Assignment 9

EMATM0061: Statistical Computing and Empirical Methods, TB1, 2022

Dr. Rihuan Ke

Introduction

This is the 9th assignment for Statistical Computing and Empirical Methods (Unit EMATM0061) on the MSc in Data Science & MSc in Financial Technology with Data Science. This assignment is mainly based on Lectures 22, 23 and 24 (see the Blackboard).

The submission deadline for this assignment is 23:59, 5 December 2022. Note that this assignment will not count towards your final grade. However, it is recommended that you try to answer the questions to gain a better understanding of the concepts.

Create an R Markdown for the assignment

It is a good practice to use R Markdown to organize your code and results.

If you are considering submitting your solutions, please generate a PDF file. For example, you can choose the "PDF" option when creating the R Markdown file (note that this option may require Tex to be installed on your computer), or use R Markdown to output an HTML and print it as a PDF file in a browser, or use your own way of creating a PDF file that contains your solutions.

Only a PDF file will be accepted in the submission of this assignment. To submit the assignment, please visit the "Assignment" tab on the Blackboard page, where you downloaded the assignment.

Wish to know more about a perticular question?

You may want to ask a question during the computer lab.

Alternatively, we are collecting questions about this assignment that need to be addressed, through the following form. And this can be done either during the labs or outside the lab sessions. So, If you found a question in this assignment interesting but had difficulty in a particular step when trying to develop your answer, please put your remark in the form via the following link. A brief description of the difficulty would be very helpful. Giving your remark is optional, but we aim to know the most common questions that you might want to get some support.

https://forms.office.com/e/9HXgjXijPN

Load packages

Some of the questions in this assignment require the tidyverse package. If it hasn't been installed on your computer, please use `install.packages()` to install them first.

To road the tidyverse package:

```
library(tidyverse)
```

1. Basic concepts in classification

In lecture 24, we introduced some concepts in classification.

(Q1) Write down your explanation of each of the following concepts. Give an example where appropriate.

1. A classification rule
2. A learning algorithm
3. Training data
4. Feature vector
5. Label
6. Expected test error
7. Train error
8. The train test split

2. A chi-squared test of population variance

Suppose we have an i.i.d. sample $X_1, \cdots, X_n \sim \mathcal{N}(\mu, \sigma^2)$ and a conjectured value for the population variance σ_0^2. We wish to test the null hypothesis that σ_0^2.

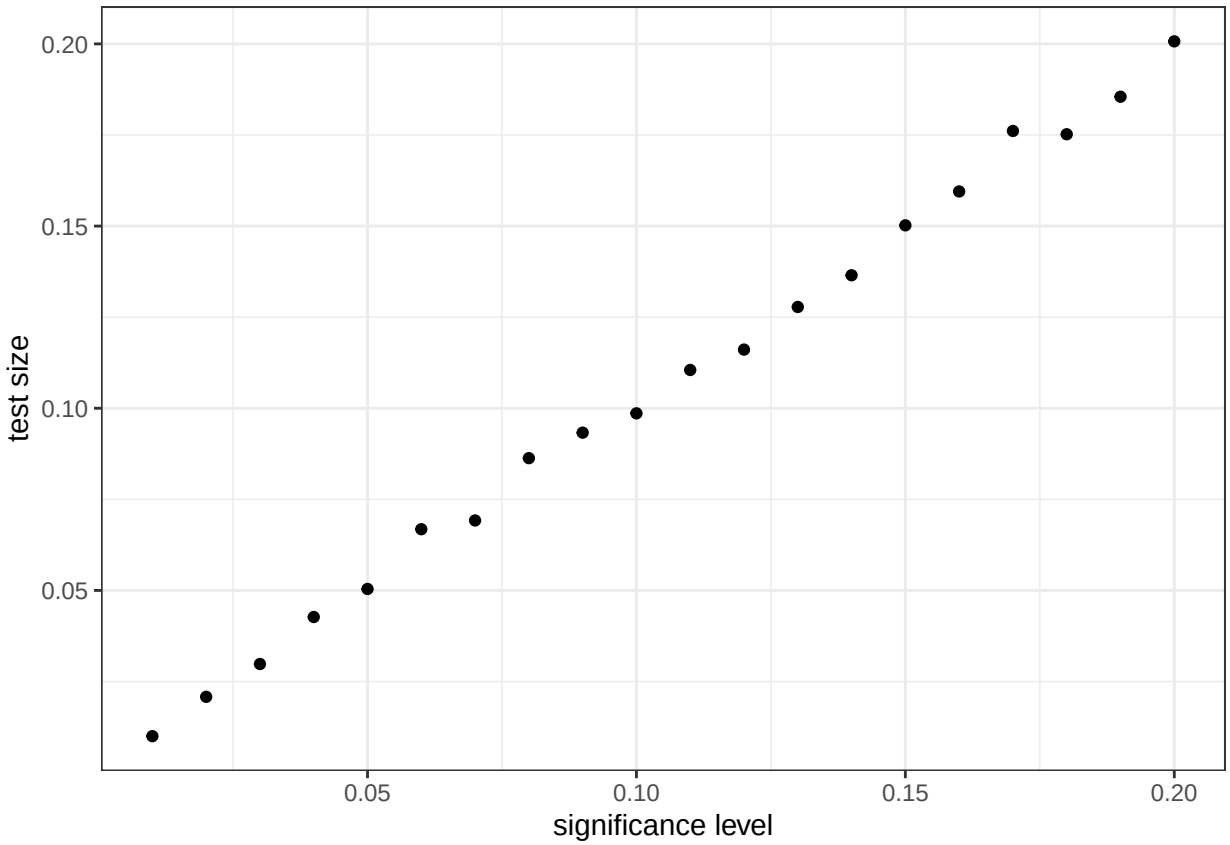
(Q1)

Implement a function called `chi_square_test_one_sample_var` which takes as input a sample called "`sample`" and a null value for the variance called `sigma_square_null`.

(Q2)

Conduct a simulation study to see how the size of the test varies as a function of the significance level. You can consider a sample size of 100, $\mu = 1$, $\sigma^2 = 4$.

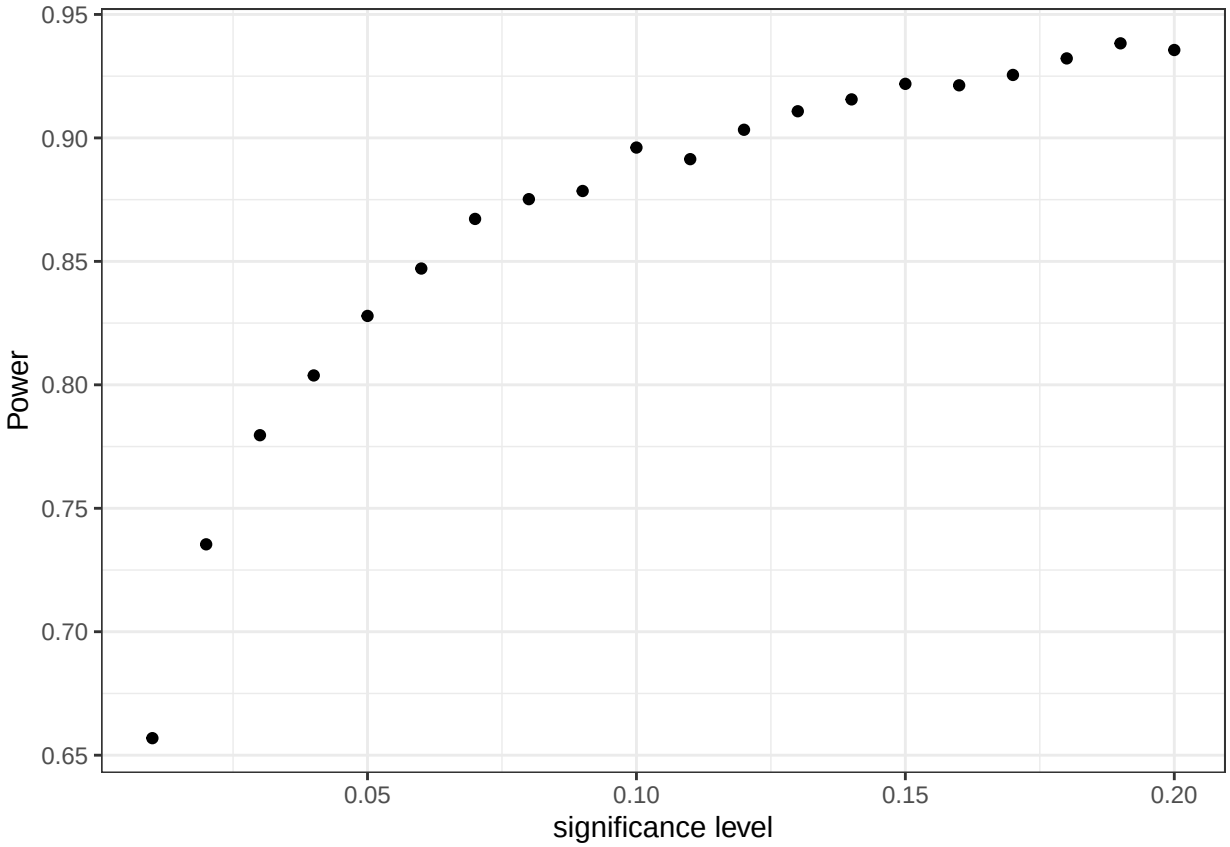
For example, you can create a plot similar to the one below.



(Q4)

Conduct a simulation study to see how the statistical power of the test varies as a function of the significance level. You can consider a sample size of 100, $\mu = 1$, $\sigma^2 = 6$ and $\sigma_0^2 = 4$.

For example, you can create a plot similar to the one below.



(Q5)

Load the "Palmer penguins" library and extract a vector called "`bill_adelie`" consisting of the bill lengths of the Adelie penguins belonging to the Adelie species.

Suppose we model the sequence of bill lengths as a sample of independent and identically distributed Gaussian random variables $X_1, \cdots, X_n \sim \mathcal{N}(\mu, \sigma^2)$ with a population mean μ and population standard deviation σ.

Now apply your function "`chi_square_test_one_sample_var`" to test the null hypothesis that the population standard deviation is 3 mm at a significance level of $\alpha = 0.1$.

3. The train test split

Suppose you want to build a classifier to predict whether a hawk belongs to either the "Sharp-shinned" or the "Cooper's" species of hawks. The feature vector will be a four-dimensional row vector containing the weight, and the lengths of the wing, the tail and the hallux. The labels will be binary: 1 if the hawk is "Sharp-shinned" and 0 if the hawk belongs to "Cooper's" species.

(Q1)

Begin by loading the "Hawks" data frame from the "Stat2Data" library. Now extract a subset of the data frame called "`hawks_total`" with five columns - "`Weight`", "`Wing`", "`Hallux`", "`Tail`" and "`Species`". The data frame should only include rows corresponding to hawks from either the "Sharp-shinned" (SS) or the "Cooper's" (CH) species, and not the "Red-tailed" (RT) species. Convert the Species column to a binary variable with a 1 if the hawk belongs to the sharp-shinned species and 0 if the hawk belongs to Cooper's species. Finally, remove any rows with missing values from one of the relevant columns.

(Q2)

Now implement a train test split for your "`hawks_total`" data frame. You should use 60% of your data within your training data and 40% in your test data. You should create a data frame consisting of training data called "`hawks_train`" and a data frame consisting of test data called "`hawks_test`". Display the number of rows in each data frame.

(Q3)

Next extract a data frame called "`hawks_train_x`" from your training data (from "`hawks_train`") containing the feature vectors and no labels. In addition, extract a vector called "`hawks_train_y`" consisting of labels from your training data. Similarly, create data frames called "`hawks_test_x`" and "`hawks_test_y`" corresponding to the feature vectors and labels within the test set, respectively.

(Q4)

Now let's consider a very simple (and not very effective) classifier which entirely ignores the feature vectors. Instead the classifier simply predicts a single fixed value $\hat{y} \in \{0, 1\}$. Hence, your classifier is of the form $\phi_{\hat{y}}(x) \equiv y$ for all $x \in \mathbb{R}^4$. Begin by choosing a value $\hat{y} \in \{0, 1\}$ based on your training data - choose the value which minimises the training error.

(Q5)

Next compute the train and test error of $\phi_{\hat{y}}$

In general, $\phi_{\hat{y}}$ performs poorly, as it does not use any information of the feature vector. However, in the example, the train error and test error seems relatively low (much less than 50%). Try to explain why the errors are relatively low. In which cases we might have a error of $\phi_{\hat{y}}$ close to 50%?

4. Multivariate distributions and parameter estimation

Suppose that we have a sample of red-tailed hawks and we want to investigate the distribution of several features (Wing, Weight and Tail) of red-tailed hawks. We model the Wing, Weight and Tail with a multivariate Gaussian distribution.

(Q1)

Again, load the "Hawks" data frame from the "Stat2Data" library. Now extract a subset of the data frame called "`hawks_rt`" that contains only the rows corresponding to hawks from the "Red-tailed" (RT) species and three columns - "`Wing`", "`Weight`", "`Tail`", "`Tail`". Remove any rows of "`hawks_rt`" with missing values from one of the relevant columns.

(Q2)

Now, lets model the three features "`Wing`", "`Weight`" and "`Tail`" with a multivariate Gaussian distribution. Suppose that your data frame `hawks_rt` consists of a i.i.d. sample $X_1, \cdots, X_n \sim \mathcal{N}(\mu, \Sigma)$. Here we model the three features "`Wing`", "`Weight`" and "`Tail`" with a multivariate Gaussian distribution with population mean μ and population covariance matrix Σ. Compute the minimum variance unbiased estimates (MVUE) of the μ and Σ.

5. Investigate hypotheis testing where the Gaussian model assumptions are violated

Many hypothesis testing approaches assume that the underlying distribution is Gaussian or approximately Gaussian. For example, when we apply one-sample t-test to test a value of the population mean for a population, we often assume that the distribution of our data is Gaussian or approximately Gaussian. In this question, we will carry out a simulation study to investigate the behaviour of the hypothesis testing when this assumption is violated. Let's focus on one sample t-test here.

Consider a continuous random variable X with the following probability density function

$$f(x) = \begin{cases} 2(x-a) & \text{if } x \in (a, a+1) \\ 0 & \text{othewise.} \end{cases}$$

where a is a parameter. We will perform a one-sample t-test in a setting where the sample are i.i.d. copies of X (which is non-Gaussian).

(Q1)

Let the population mean of X be denoted by μ. Express a as a function of μ.

Then write down a formula for the cumulative distribution F_X of X and write down a formula for the quantile function F_X^{-1} of X.

(Q2)

Write an R function called `generate_sample_X` for generating samples of X. The function should take as input two arguments the sample size called `sample_size` and the population mean called `mu` and output a vector. The output vector is computed as follows:

1. First, generate a sample call `sample_U` (the size of `sample_U` should be `sample_size`) from the uniform distribution between 0 and 1. You can use the function `runif`.
2. Second, apply the function F_X^{-1} to each element of `sample_U` to get a new vector (and this is the output of `generate_sample_X`).

(Q3) (*optional)

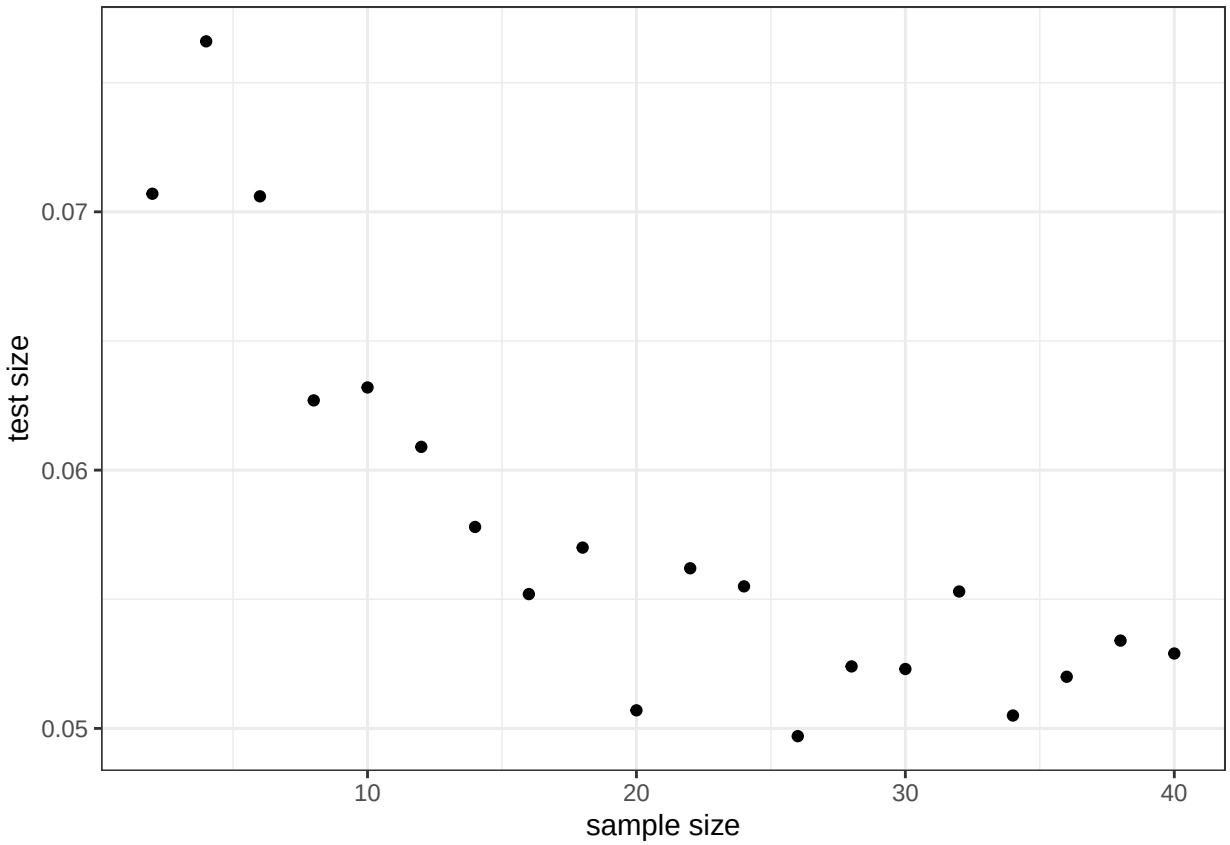
Explain why the function `generate_sample_X` generate a sample of X.

(Q4)

Suppose that, given a sample of X, we want to test if the population mean μ is equal to a given number μ_0.

Conduct a simulation study to investigate the test size (the average number of rejecting the null hypothesis when $\mu = \mu_0$) of one sample t-test applied to the case where the underlying distribution is the distribution X rather than a Gaussian. To obtain samples of X you can use your function `generate_sample_X`. Plot the test size as a function of the sample size. In your simulation study, you can set the population mean of X to 3 (in this case the μ_0 should be also 3). Consider a set of different sample sizes ranging from 2 to 40, and for each sample size, carry out 10000 trials.

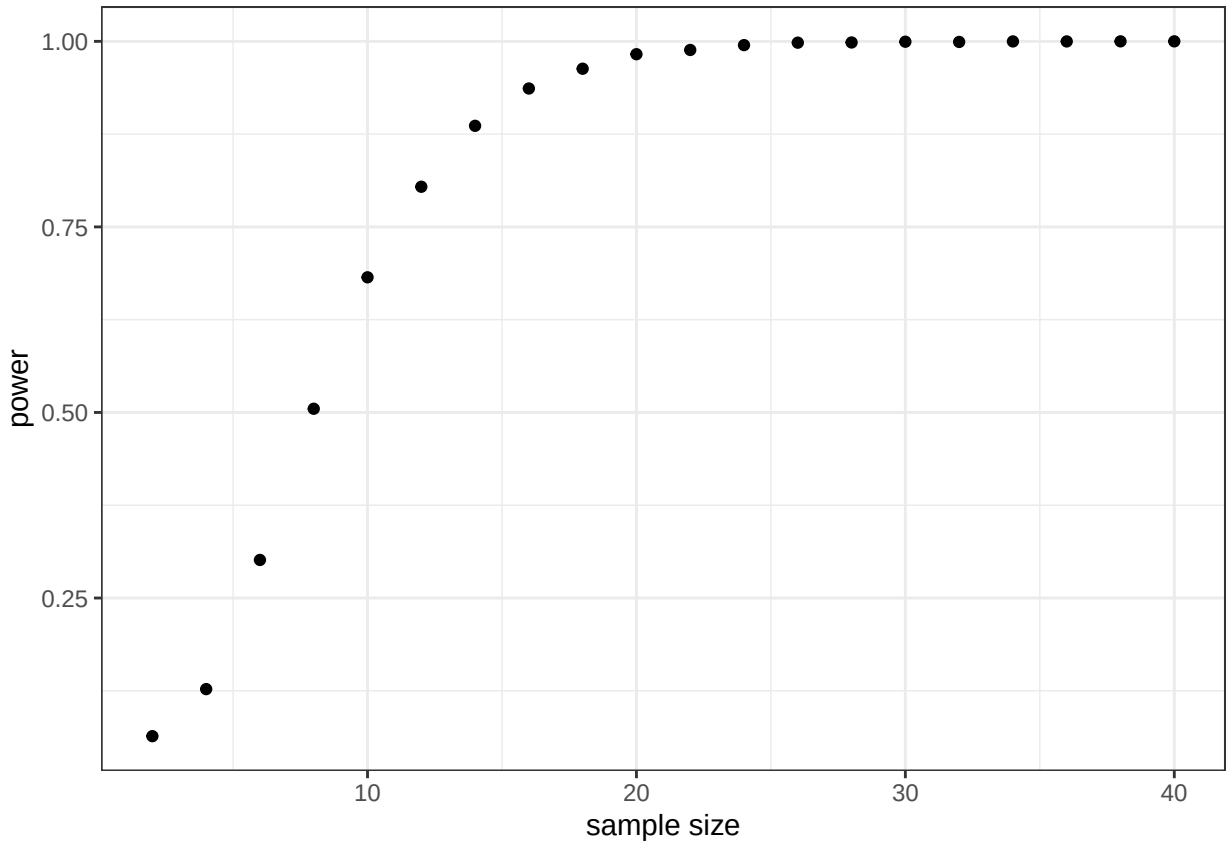
Your plot may look like this:



(Q5)

Conduct a simulation study to investigate the statistical power (the average number of rejecting the null hypothesis when $\mu \neq \mu_0$) of one sample t-test applied to the case where the underlying distribution is the distribution X rather than a Gaussian. To obtain samples of X you can use your function `generate_sample_X`. Plot the power as a function of the sample size. In your simulation study, you can set the population mean of X to 3 and $\mu_0 = 3.2$. Consider sample sizes ranging from 2 to 40, and for each sample size, carry out 10000 trials.

Your plot may look like this:


1.00
0.75
0.50
0.25
power
10
20
30
40
sample size