta_- \leq \theta \leq \theta_+)$ holds.
- **Hypothesis testing**. Is a certain hypothesis, H_0, about $\mathrm{Model}(\theta)$ plausible?

Confidence intervals

About confidence intervals

- Confidence interval are regions where we think certain parameters θ is.
- For a level of confidence $1 - \alpha$ (we should think α small), we talk about $(1 - \alpha)100\%$ confidence interval (CI) for parameter θ.
- A $(1 - \alpha)100\%$ CI for θ is a region between θ_- and θ_+ such that

$$\mathrm{P}(\theta_- \leq \theta \leq \theta_+) = 1 - \alpha.$$

- We can have confidence intervals for different types of parameters θ:
  - For the expected value of a random variable: **mean**, μ, if continuous or **proportion**, π, if dichotomous (binomial with $n = 1$).
  - For coefficients/parameters of a model.

Confidence intervals for proportions

Let's assume $X \sim Bin(n = 1, \pi)$, for example:

```
X1 = c(0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0)
```

- What is our best guess for π?
- How can we measure our uncertainty about π?

Suppose we have:

```
X2 = c(0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1,
       0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0)
```

- If we decide to use this new sample, how it will affect to our uncertainty of π?

Bootstrap

- Bootstrap is a simple and powerful tool that, among others, is useful to build confidence intervals for any type of parameter.
- The method consists in resampling from the original sample multiple times **using replacement**.

```r
v_proportions = replicate(10000, mean(sample(X1, length(X1), replace = TRUE)))

# Summary the vector of means
summary(v_proportions)
#>    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#>  0.0000  0.1500  0.2000  0.1998  0.2500  0.6000
```

- A 95%-CI for the proportion using sample X1 would be:

```r
alpha = 0.05
quantile(v_proportions, c(alpha/2, 1-alpha/2))
#>  2.5% 97.5%
#>  0.05  0.40
```

Asymptotic results

If $X \sim Bin(n = 1, \pi)$ and N is high enough,

$$\bar{X} \sim N(\mathbb{E}[X], \sqrt{\text{var}[X]/N}) \iff \pi = \mathbb{E}[X] \sim N(\bar{X}, \sqrt{\pi(1-\pi)/N}).$$

- Assuming $\pi(1-\pi) \approx \bar{X}(1-\bar{X})$, we know how π is distributed,

$$\pi = \mathbb{E}[X] \sim N(\bar{X}, \sqrt{\bar{X}(1-\bar{X})/N})$$

- A 95%-CI for the proportion using sample X1 would be:

```
p = mean(X1)
N = length(X1)
qnorm(c(alpha/2, 1-alpha/2), p, sqrt(p*(1-p)/N))
#> [1] 0.02469549 0.37530451

# Equivalently,
# p + qnorm(c(alpha/2, 1-alpha/2)) * sqrt(p*(1-p)/N) or
# p + c(1,-1) * qnorm(alpha/2) * sqrt(p*(1-p)/N)
```

Sample size and uncertainty

- The higher the sample the lower the uncertainty. $N_1 = 20, N_2 = 100$.

Bootstrap

```r
replicate(10000, mean(sample(X1, length(X1), replace = TRUE))) %>%
  quantile(c(alpha/2, 1-alpha/2))
#>  2.5% 97.5%
#>  0.05  0.40
replicate(10000, mean(sample(X2, length(X2), replace = TRUE))) %>%
  quantile(c(alpha/2, 1-alpha/2))
#>  2.5% 97.5%
#>  0.16  0.32
```

Asymptotic aproximation

```r
N1 = length(X1); N2 = length(X2)
p1 = mean(X1); p2 = mean(X2)
qnorm(c(alpha/2, 1-alpha/2), p1, sqrt(p1*(1-p1)/N1))
#> [1] 0.02469549 0.37530451
qnorm(c(alpha/2, 1-alpha/2), p2, sqrt(p2*(1-p2)/N2))
#> [1] 0.1562932 0.3237068
```

Confidence interval for means

Given a numeric sample:

```
X = c(11.2, 10.28, 10.24, 10.69, 8.34, 11.06, 9.53, 10.6, 8.78, 9.16,
      9.52, 10.63, 10.18, 9.06, 9.87, 11.18, 9.91, 9.26, 9.25, 10.35,
      9.52, 10.61, 9.83, 10.14, 9.42, 8.82, 8.84, 9.65, 11.09, 9.49,
      10.03, 10.59, 10.64, 10.74, 10.59, 9.04, 8.52, 9.83, 9.62, 9.45,
      10.83, 9.65, 10.37, 10.38, 9.66, 10.45, 9.99, 11.11, 10.47, 9.41)
```

- Bootstrap approach

```
N = length(X)
x_means = replicate(10000, mean(sample(X, N, replace = TRUE)))
quantile(x_means, c(alpha/2, 1-alpha/2))
#>      2.5%     97.5%
#>  9.755395 10.157400
```

- Asymptotic results (sampling distribution)

```
mean(X) + qt(c(alpha/2, 1-alpha/2), 49) * sqrt(var(X)/N)
#> [1]  9.748169 10.166631
```

`diamonds` activity

Explain the price's of the diamonds using information from the other variables. What are the more relevant variables to explain the price?

```
#> # A tibble: 53,940 × 10
#>    carat cut       color clarity depth table price     x     y     z
#>    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
#>  1  0.23 Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43
#>  2  0.21 Premium   E     SI1      59.8    61   326  3.89  3.84  2.31
#>  3  0.23 Good      E     VS1      56.9    65   327  4.05  4.07  2.31
#>  4  0.29 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63
#>  5  0.31 Good      J     SI2      63.3    58   335  4.34  4.35  2.75
#>  6  0.24 Very Good J     VVS2     62.8    57   336  3.94  3.96  2.48
#>  7  0.24 Very Good I     VVS1     62.3    57   336  3.95  3.98  2.47
#>  8  0.26 Very Good H     SI1      61.9    55   337  4.07  4.11  2.53
#>  9  0.22 Fair      E     VS2      65.1    61   337  3.87  3.78  2.49
#> 10  0.23 Very Good H     VS1      59.4    61   338  4     4.05  2.39
#> # … with 53,930 more rows
```

- `diamonds` is a well-known dataset and analyzed on many websites through the internet. You can borrow ideas from there.
- If you want, you can use either `quarto` o `Rmarkdown` to generate your document.

Hypothesis testing

About hypothesis testing

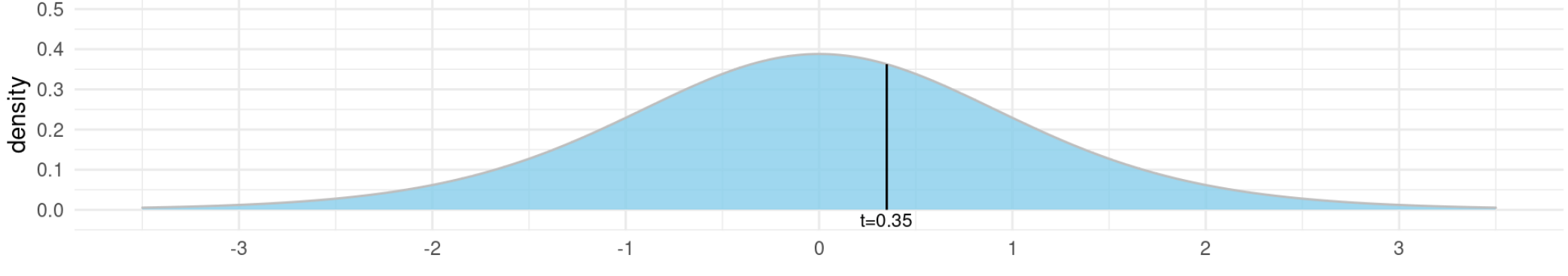
- Hypothesis testing is a process for which **after assuming a certain hypothesis** we associate our sample to the realization of a certain random variable.
  - Step 1. **Set a hypothesis** H_0 about the distribution generating X.
  - Step 2. **Collect** a sample X.
  - Step 3. **Compute a test statistics from sample** X, for which we know its distribution.
  - Step 4. **Calculate the probability**, p, to obtain test statistics associated to samples that are less likely to hold hypothesis H_0. The probability p is called p-value.
  - Step 5. **Decide** about the credibility of H_0 using probability p.

Hypothesis testing for the mean

- Step 1. $H_0 : \mu = 10$
- Step 2. Collect a sample

```
1  X_num = c(10.2, 8.8, 11.2, 9, 9.6, 10.3, 10.2, 13.2, 12.5, 7)
```

- Step 3. Compute the test statistic $t = (\bar{x} - \mu)/(s_x/\sqrt{N}) = 0.35.$
  - If H_0 is true, t is a realization of t_{N-1}.
- Step 4. The probability of obtaining "rarer" test statistics is given by the red area:



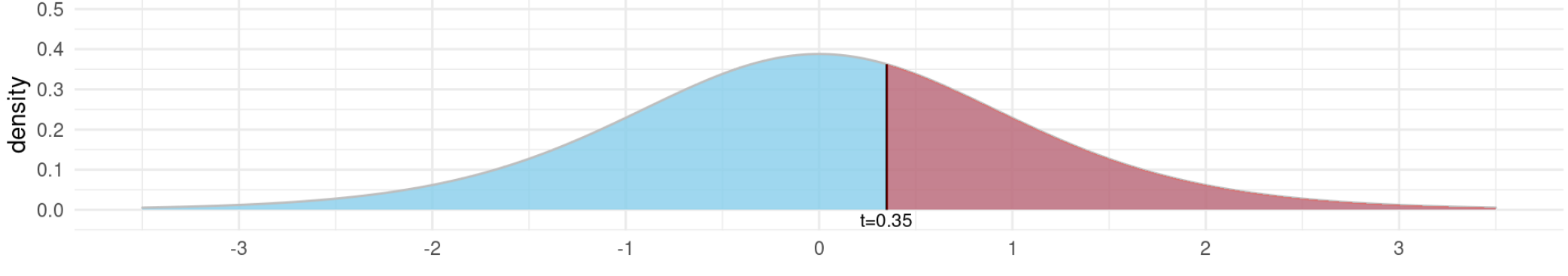
- Step 5. Is it rare an event that rarer events happen with probability 0.7344?

Hypothesis testing for the mean

- Step 1. $H_0 : \mu = 10$
- Step 2. Collect a sample

```
1  X_num = c(10.2, 8.8, 11.2, 9, 9.6, 10.3, 10.2, 13.2, 12.5, 7)
```

- Step 3. Compute the test statistic $t = (\bar{x} - \mu)/(s_x/\sqrt{N}) = 0.35.$
  - If H_0 is true, t is a realization of t_{N-1}.
- Step 4. The probability of obtaining "rarer" test statistics is given by the red area:



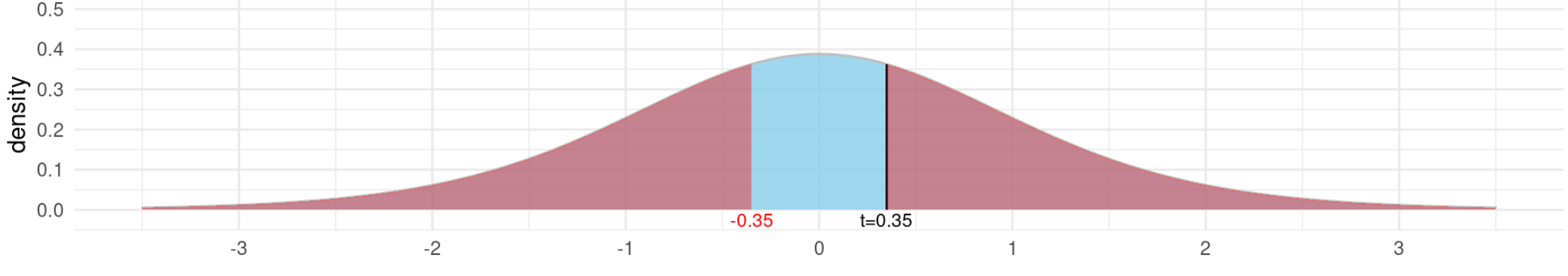
- Step 5. Is it rare an event that rarer events happen with probability 0.7344?

Hypothesis testing for the mean

- Step 1. $H_0 : \mu = 10$
- Step 2. Collect a sample

```
1  X_num = c(10.2, 8.8, 11.2, 9, 9.6, 10.3, 10.2, 13.2, 12.5, 7)
```

- Step 3. Compute the test statistic $t = (\bar{x} - \mu)/(s_x/\sqrt{N}) = 0.35.$
  - If H_0 is true, t is a realization of t_{N-1}.
- Step 4. The probability of obtaining "rarer" test statistics is given by the red area:



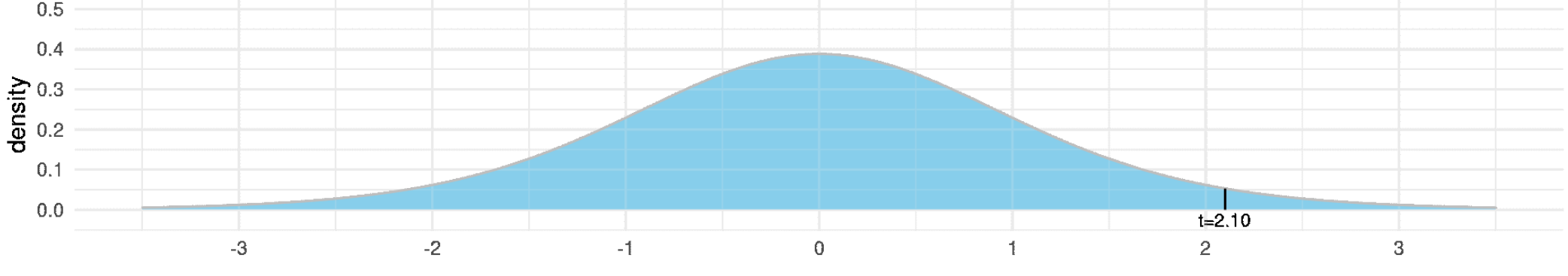
- Step 5. Is it rare an event that rarer events happen with probability 0.7344?

Hypothesis testing for the mean

- Step 1. $H_0 : \mu = 9$
- Step 2. Collect a sample

```
1  X_num = c(10.2, 8.8, 11.2, 9, 9.6, 10.3, 10.2, 13.2, 12.5, 7)
```

- Step 3. Compute the test statistic $t = (\bar{x} - \mu)/(s_x/\sqrt{N}) = 2.1$.
  - If H_0 is true, t is a realization of t_{N-1}.
- Step 4. The probability of obtaining "rarer" test statistics is given by the red area:



- Step 5. Is it rare an event that rarer events happen with probability 0.0651?

Statistical significance

We say that a result has statistical significance when the result is very unlikely to be occurred given the null hypothesis. Before a hypothesis test is done, a significance level, α, is set. Usually $\alpha = 0.05$.

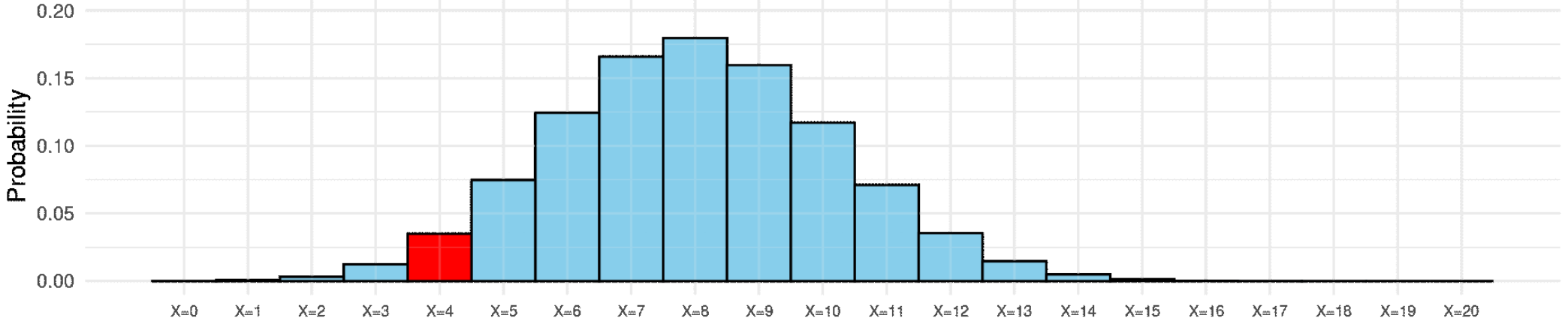
- So in a hypothesis test is critical to know what is the null hypothesis, and
- what are the assumptions to assert that the test statistic follows a certain distribution.

Hypothesis testing for proportions

- Step 1. $H_0 : \pi = 0.4$. Is it reasonable to assume $\pi = 0.4$?
- Step 2. Given a dichotomous sample $X = \{x_1, \dots, x_{20}\}$:

```
1  X_bin = c(0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0)
```

- Step 3. $\sum_i x_i = 4 \sim Bin(n = N, \pi)$
- Step 4. The probability of obtaining "rarer" test statistics is given by the red area:



- Step 5. Is it rare an event that rarer events happen with probability 0.072?

One sample t-test with R

- $H_0 : \mu = 10$

```
t.test(X_num, mu = 10)
#>
#>  One Sample t-test
#>
#> data:  X_num
#> t = 0.35052, df = 9, p-value = 0.734
#> alternative hypothesis: true mean is not equal to 10
#> 95 percent confidence interval:
#>   8.90927 11.49073
#> sample estimates:
#> mean of x
#>      10.2
```

One sample t-test with R

- $H_0 : \mu = 9$

```
t.test(X_num, mu = 9)
#>
#>  One Sample t-test
#>
#> data:  X_num
#> t = 2.1031, df = 9, p-value = 0.06479
#> alternative hypothesis: true mean is not equal to 9
#> 95 percent confidence interval:
#>   8.90927 11.49073
#> sample estimates:
#> mean of x
#>      10.2
```

Exact binomial test with R

- $H_0 : \pi = 0.4$

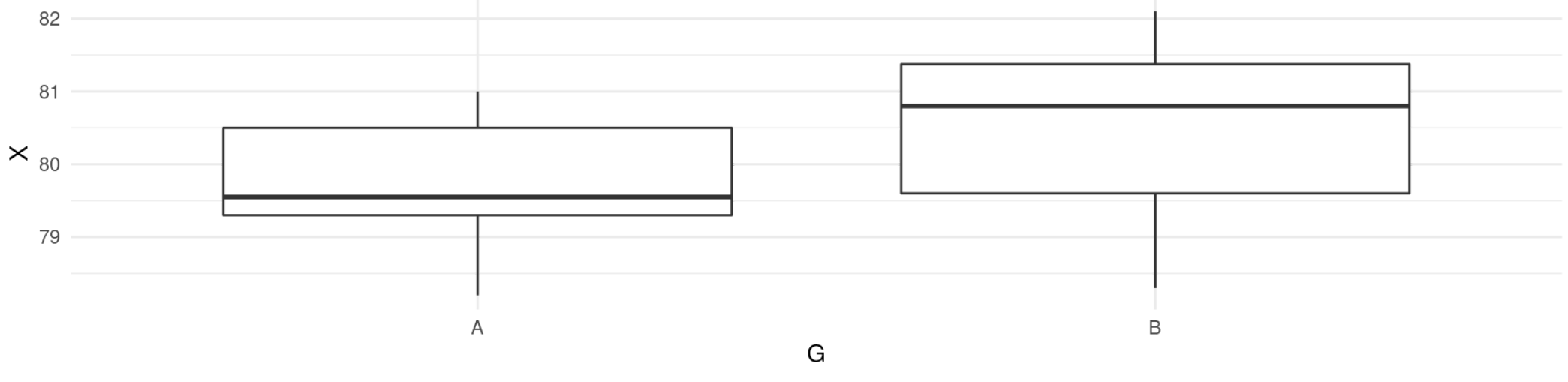
```
binom.test(sum(X_bin), n = length(X_bin), p = 0.4)
#>
#>  Exact binomial test
#>
#> data:  sum(X_bin) and length(X_bin)
#> number of successes = 4, number of trials = 20, p-value = 0.07198
#> alternative hypothesis: true probability of success is not equal to 0.4
#> 95 percent confidence interval:
#>  0.057334 0.436614
#> sample estimates:
#> probability of success
#>                    0.2
```

Comparing group means (1)

- Step 1.

```r
data = tibble(
  X = c(80.8, 78.2, 81, 79.9, 80.7, 79.3, 78.7, 79.6, 79.5, 79.3,  # <- A
        81.8, 81.4, 78.3, 79.4, 82.1, 81.3, 79, 80.5, 81.1, 80.2), # <- B
  G = c(rep('A', 10), rep('B', 10))
)
```

- Step 2. $H_0 : \mu_A = \mu_B$



Comparing group means (2)

- Step 3, 4, 5. Test statistics and p-values are easily calculated with R.

```
t.test(X~G, data = data)
#>
#>  Welch Two Sample t-test
#>
#> data:  X by G
#> t = -1.638, df = 16.394, p-value = 0.1205
#> alternative hypothesis: true difference in means between group A and group B is not equal to 0
#> 95 percent confidence interval:
#>  -1.8562806  0.2362806
#> sample estimates:
#> mean in group A mean in group B
#>           79.70           80.51
```

Independence tests

- H_0 when comparing the means of two groups are we are deciding whether the expected value of a numerical variable is independent of a binary group variable.

Depending on the nature of the two variables, other tests exist to contrast their independence.

- Between categorical variables: Fisher exact test, or chi-squared contingency table test if Fisher does not work.

```
1 fisher.test(table(X,Y)) # chisq.test(table(X,Y))
```

- Between numerical variable: test for Association/Correlation Between Paired Samples.

```
1 cor.test(X,Y)
```

- Between categorical (X) and numerical (Y) variables: ANOVA test.

```
1 summary(aov(Y~X))
```

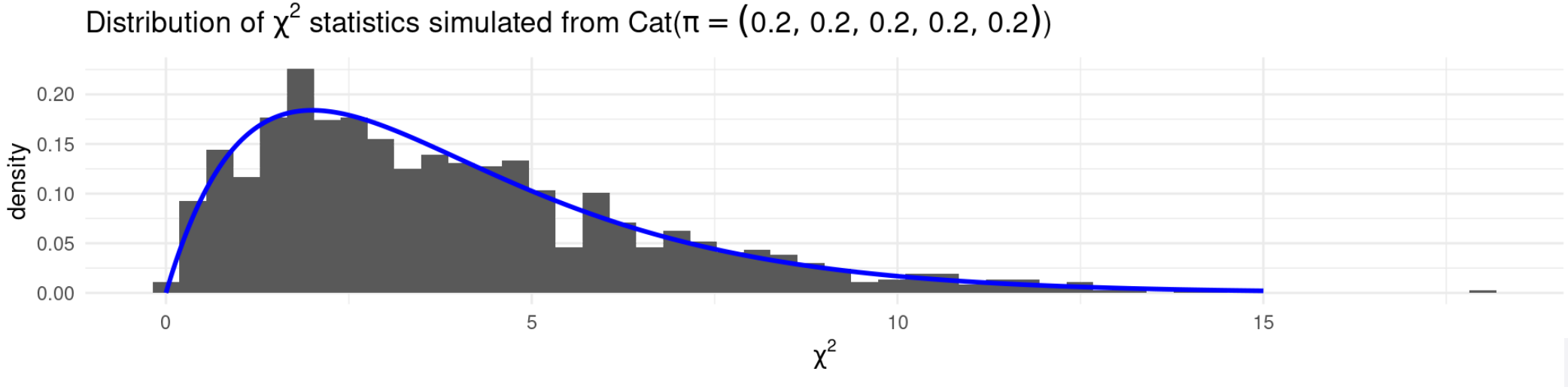
Categorical distribution

Pearson's chi-squared test

- $H_0: X \sim Cat(\boldsymbol{\pi} = (\pi_1, \ldots, \pi_k))$.
- Test statistic:

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i} \sim \chi^2_{k-1}$$

where O_i is the number of times i was observed and $E_i = n \times \pi_i$.



Gaussian distribution

Shapiro-Wilk test

- $H_0: X \sim N(\mu, \sigma)$
- Test statistic:

$$W = \frac{(\sum_{i=1}^{n} a_i x_{[i]})^2}{(\sum_{i=1}^{n} (x_i - \bar{x})^2)^2} \sim f_W$$

where a_i's are certain constants, $x_{[i]}$ is the i-th smallest observation in X and f is a probability distribution for r.v. W.

```
set.seed(1)
x = rnorm(100)
shapiro.test(x)
#>
#>  Shapiro-Wilk normality test
#>
#> data:  x
#> W = 0.9956, p-value = 0.9876
```

Comparing group variances (homoscedasticity)

For X categorical and Y numerical.

- H_0: $\mathrm{Var}(Y|X)$ is constant.

```
1 bartlett.test(Y~X)
```

- Bartlett's test assumes normality. For non-normal variables better to use the Levene test (available in package `car`)

```
1 car::leveneTest(Y~X)
```

broom Statistical Objects into Tibbles

```
binom.test(c(682, 243), p = 3/4)
#>
#>  Exact binomial test
#>
#> data:  c(682, 243)
#> number of successes = 682, number of trials = 925, p-value = 0.3825
#> alternative hypothesis: true probability of success is not equal to 0.75
#> 95 percent confidence interval:
#>  0.7076683 0.7654066
#> sample estimates:
#> probability of success
#>              0.7372973
```

With broom:

```
library(broom)
tidy(binom.test(c(682, 243), p = 3/4))
#> # A tibble: 1 × 8
#>   estimate statistic p.value parameter conf.low conf.high method          alter…¹
#>      <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl> <chr>           <chr>
#> 1    0.737       682   0.382       925    0.708     0.765 Exact binomia… two.si…
#> # … with abbreviated variable name ¹alternative
```

...

```
km_ = kmeans(iris[1:4], 2, nstart = 100)
tidy(km_)
#> # A tibble: 2 × 7
#>   Sepal.Length Sepal.Width Petal.Length Petal.Width  size withinss cluster
#>          <dbl>       <dbl>        <dbl>       <dbl> <int>    <dbl> <fct>
#> 1         6.30        2.89         4.96       1.70     97    124.  1
```

That’s all for today

Next week session

Overview Data Science and Data preprocessing with *Karina Gibert*.