Practical application of the Wilcoxon-Mann-Whitney test in valuation.

Selection of attributes as pricing factors based on the principle of unbiased estimates

K. A. Murashev

August 16, 2022

In their practice appraisers often face the need to take into account differences in quantitative and qualitative characteristics of objects. In particular, one of the standard tasks is to determine the attributes that influence the cost (so-called "pricing factors") and to separate them from the attributes that do not or cannot be determined.

Subjective selection of attributes taken into account in determining the value is widespread in valuation practice. In this case, specific quantitative indicators of the impact of these attributes on the cost are often taken from the so-called "reference books". While not denying the speed and low cost of this approach, it should be recognized that only data directly observed in the open markets is a reliable basis for a value judgment. The priority of such data over other data, in particular those obtained by expert survey, is enshrined, among others, in RICS Valuation — Global Standards 2022 [14], International Valuation Standards 2022 [15], as well as in IFRS 13 "Fair Value Measurement" [11]. Therefore, we can say that mathematical methods for analyzing data from the open market are the most reliable means of interpreting market information used in market research and predicting the value of individual objects.

The aim of this work is to justify the necessity and possibility of using a rigorous mathematical Wilcoxon–Mann–Whitney test, which allows us to answer the question about the necessity of taking into account the binary attribute as a price-generating factor. Instead of the judgmental approach, which is most commonly used by appraisers in selecting the attributes to be considered in appraisal, this paper proposes the idea of prioritizing the measuring approach based on the results of a mathematical test that allows to draw a conclusion about the importance or otherwise of the binary attribute influence on the value. It should be noted that despite the fact that the statistical test under consideration belongs to frequentist statistics, it, through its connection to ROC analysis and AUC, is related to modern machine learning methods, which will be discussed later in the text of this material. The presence of this relationship and elements of Bayesian statistics seems particularly interesting and promising from the point of view

of introducing machine learning and data analysis methods into the everyday practice of appraisers.

Users should have some general math background and basic Python and R programming skills to understand and practice all of the material in the text, but lack of that knowledge and skill is not a barrier to learning most of the material and implementing the test in the spreadsheet that comes with it.

The material consists of four blocks:

- a description of the Wilcoxon–Mann–Whitney test (hereafter "U-test"), its probabilistic meaning, and its relationship to other mathematical methods;
- a practical implementation of the U-test in a spreadsheet on an example of test random data;
- practical implementation of the U-test on the real data of the residential real estate market of St. Petersburg agglomeration by means of Python programming language, the purpose of the analysis was to check the significance of the difference in the unit price between the objects located in the urban and suburban parts of the agglomeration;
- practical implementation of the U-test on real data of residential real estate market of Almaty by means of R programming language, the purpose of the analysis was to check the significance of difference in unit price between the objects sold without demountable improvements and the objects sold with them.

The current version of this material, its source code, Python and R scripts, and the spreadsheet are in the repository on the GitHub portal and are available at the permanent link [16].

Contents

List of Tables

List of Figures

Listings

1 Technical details

This material, as well as the appendices to it, are available at permanent link [16]. The source code for this work was created using the language TEX [36] with a set of macro extensions LATEX 2_ε [37], distribution TeXLive [38] and Editor TeXstudio [68]. The spreadsheet calculation was done with LibreOffice Calc [22] (Version: 7.3.4. 2 / LibreOffice Community Build ID: 30(Build:2); CPU threads: 4; OS: Linux 5.11; UI render: default; VCL: kf5 (cairo+xcb) Locale: en-US (en_US.UTF-8); UI: en-US Ubuntu package version: 1:7.3.4 rc2-0ubuntu0.20.04.1 lo1; Calc: threaded). The calculation in R [39] (version 4.2.1 (2022-06-23) – "Funny-Looking Kid") was done using an IDE RStudio (RStudio 2022. 02.3+492 "Prairie Trillium" Release (1db809b8, 2022-05-20) for Ubuntu Bionic; Mozilla/5.0 (X11; Linux x86_64); AppleWebKit/537.36 (KHTML, like Gecko); QtWebEngine/5.12.8; Chrome/69.0.3497.128; Safari/537.36) [35]. The calculation in Python (Version 3.9.12) [21] was performed using the development environment Jupyter Lab (Version 3.4.2) [28] and IDE Spyder (Spyder version: 5.1.5 None* Python version: 3.9.12 64-bit * Qt version: 5.9.7 * PyQt5 version: 5.9.2 * Operating System: Linux 5.11.0-37-generic) [34]. The graphics used in the subsection **??** were prepared using Geogebra (Version 6.0.666.0-202109211234) [23]. The following values were used in this material as well as in most of the works in the series:

- significance level: $\alpha = 0.05$;
- confidence interval: $Pr = 0.95$;
- initial position of the pseudo-random number generator: $seed = 19190709$.

A dot is used as a decimal point. Most of the mathematical notations are written as they are used in English-speaking circles. For example,

a tangent is written as tan, not tg. The results of statistical tests are considered significant when

$$p \leq \alpha. \tag{1.1}$$

This decision is based, in part, on the results of the discussion that took place on researchgate.net [27].

2 Subject of research

When working with market data, the appraiser is often faced with the task of testing the hypothesis of whether a quantitative, ordinal or nominal attribute has a significant effect on the price. Real estate market analysts, developers, realtors, employees of collateral departments of banks, leasing and insurance companies, tax inspectors and other specialists have a similar task. At the same time, it is often impossible to collect large amounts of data that would allow a wide range of machine learning methods to be applied. In some cases appraisers consciously narrow the area of data collection to the narrow market segment, resulting in only very small samples of less than thirty observations at their disposal. In this case, the price data most often has a distribution that differs from the normal one. In this case, a rational solution is to use U-test. Let us formulate the problem:

- suppose that we have two samples of unit prices for commercial premises, some of which have some attribute (e. g., having a separate entrance) and some of which do not;
- it is necessary to determine whether the presence of this feature has a significant impact on the unit value of this type of real estate or not.

At first glance, according to established practice, an appraiser can simply subjectively recognize some attributes as significant and others as not, and then accept the adjustment values for differences in these attributes from the reference books. However, as mentioned above, this approach is hardly considered best practice because it lacks any market analysis. Also, in that case, it is unlikely that such work is of any serious value at all.

Instead, it is possible to use random samples of market data and apply mathematical analysis to them, allowing scientific and evidence-based conclusions to be drawn about the significance of a particular attribute's impact on value. The data used in this paper to perform the U-test using Python and R are real market data, some of which were collected by the author through web scraping and some provided by colleagues for the analysis. The attached spreadsheet is set up so that test raw data can be generated in a pseudo-random fashion.

The subject of this paper is the nonparametric Wilcoxon-Mann-Whitney test, specifically designed for samples that have a distribution other than normal. This circumstance is important because the price data that appraisers deal with most often have this distribution, which excludes the possibility of applying the parametric t-criterion and z-criterion. In addition, the test under consideration is of great interest because it has a connection to machine learning methods through AUC, the calculation of which through the formula provided in the test framework gives a value equal to that calculated by ROC analysis. Thus, the study of the U-test paves the way for a further dive into the world of machine learning, which is entering many areas of human activity and will significantly change the field of value estimation in the foreseeable future.

The material contains a description of the test and instructions for performing it, sufficient in the author's opinion for its demonstrable use in the estimation process.

3 Basic information about the test

3.1 Assumptions and formalization of hypotheses

First of all, it should be said that, in spite of the stated common name, it is more correct to speak of two tests:

- Wilcoxon rank-sum test developed by Frank Wilcoxon in 1945 [31];
- Mann–Whitney U-test which is a further development of the aforementioned criterion developed by Henry Mann and Donald Whitney in 1947 [29].

Looking ahead we can say that the statistics of these criteria are linearly related and their p-values are almost the same which from a practical point of view allows us to talk about variations of one test rather than two separate tests. This paper uses the common name throughout the text, as well as a shortened version of "U-test" which historically refers to the Mann-Whitney test. Some authors[9] recommend using the Wilcoxon rank-sum test when there are no assumptions about variance, and the Mann-Whitney U-test when variance of the two samples are equal. However, the experimental data indicate that the Wilcoxon rank-sum test and Mann-Whitney U-test values are essentially the same when the variance of the samples is significantly different. Adhering to the KISS principle [52] underlying the entire series of publications, the author concludes that a unified approach is possible. Also remember that the Wilcoxon signed-rank test is a separate test designed to analyze differences between two matched

samples, whereas the Mann-Whitney U-test discussed in this paper is designed to work with two independent samples.

Suppose that there are two samples:

$$x^m = (x_1, x_2, \ldots, x_m), x_i \in \mathbb{R}; \quad y^n = (y_1, y_2, \ldots, y_n), y_i \in \mathbb{R} \quad : m \leq n.$$

- Both samples are simple and random (i.e., SRS [63]), the combined sample is independent.
- The samples are taken from unknown continuous distributions *F(x)* and *G(y)*, respectively.

Simple random sample (SRS) — is a subset of individuals (*a sample*) chosen from a larger set (*a population*) in which a subset of individuals are chosen randomly, all with the same probability. It is a process of selecting a sample in a random way. In **SRS**, each subset of k individuals has the same probability of being chosen for the sample as any other subset of k individuals.A simple random sample is an unbiased sampling technique. Equivalent definition: a sample $x^m = (x_1, x_2, \ldots, x_m)$ is simple if the values $(x_1, x_2, \ldots, x_m)$ are realizations of m independent equally distributed random variables. In other words, the selection of observations is not only random but also does not imply any special selection rules (e.g., choosing every 10th observation).

The U-test — is a nonparametric criterion to test the null hypothesis that for randomly chosen from two samples of observations $x \in X$ and $y \in Y$ the probability that x is greater than y is equal to the probability that y is greater than x. In mathematical language, the null hypothesis is written as follows

$$H_0 : P\{x < y = \frac{1}{2}\}. \tag{3.1}$$

For the test's own consistency, an alternative hypothesis is required, which is that the probability that the value of a characteristic of observation from X is greater than that of observation from Y differs upward or downward from the probability that

the value of a characteristic of observation from Y is greater than that of observation from X. In mathematical language, the alternative hypothesis is written as follows

$$H_1 : P\{x < y\} \neq P\{y < x\} \vee P\{x < y\} + 0.5 \cdot P\{x = y\} \neq 0.5. \tag{3.2}$$

According to the basic concept of the U-test, if the null hypothesis is true, the distribution of the two samples is continuous; if the alternative hypothesis is true, the distribution of one sample is stochastically greater than the distribution of the other. In this case, it is possible to formulate a number of null and alternative hypotheses for which this test will give a correct result. His most extensive generalization lies in the following assumptions:

- the observations in both samples are independent;
- the data type is at least ranked, i. e., with respect to any two observations you can tell which one is greater;
- the null hypothesis assumes that the distributions of the two samples are equal;
- the alternative hypothesis assumes that the distributions of the two samples are unequal.

With a stricter set of assumptions than those given above, for example the assumption that the distribution of the two samples is continuous if the null hypothesis is valid and that the distribution of the two samples has a shift in the distribution if the alternative one is valid i. e. $f_1(x) = f_2(x + \sigma)$, we can say that the U-test is a test for the hypothesis of equality of medians. In this case, the U-test can be interpreted as a test of whether Hodges–Lehman's estimate of the difference in central tendency measures differs from zero. In this situation, the Hodges–Lehman estimate is the median of all possible values of differences between the observations in the first and second samples. However, if both the variance and the shape of the distribution of the two samples differ, the U-test cannot correctly test the medians.

Examples can be shown where the medians are numerically equal and the test rejects the null hypothesis because of the small p-value. Thus, a more correct interpretation of the U-test is to use it to test the shift hypothesis [30].

Shift hypothesis — is a statistical hypothesis often considered as an alternative to the hypothesis of complete homogeneity of samples. Let us have two samples of data. Let us also give two random variables X and Y, which are distributed as elements of these samples and have distribution functions *F(x)* and *G(y)*, respectively. In these terms, the shift hypothesis can be written as follows

$$H : F(x) = G(x + \sigma) \ : \forall x, \ \sigma \neq 0. \tag{3.3}$$

In this case, the U-criterion is valid regardless of the characteristics of the samples.

Simply put, the essence of the U-test is that it allows us to answer the question of whether there is a significant difference in the value of the quantitative attribute of the two samples. With regard to valuation, we can say that the use of this test helps to answer the question of whether it is necessary to take into account one or another attribute as a price-generating factor. It follows from the above that by default we are talking about a two-sided test. In practice, this means that the test does not give a direct answer to the question, for example: "Is there a significant excess of the unit value of premises with a separate entrance to the premises that do not have it. At the same time, there are also one-sided realizations that allow us to answer the question about the sign of the difference in the value of the attribute in the two samples.

In addition to the above requirements for the samples themselves, the conditions for applying the U-test are:

- the distribution of quantitative attribute values of samples is different from normal (otherwise it is advisable to use parametric Student's t-test or z-test for independent samples).
- at least three observations in each sample, it is allowed to have

two observations in one of the samples, provided that there are at least five in the other sample.

To summarize the above, there are three variants of the null hypothesis, depending on the level of rigor outlined in the table below 3.1.

Table 3.1: Variants of the null hypothesis when using the U-test in valuation.

Type of hypothesis	Formulation
Scientific	The two samples are completely homogeneous, i. e. they belong to the same distribution, there is no shift and the estimate made for the first sample is unbiased for the second one.
Practical	The medians of the two samples are equal to each other.
Set forth in terms of valuation	The difference in the attribute between the two samples of object-analogues is not significant, its accounting is not required and this attribute is not a pricing factor.

3.2 Test implementation

3.2.1 Test statistic

Suppose that the elements $x_1, \ldots, x_n$ represent a simple independent sample from $X \in \mathbb{R}$, and the elements $y_1, \ldots, y_n$ represent a simple independent sample from $Y \in \mathbb{R}$ and the samples are independent

of each other. Then the relevant U-statistic is defined as follows:

$$U = \sum_{i=1}^{m} \sum_{j=1}^{n} S(x_i, y_j),$$

$$S(x, y) = \begin{cases} 1, & x > y, \\ \frac{1}{2}, & x = y, \\ 0, & x < y. \end{cases} \tag{3.4}$$

3.2.2 Calculation methods

The test involves calculating a statistic usually called the U-statistic whose distribution is known if the null hypothesis is true. When working with very small samples, the distribution is specified tabularly; when the sample size is more than twenty observations, it is approximated quite well by the normal distribution. There are two methods of calculating U-statistics: manual calculation using the formula 3.4 or using a special algorithm. The first method, due to its labor-intensive nature, is only suitable for very small samples. The second method can be formalized as a step-by-step set of instructions and will be described below.

1. You must construct a common variation series for the two samples and then assign a rank to each observation, starting with one for the smallest of them. If there are ties, i. e. groups of repeating values (such a group can be, e. g., only two equal values), each observation from such a group is assigned a value equal to the median of the group ranks before adjustment (for example, in the case of a variation series (*3, 5, 5, 5, 5, 8*) the ranks before adjustment are (*1, 2, 3, 4, 5, 6*) after — (*1, 3.5, 3.5, 3.5, 3.5, 6*).

2. It is necessary to calculate the sums of the ranks of the observations of each sample, denoted as R_1, R_2 respectively. In this case, the total sum of ranks can be calculated by the formula

$$R = \frac{N(N+1)}{2}, \tag{3.5}$$

where N —the total number of observations in both samples.

3. Next, we calculate the U-value for the first sample:

$$U_1 = R_1 - \frac{n_1(n_1+1)}{2}, \tag{3.6}$$

where R_1 —the sum of ranks of the first sample, n_1 — the number of observations in the first sample.

4. The U-value for the second sample is calculated in the same way:

$$U_2 = R_2 - \frac{n_2(n_2+1)}{2}, \tag{3.7}$$

where R_2 —the sum of ranks of the second sample, n_2 — the number of observations in the second sample.

From the above formulas it follows that

$$U_1 + U_2 = R_1 - \frac{n_1(n_1+1)}{2} + R_2 - \frac{n_2(n_2+1)}{2}. \tag{3.8}$$

It is also known that

$$\begin{cases} R_1 + R_2 = \dfrac{N(N+1)}{2} \\ N = n_1 + n_2. \end{cases} \tag{3.9}$$

Then

$$U_1 + U_2 = n_1 n2. \tag{3.10}$$

Using this formula as a control ratio can be useful for checking the correctness of calculations in a spreadsheet processor.

5. From the two values of $U_1,\ U_2$ in all cases we choose the smaller which will be the U-statistic and used in further calculations. Let us denote it as U.

3.2.3 Interpretation of the result

For a correct interpretation of the test result it is necessary to specify:

- size of each sample;
- values of the measure of central tendency for each sample (given the nonparametric nature of the test, the median appears to be the appropriate measure of central tendency);
- the value of the U-statistic itself;
- the CLES index [43] the value of which is equivalent to the AUC and ρ-statistic;
- rank-biserial correlation coefficient (RBC) [59];
- the accepted level of significance (usually 0.05);
- the calculated p-value.

The concept of U-statistic was discussed earlier and most of the other indicators are widely known and do not require any particular consideration.

3.2.3.1 CLES = ρ-statistic = AUC

First of all, it must be said that all of these indicators are equivalent to each other. Thus

$$CLES = f = AUC_1 = \rho. \tag{3.11}$$

3.2.3.1.1 Common language effect size (CLES)

Common language effect size (CLES) — is the probability that the value of a randomly chosen observation from the first sample is greater than the value of a randomly chosen observation from the second sample. This indicator is calculated by the formula

$$CLES = \frac{U_1}{n_1 n_2}. \tag{3.12}$$

The designation *f (favorable)* is often used instead of *CLES*. This sample value is an unbiased estimate of the value for the entire population of objects belonging to the set.

It should be noted that the value and meaning of this indicator is equivalent to the value and meaning of the AUC[60]. Thus, we can say that this indicator characterizes the quality of the U-test as a binary classifier.

$$CLES = f = AUC_1 = \frac{U_1}{n_1 n_2}. \tag{3.13}$$

The relationship between the U-statistic and AUC is discussed in 3.4.

3.2.3.1.2 ρ-statistic A statistic called ρ that is linearly related to U and widely used in studies of categorization (discrimination learning involving concepts), and elsewhere, is calculated by dividing U by its maximum value for the given sample sizes, which is simply $n1 \times n2$. Thus, ρ is a non-parametric measure of the overlap between two distributions; it can take values between 0 and 1, and it is an estimate of $P(Y > X) + 0.5P(Y = X)$, where X and Y are randomly chosen observations from the two distributions. Both extreme values represent complete separation of the distributions, while a $= 0.5$ represents complete overlap. This statistic is useful in particular when despite a large p-value the medians of the two samples are essentially equal to each other.

3.2.3.2 Rank-biserial correlation (RBC)

The method of representing the measure of impact for the U-test is to use a measure of rank correlation known as rank-biserial correlation (hereafter RBC). As in the case of other measures of correlation, the value of the RBC coefficient has a range of values [-1;1], with a zero value indicating the absence of any relationship. The RBC coefficient is usually denoted as r. A simple formula based on the CLES (AUC, t, ρ) value is used to calculate it. Let us state the hypothesis that in a pair of random observations, one of which is taken from the first sample and

the other from the second, the value of the first is greater.Let's write it down in mathematical language:

$$H : x_i > y_j, \quad x \ \in X, \ y \in Y. \tag{3.14}$$

Then the value of the RBC coefficient is the difference between the proportion of random pairs of observations that are favorable (f) to the hypothesis and the complementary proportion of random pairs that are unfavorable to the hypothesis. Thus, this formula is a formula for the difference between the CLES scores for each of the groups.

$$r = f - u = CLES_1 - CLES_2 = f - (1 - f) \tag{3.15}$$

There are also a number of alternative formulas that give identical results:

$$r = 2f - 1 = \frac{2U_1}{n_1 n_2} - 1 = 1 - \frac{2U_2}{n_1 n_2}. \tag{3.16}$$

3.2.4 Calculation of the p-value and the final test of the null hypothesis

If the number of observations in both samples is large enough, the U-statistic has an approximately normal distribution. Then its standardized value (z-score) [64] can be calculated by the formula

$$z = \frac{U - m_U}{\sigma_U}, \tag{3.17}$$

where m_U is mean for U and σ_U is its standard deviation. A visualization of the concept of *standardized value for a normal distribution* is shown in Figure 3.1. The mean for the U is calculated by the formula

$$m_U = \frac{n_1 n_2}{2}. \tag{3.18}$$

The formula for the standard deviation in the case of no ties is as follows:

$$\sigma_U = \sqrt{\frac{n_1 n_2 (n_1 + n_2 + 1)}{12}}. \tag{3.19}$$

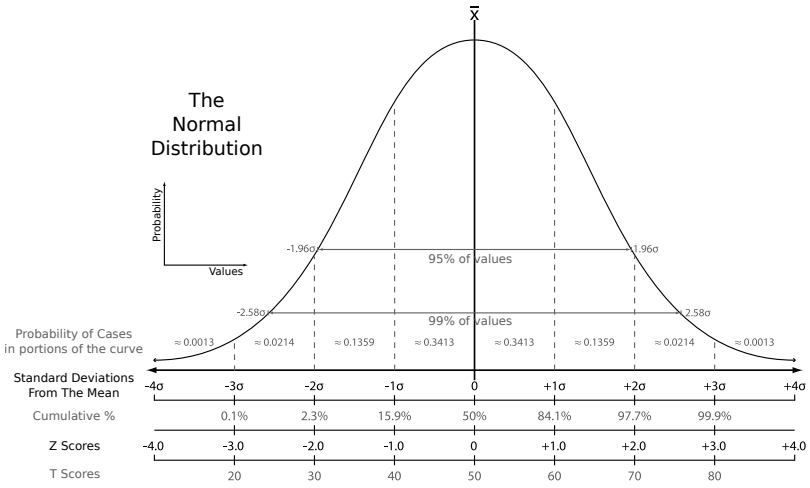

Figure 3.1: A visualization of the concept of standardized value for a normal distribution [64]

In case of the presence of tied ranks, a different formula is used:

$$\sigma_{U_{ties}} = \sqrt{\frac{n_1 n_2 (n_1 + n_2 + 1)}{12} - \frac{n_1 n_2 \sum_{k=1}^{K} ({t_k}^3 - t_k)}{12n(n-1)}} = \sqrt{\frac{n_1 n_2}{12}\left((n+1) - \frac{\sum_{k=1}^{K} ({t_k}^3} {n(n-} \right.} \tag{3.20}$$

where t_k is the number of observations with rank k and K is the total number of tied ranks. Then, by obtaining a standardized value (z-score) and using an approximation of the standard normal distribution, the p-value for a given level of significance (usually 0.05) is calculated. The interpretation of the result is as follows:

$$\begin{aligned} p \leq 0.05 &\Rightarrow \text{the null hypothesis is rejected} \\ p > 0.05 &\Rightarrow \text{the null hypothesis can not be rejected.} \end{aligned} \tag{3.21}$$

However, there is also an alternative interpretation:

$$\begin{aligned} p < 0.05 &\Rightarrow \text{the null hypothesis is rejected} \\ p \geq 0.05 &\Rightarrow \text{the null hypothesis can not be rejected.} \end{aligned} \tag{3.22}$$

To date, there is no unambiguous position on how the situation when $p = \alpha$ should be interpreted. This paper uses the version described in 3.21.

3.3 Relationship to other statistical tests

3.3.1 Comparison of Wilcoxon-Mann-Whitney U-test with Student's t-test

You often hear that the U-test is the nonparametric counterpart of the Student's t-test, designed for data whose distribution differs from the normal one. From a purely practical point of view, we can indeed say that in the case of a normal distribution it is advisable to determine whether there is a significant difference between the two samples by means of the t-test, and in the case of a distribution that differs from the normal by means of the U-test. Thus, it can be said that these tests are used for the same ultimate purpose.

However, the mathematical meaning of the U-test and the t-test are significantly different. As stated earlier, the U-test is designed to test the null hypothesis, which is that for randomly chosen from two samples of observations $x \in X$ and $y \in Y$ the probability that x is greater than y is equal to the probability that y is greater than x, the alternative hypothesis carries the claim that these probabilities are not equal. At the same time, the t-test is designed to test the null hypothesis that the means of the two samples are equal, while the alternative hypothesis is that the means of the two samples are not equal. In this regard, when comparing these tests, we should keep in mind that, in general, the U-test and the t-test check different null hypotheses, although they have partly similar practical meaning. The result of the U-test is most often very close to the result of the two-sample t-test for ranked data. Table 3.2 then provides a general comparison of the U-test with the t-test.

Table 3.2: Properties of the U-test relative to the t-test.

Property	Description
Applicability to ordinal data	When working with ordinal (rank) data, rather than quantitative data, the U-test is preferable to the t-test, remembering that the distance between neighboring values of the variation series cannot be considered constant.
Robustness	Since the U-test handles the sum of ranks rather than trait values, it is less likely than the t-test to erroneously indicate significance due to outliers. However, in general, the U-test is more prone to type I error in the case when the data simultaneously have the property of heteroscedasticity and have a distribution other than normal.
Efficiency	In the case of a normal distribution, the asymptotic efficiency of the U-test is $\frac{3}{4}\pi \approx 0.95$ of the t-test [4]. If the distribution differs significantly from the normal one and the number of observations is large enough, the efficiency of the U-test is significantly higher than the efficiency of the t-test [3]. However, this efficiency comparison should be interpreted with caution, because the U-test and the t-test examine different hypotheses and estimate different values. In the case, for example, of the need to compare means, the use of the U-test is not justified in principle.

3.3.2 Alternative tests in the case of inequality of distributions

If it is necessary to test the stochastic ordering of two samples (i.e. the alternative hypothesis: $H1 : \ P(Y > X) + 0.5P(Y = X) \neq 0.5$) without assuming equality of their distributions (i.e. when the null hypothesis is $H0 : \ P(Y > X) + 0.5P(Y = X) = 0.5$ but not $F(X) = G(Y)$), more appropriate tests should be used. These include the Brunner-Munzel test [12], which is a heteroskedasticity-resistant

analog of the U-test, and the Fligner-Policello test [20], which is a test for equality of medians. In particular, in the case of a more general null hypothesis $H0: \ P(Y > X) + 0.5P(Y = X) = 0.5$, the U-test can often lead to a type I error even in the case of large samples (especially in the case of disparity of variance and significantly different sample sizes), so that in such cases the use of alternative tests is preferable [13]. Thus, in the absence of the assumption of equality of distributions in case the null hypothesis is valid, the use of alternative tests will be preferable.

In the case of testing the hypothesis of a shift with significantly different distributions, the U-test may give an erroneous interpretation of significance [5], so in such circumstances it is preferable to use a variant of the t-test [41] designed for cases of unequal variance [5]. In some cases, it may be justified to convert quantitative data into ranks and then perform the t-test in some variant depending on the assumption of equality of variance. When converting quantitative data to ordinal data, the original variances will not be preserved; they must be recalculated for the ranks themselves. In the case of equal variance, a suitable nonparametric substitute for the F-test [19] can be the Brown-Forsythe test [2].

3.3.3 The relationship between the U-test and the classification tasks

The U-test is a particular case of the ordered logit model [32].

3.4 The relationship between the U-test and the concepts of Receiver operating characteristics (ROC) and Area Under Curve (AUC)

Based on what was said in 3.3.3, we can conclude that the U-test is not only a test for testing the shift hypothesis (or another one similar

in meaning), but also represents a kind of classifier. Looking ahead, the meaning of the U-test as a classifier is as follows:

- there is a "positive" outcome of comparing two random observations, which is that the observation from X is greater than the observation from Y;
- the proportion of the sum of the ranks of the "positive" elements is calculated.
- as in general with ROC, if the value of the share of "positive" elements exceeds 0.5, this means that the classifier generally performs its function; if it is equal to 0.5, its efficiency is equal to guessing with a coin flip; if it is less than 0.5, using such classifier yields the opposite result.

At first glance, the relationship between the U-test and ROC does not seem obvious. This section will attempt to understand why these concepts are related and what is the essence of the U-test as a classifier.

ROC analysis itself is outside the scope of this paper. Therefore, let us consider only its main points.

ROC curve — is a graphical plot that allows us to evaluate the quality of binary classification. It displays the ratio between the proportion of objects from the total number of feature carriers correctly classified as carrying the feature (True Positive Rate (TPR), called the *sensitivity of the classification algorithm*) and the proportion of objects from the total number of objects not carrying the feature, incorrectly classified as carrying the feature (False Positive Rate (FPR), the **1-FPR** value is called the *specificity of the classification algorithm*), when varying the threshold of the deciding rule. It is also known as **error curve**. Analysis of classifications using ROC curves is called **ROC analysis**.

Quantitative interpretation of the ROC curve gives the Area under curve (AUC).

Area under curve (AUC) — is the area bounded by the ROC curve and the axis of the proportion of false positive classifications (abscissa axis).

The higher the AUC, the better the quality of the classifier, while a value of 0.5 demonstrates the unsuitability of the chosen classification method (corresponding to a random coin guessing). A value of less than 0.5 indicates that the classifier works exactly the other way around: if you call positive results negative and vice versa, the classifier will perform better [60].

Let's introduce some terms.

Condition positive (P) — the number of real positive cases in the data.

Condition negative (N)— the number of real negative cases in the data.

True positive (TP) — a test result that correctly indicates the presence of a condition or characteristic.

True negative (TN) — a test result that correctly indicates the absence of a condition or characteristic.

False positive (FP) — a test result which wrongly indicates that a particular condition or attribute is present.

False negative (FN) — a test result which wrongly indicates that a particular condition or attribute is absent.

Based on the above, we can create a contingency table of the results of applying the binary classifier. The rows contain data on the actual presence or absence of the feature, the columns on the predicted with the classifier. As can be seen from Table 3.3, the binary classifier can

Table 3.3: Binary classifier contingency table.

Total $P + N$	Predicted Positive (PP)	Predicted negative (PN)
Positive (P)	TP	FN, type II error [65]
Negative (N)	FP, type I error [65]	TN

lead to errors of two types. Let's introduce some more definitions and

Table 3.4: Additional definitions and formulas for calculating the probabilities of binary classifier outcomes (part 1 of 3).

Notation	Formula	Deciphering the notation and alternative terms.
TPR (SEN)	$TPR = \frac{TP}{P} = 1 - FNR = \frac{TP}{TP + FN}$ (3.23)	**true positive rate**, **sensitivity** [61], recall [57], probability of detection, hit rate [49], power
FPR	$FPR = \frac{FP}{N} = 1 - TNR = \frac{FP}{FP + TN}$ (3.24)	**false positive rate**, probability of false alarm, fall-out [47]
FNR	$FNR = \frac{FN}{P} = 1 - TPR = \frac{FN}{FN + TP}$ (3.25)	**false negative rate** [66], miss rate
TNR (SPC)	$TNR = \frac{TN}{N} = 1 - FPR = \frac{TN}{TN + FP}$ (3.26)	**true negative rate**, **specificity**, selectivity [61]
PPV	$PPV = \frac{TP}{TP + FP} = 1 - FDR$ (3.27)	**positive predictive value** [56], precision [50]
NPV	$NPV = \frac{TN}{TN + FN} = 1 - FOR$ (3.28)	**negative predictive value** [56]
FDR	$FDR = \frac{FP}{FP + TP} = 1 - PPV$ (3.29)	**false discovery rate** [46]

define the formulas for calculating the probabilities of its outcomes (see tables 3.4–3.6). The TPR probability can be written as

$$P_{TPR} = \mathbb{P}(1,\ x \in C_1), \tag{3.44}$$

which means that if object x belongs to class C_1, this indicator estimates the probability that the binary classifier assigns object x to this class. The probability of FPR is written as

$$P_{FPR} = \mathbb{P}(1,\ x \in C_0), \tag{3.45}$$

which means the probability that an object belonging to class C_0 will be mistakenly assigned to class C_1.

Table 3.5: Additional definitions and formulas for calculating the probabilities of binary classifier outcomes (part 2 of 3).

Notation	Formula	Deciphering the notation and alternative terms.
FOR	$FOR = \frac{FN}{FN+TN} = 1 - NPV$ (3.30)	**false omission rate** [56]
LR+	$LR+ = \frac{TPR}{FPR}$ (3.31)	**positive likelihood ratio** [53]
LR-	$LR- = \frac{FNR}{TNR}$ (3.32)	**negative likelihood ratio** [53]
PT	$PT = \frac{\sqrt{TPR(-TNR+1)}+TNR-1}{TPR+TNR-1} = \frac{\sqrt{FPR}}{\sqrt{TPR}+\sqrt{FPR}}$ (3.33)	**prevalence threshold** [61]
TS (CSI)	$TS = \frac{TP}{TP+TN+FP}$ (3.34)	Jaccard index **threat score**, **critical success index** [51]
PRV	$PRV = \frac{P}{P+N}$ (3.35)	**prevalence** [58]
ACC	$ACC = \frac{TP+TN}{P+N} = \frac{TP+TN}{TP+TN+FP+FN}$ (3.36)	**accuracy** [42]

Typically, the working principle of a binary classifier is based on comparing the measurement of x with some fixed threshold c. It follows that the previous two expressions can be rewritten and combined into a system.

$$\begin{cases} P_{TPR} = \mathbb{P}(x > c,\ x \in C_1) \\ P_{FPR} = \mathbb{P}(x > c,\ x \in C_0) \end{cases} \tag{3.46}$$

It follows that the ROC curve is a diagram

$$P_{FPR}(c),\ P_{TPR}(c), \tag{3.47}$$

thus, drawing the curve means changing the value of threshold c.

Let's consider the example [17]. Let's take $f(x \in C_0) = \mathcal{N}(0,1)$ and $f(x \in C_1) = \mathcal{N}(2,1)$ as probability density functions C_0 and C_1,

Table 3.6: Additional definitions and formulas for calculating the probabilities of binary classifier outcomes (part 3 of 3).

Notation	Formula	Deciphering the notation and alternative terms.
BA	$BA = \frac{TPR + TNR}{2}$ (3.37)	**balanced accuracy**
F1 score	$F_1 = 2 \times \frac{PPV \times TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$ (3.38)	F1 score is the harmonic mean of precision and sensitivity [45]
MCC (ϕ or r_ϕ)	$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$ (3.39)	**Matthews correlation coefficient**,**phi coefficient** [55]
FM	$FM = \sqrt{\frac{TP}{TP + FP} \times \frac{TP}{TP + FN}} = \sqrt{PPV \times TPR}$ (3.40)	Fowlkes–Mallows index [48]
BM	$BM = TPR + TNR - 1$ (3.41)	**bookmaker informedness**, informedness [67]
MK (δP)	$MK = PPV + NPV - 1$ (3.42)	**markedness**, deltaP [54]
DOR	$\frac{LR+}{LR-}$ (3.43)	**diagnostic odds ration** [44]

respectively. Next we build the ROC curve step by step using the Python language. At the first step, consider diagram 3.2, built using the code given in script 3.1. The area shaded blue shows the probability of FPR, i. e., false-positive significance detection, while the area shaded green shows the probability density of TPR, i. e., correct significance detection. The ROC curve shows the values of these very indicators. The vertical dashed line is the sensitivity threshold c. In this situation it is at 0 on the abscissa axis. If it is moved to 1, the area under the FPR curve (blue) will significantly decrease, i. e. the probability of false-positive detection will decrease, but the TPR area (green) will decrease as well, which means an increase in the probability of false-negative results. This situation is illustrated in Diagram 3.3.

Listing 3.1: Plotting TPR and FPR probability density functions

```
# Import Libraries
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats

# Plot
f0 = stats.norm(0, 1)
f1 = stats.norm(2, 1)
fig, ax = plt.subplots()
xi = np.linspace(-2, 5, 100)
ax.plot(xi, f0.pdf(xi), label=r'$f(x|C_0)$')
ax.plot(xi, f1.pdf(xi), label=r'$f(x|C_1)$')
ax.legend(fontsize=16, loc=(1, 0))
ax.set_xlabel(r'$x$', fontsize=18)
ax.vlines(0, 0, ax.axis()[-1] * 1.1, linestyles='--',
lw=3.)
ax.fill_between(xi, f1.pdf(xi), where=xi > 0, alpha=.3,
color='g')
ax.fill_between(xi, f0.pdf(xi), where=xi > 0, alpha=.3,
color='b')

# Save to .pdf
plt.savefig('Plot-ROC-step-1.pdf', bbox_inches='tight')
```

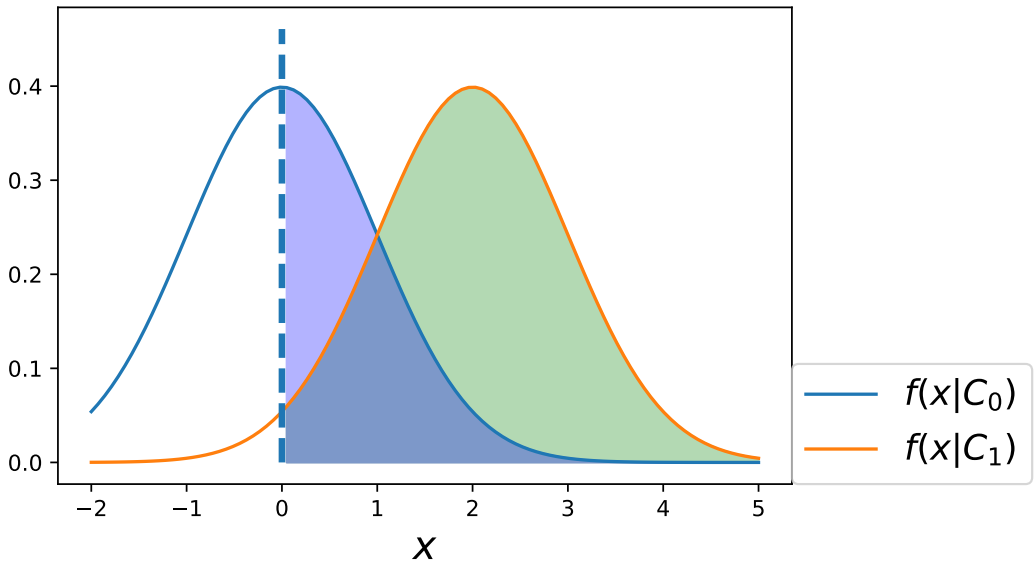

Figure 3.2: Diagram of TPR and FPR probability distribution densities at threshold 0.

As you can see from the diagrams above, increasing the threshold leads to the loss of a part of both true-positive and false-positive results, while decreasing it leads to an increase in the number of fixations of the feature presence (both true and false). In extreme cases, too low a threshold value will lead to the fact that all results will be interpreted as positive, too high — to a zero number of observations in which the feature was detected. The task of ROC analysis is to choose a rational threshold value.

Let's add the ROC curves corresponding to thresholds 0 and 1 to the already existing diagrams. And also create an interactive diagram using the code from script 3.2. The PDF format does not allow you to add such interactive elements, so let's consider cases with fixed values of 0 and 1, shown in Diagrams 3.4 and 3.5, respectively. The left side of each of them shows the already familiar probability density function graphs for the TPR and FPR distributions. The right part shows the ROC curve and the point corresponding to the set threshold value. It is easy to guess that the x-coordinate of the point matches the area under the FPR curve, and the y-coordinate matches the area

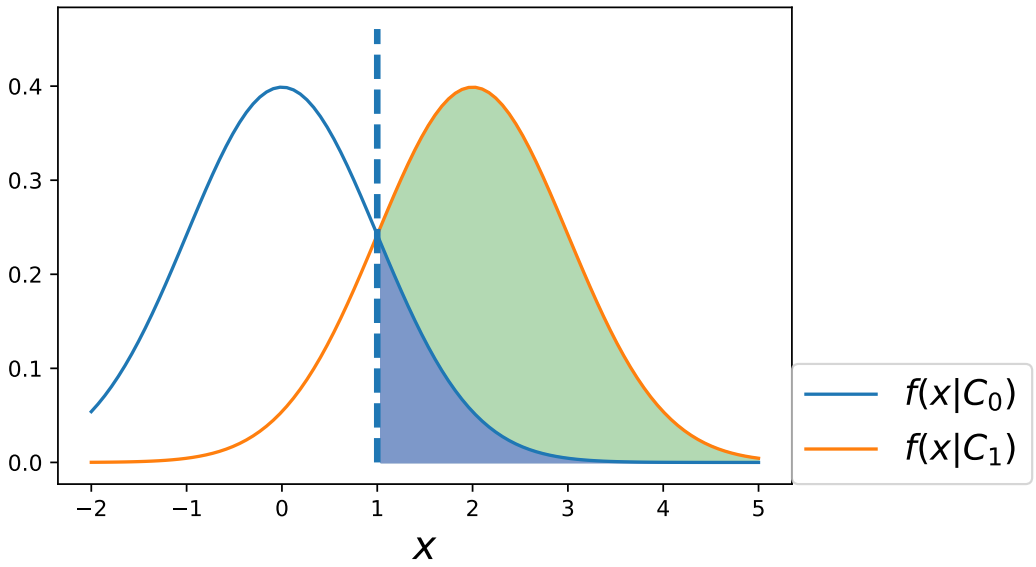

Figure 3.3: Diagram of TPR and FPR probability distribution densities at threshold 1.

under the TPR curve. Increasing the threshold value entails shifting the point to the left, decreasing it to the right.

The better the binary classifier itself, the closer to the upper left corner will be the ROC curve corresponding to it, because in this case a high TPR value will be combined with a low FPR value. The binary classifier, which works as well (actually badly) as the coin flip guessing algorithm (in case the coin is "fair"), gives a ROC curve, which is a straight line between (0,0) and (1,1). In this case, the left part of the diagram will show a complete overlap of TPR and FPR probability density function curves. Such a case is shown in Diagram 3.4. For self-practice, you can use Script 3.2 by running it in the Jupyter Lab environment, which allows you to use the interactive features of the browser.

3.4.1 The concept of AUC and its calculation

As the name implies, the AUC is the area under the ROC curve bounded by the point corresponding to a given threshold value. In the

Listing 3.2: Build an interactive graph of TPR and FPR distribution density and its corresponding ROC curve for a given threshold value

```
# Import Libraries
%matplotlib inline
from ipywidgets import interact
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats

# Plot
f0 = stats.norm(0, 1)
f1 = stats.norm(2, 1)
fig, ax = plt.subplots()
xi = np.linspace(-2, 5, 100)
ax.plot(xi, f0.pdf(xi), label=r'$f(x|C_0)$')
ax.plot(xi, f1.pdf(xi), label=r'$f(x|C_1)$')
ax.legend(fontsize=16, loc=(1, 0))
ax.set_xlabel(r'$x$', fontsize=18)
ax.vlines(0, 0, ax.axis()[-1] * 1.1, linestyles='--',
lw=3.)
ax.fill_between(xi, f1.pdf(xi), where=xi > 0, alpha=.3,
color='g')
ax.fill_between(xi, f0.pdf(xi), where=xi > 0, alpha=.3,
color='b')

# Plot ROC-curve and make all interactive
def plot_roc_interact(c=0):
xi = np.linspace(-3,5,100)
fig,axs = plt.subplots(1,2)
fig.set_size_inches((10,3))
ax = axs[0]
ax.plot(xi,f0.pdf(xi),label=r'$f(x|C_0)$')
ax.plot(xi,f1.pdf(xi),label=r'$f(x|C_1)$')
ax.set_xlabel(r'$x$',fontsize=18)
ax.vlines(c,0,ax.axis()[-1]*1.1,linestyles='--',lw=3.)
ax.fill_between(xi,f1.pdf(xi),where=xi>c,alpha=.3,color='g')
ax.fill_between(xi,f0.pdf(xi),where=xi>c,alpha=.3,color='b')
ax.axis(xmin=-3,xmax=5)
crange = np.linspace(-3,5,50)
ax=axs[1]
ax.plot(1-f0.cdf(crange),1-f1.cdf(crange))
ax.plot(1-f0.cdf(c),1-f1.cdf(c),'o',ms=15.)
ax.set_xlabel('False-alarm probability')
ax.set_ylabel('Detection probability')
```

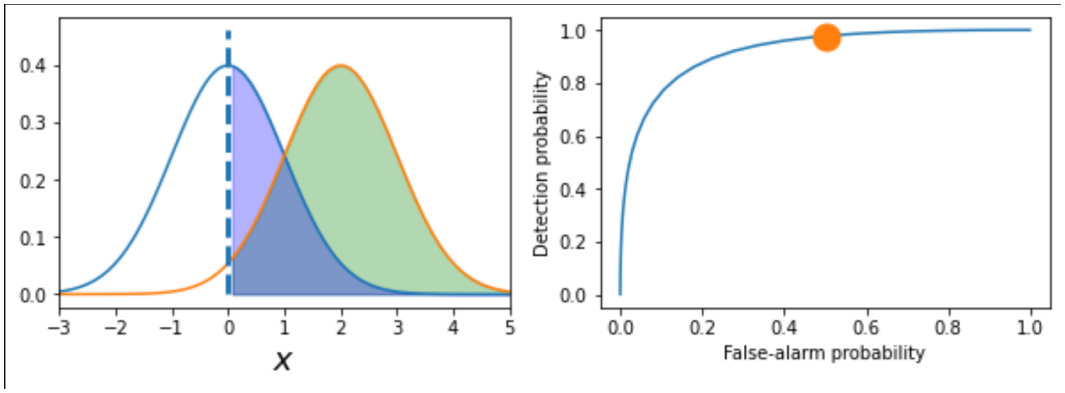

Figure 3.4: Diagram of TPR and FPR probability distribution densities at threshold 0.

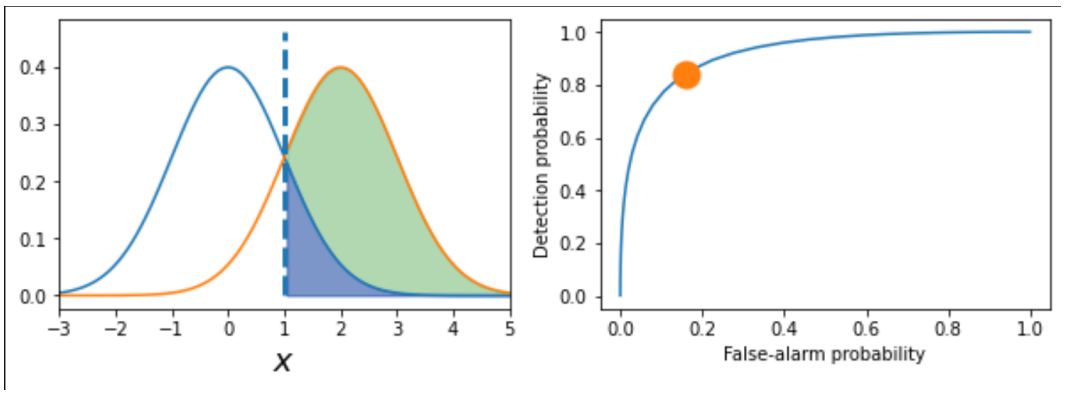

Figure 3.5: Diagram of TPR and FPR probability distribution densities at threshold 1.

normalized space in which the ROC curve is usually plotted, the AUC value is equivalent to the probability that the classifier assigns a higher weight to a randomly chosen positive entity than to a randomly chosen negative entity. The AUC does not depend on a specific threshold value, because the ROC curve is constructed by fitting it. This means that the AUC is calculated by integrating over the thresholds. The AUC is given by the expression:

$$AUC = \int P_{TPR}(P_{FPR}) dP_{FPR}. \tag{3.48}$$

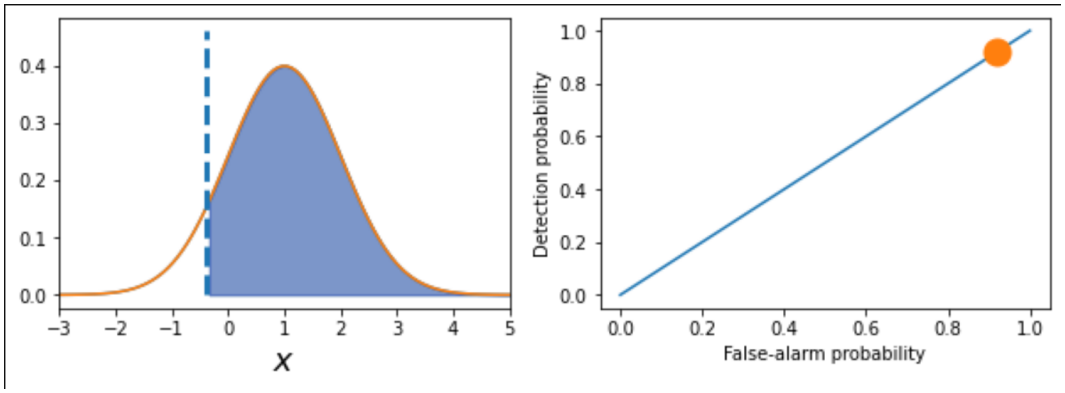

Figure 3.6: Diagram of probability densities of TPR and FPR probability distributions at equal mean.

The step-by-step calculation of the AUC is as follows.

$$P_{TPR}(c) = 1 - F_1(c), \tag{3.49}$$

where F_1 is the cumulative density function for C_1. Similarly calculate

$$P_{FPR}(c) = 1 - F_0(c), \tag{3.50}$$

where F_0 is the cumulative density function for C_0.

Let us take some particular value of c^* to which a certain $P_{FPR}(c^*)$ corresponds. In other words, it corresponds to the probability that a random element x_0 belonging to class C_0 is greater than the threshold value c^*, i.e.

$$P_{FPR}(c^*) = \mathbb{P}(x_0 > c^* | x_0 \in C_0). \tag{3.51}$$

Then, reasoning similarly with respect to TPR, we get

$$P_{TPR}(c^*) = \mathbb{P}(x_1 > c^* | x_1 \in C_1). \tag{3.52}$$

Next, based on the fact that the AUC is realized through an integral, we select its value so that the distribution of c^* matches the distribution of F_0. In this case, P_{TPR} is an independent random variable with a corresponding expectation in the form of

$$\mathbb{E}(P_{TPR}) = \int P_{TPR} dP_{FPR} = AUC. \tag{3.53}$$

It is now possible to formulate a definition for the AUC.

AUC — is the expected probability that element $x_1 \in C_1$ will be assigned to C_1 with higher probability than element $x_0 \in C_0$. Thus,

$$1 - F_1(t) > 1 - F_0(t) \forall t. \tag{3.54}$$

The wording "for any t" means that $1 - F_1(t)$ is *stochastically* greater than $1 - F_0(t)$. The latter circumstance is key in terms of the relationship of the AUC to the U-test, which will be shown later.

3.4.2 Relation between U-test and AUC

A fairly detailed description of the U-test was given earlier. This subsection contains only brief information about it, which is directly relevant to the question of its relationship to the AUC.

The U-test is a non-parametric test that allows you to test whether two samples belong to the same distribution. His basic idea is that if there is no difference between two classes, then combining them into one larger class (set) and then calculating any statistic for the new larger class will give an unbiased estimate for any of the initial classes. In other words, if there is no difference in the distribution of the two samples, combining them and assuming that the actually observed data from the two samples represent only one of the equal-valued variants of the moving observations means that there is no difference in any statistical estimate for any of the moving variants relative to the other, and relative to the combined set.

Let's suppose that we need to compare two samples using the median, the mean, or some other measure of central tendency. In terms of cumulative distribution functions for the two populations, in the case of $H0$ we have the following:

$$H_0 : F_X(t) = F_Y(t), \quad \forall t, \tag{3.55}$$

which indicates that all observations belong to the same distribution. Then an alternative hypothesis is that

$$H_1 : F_X(t) < F_Y(t), \quad \forall t, \tag{3.56}$$

Listing 3.3: Calculation of the p-value for the test data

```
print('p-value:',stats.wilcoxon(f1.rvs(30),
f0.rvs(30))[1])
```

which is possible, in particular, in the case of the existence of a shift of one distribution relative to the other. In this case, the samples $X_{i\,i=1}^{n}$, $X_{j\,j=1}^{m}$ represent independent groups of observations. In this case, the size of the samples may vary.

The test technique consists of combining two samples into one set and assigning ranks to each item within it. The U-statistic is the sum of the ranks for the set X. If the value of the statistic is small enough, it means that the distribution of set X is stochastically shifted to the left relative to the distribution of set Y, i. e. $F_X t < F_Y t$.

Since with a sufficiently large number of observations (20 or more) the distribution of U-statistics is well approximated by the normal distribution, the p-value is suitable for assessing significance. Let's calculate it using the Python language according to the script 3.3. The p-value is 1.9729484515803686e-05, which is less than the significance level (0.05), so we can reject the null hypothesis 3.55. Since the data were randomly generated, if the experiment is repeated, the particular p-value will differ from that obtained when writing this paper. However, it will always be below the threshold because of the parameters set in the algorithm.

The U-statistic can be written as follows:

$$U = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \mathbb{1}(Y_j > X_i), \tag{3.57}$$

where $\mathbb{1}(Y_j > X_i)$ is the indicator (characteristic) function showing that the statistic (for the discrete case) estimates the probability that Y is stochastically greater than X. Thus, this correspondence means that its value is equal to the AUC. The relationship between the AUC and the U-test is in a similar sense: checking the stochastic excess value of observations belonging to one sample relative to observations be-

longing to another sample.

3.4.3 Practice of ROC analysis and AUC calculation.

This subsection is not required reading if the goal is only the practical implementation of the U-test itself. However, it gives an insight into machine learning methods that are not related to the so-called *frequentist statistics* to which the U-test itself belongs, and shows the relationship between these areas of data analysis. In addition, it will provide sufficient knowledge to perform a ROC analysis as such, which may be useful in other situations that an appraiser may encounter in his or her practice.

3.4.3.1 Plotting the ROC curve

ROC analysis and in particular the construction of ROC curves are widely used to find a compromise between the *sensitivity* and *specificity* of a binary classifier. Most of the classifiers used in machine learning produce a result in the form of a quantification that a given object has a "positive" feature value. Some threshold value is needed to convert such a quantitative assessment into a concrete "yes" or "no" prediction. In his case, observations with a score above this threshold will be classified as "positive", below as "negative". Different thresholds provide different levels of sensitivity and specificity. Setting a relatively high threshold value provides a conservative approach to the issue of classifying a particular case as "positive", which reduces the likelihood of false positives. At the same time, this increases the risk of missing the observed positive values, i. e., it reduces the level of true positive classification results. A relatively low threshold value provides a more liberal approach to classifying observations as "positive". This reduces specificity (increases the number of false negatives) and increases sensitivity (increases the number of true positives). The ROC curve shows the ratio of true positives to false positives, giving an overview of the entire spectrum of such trade-offs. There are many R language libraries that plot ROC curves and calculate metrics for ROC analysis. In this case, to better understand the essence of ROC analysis, some actions

Listing 3.4: Creating a function to calculate TPR and FPR

```
# create own function for ROC
appraiserRoc <- function(labels, scores){
labels <- labels[order(scores, decreasing=TRUE)]
data.frame(TPR=cumsum(labels)/sum(labels),
FPR=cumsum(!labels)/sum(!labels), labels)
}
```

will be performed by writing our own functions. The following will show an algorithm for constructing a ROC curve based on a set of real outcomes and their corresponding estimates. The calculation involves two steps:

- sort the observed outcomes in descending order by their predicted scores;
- calculation of total true positive (TPR) and true negative (TNR) scores for ordered observed outcomes.

Let's create an appropriate function (script 3.4). This function has two inputs:

- *labels* — Boolean vector containing actual classification data;
- *scores* — a vector of real numbers containing data about the scores predicted by some classifier.

Since only two classification outcomes are possible, the labels vector can only contain *TRUE* or *FALSE* values (or *1* and *0* depending on the analyst's preference). A sequence of such binary values can be interpreted as a set of instructions for a turtle graphics [40]. There is one important feature: in this case the turtle has a compass and receives instructions for absolute directions of movement: "to the north" or "to the east" instead of relative "to the right" and "to the left". The turtle starts its movement from the starting point with coordinates (0,0) and makes its way on the plane according to the sequence

of instructions. When a *TRUE* command is received, it takes one step north, i. e., in the positive direction of the y-axis, and when a *FALSE* command is received, it takes one step east, i. e., in the positive direction of the x-axis. The length of the steps is chosen in such a way that if all *TRUE (1)* commands are received consecutively, the turtle will be at a point with coordinates (0,1), all *FALSE (0)* commands at a point with coordinates (1,0). Thus, the length of the step "to the north" may be different from the length of the step "to the east". The path in the plane is determined by the order of the *TRUE (1)* and *FALSE (0)* commands and always ends at (1,1).

Advancing the turtle through the bits of the instruction string is an adjustment of the classification threshold to less and less stringent. Once the turtle has passed the bit, it means that it has decided to classify that bit as "positive". If this bit was actually "positive", it is a true positive, if it was actually "negative" it is a false positive. The y-axis shows the TPR, calculated as the ratio of the number of positive results detected to this time to the total number of actual positive results. The x-axis shows the (FPR), calculated as the ratio of the number of currently detected positive results to the total number of actual negative results. The vectorized implementation of this logic uses cumulative sums (the **cumsum** function) instead of going through the values one by one, although that is what the computer does at a lower level.

The ROC curve calculated in this way is actually a step function. With a very large number of positive and negative cases, these steps are very small, and the curve looks smooth. In this case, with a really large number of observations, the construction of each point is difficult. As a consequence, in practice, most ROC curve functions used for practical purposes contain additional steps and often use some form of approximation.

As an example, consider a situation in which an appraiser evaluates parts manufactured by an enterprise. Some of the parts are known to be of good quality and some are defective. The valuation of quality parts is carried out on the basis of cost market approaches in the usual manner. And defective parts are valued at a scrap value. In this case, it is necessary to assign each part to one or another category. There

is some feature x, which can be measured by the appraiser. And there is also some feature y, which cannot be measured by the appraiser. The value of the feature y allows you to classify parts as quality or defective. It is also known that there is some finitary relation function between features x and y. Thus, knowing the value of x, we can infer the value of y with some probability. In this case, it is advisable to take a certain sample of parts. Then, together with the specialists of the customer company, measure the values of features *0* and y for each element of this sample.

We will use simulated data to consider the example. There is some input feature x that is linearly related to the implicit result y. This relationship implies the presence of some randomness. The y-value shows whether the part exceeds the tolerance requirements. If so, it should be classified as defective. The algorithm used in this paper involves the following steps:

- create the **sim_parts_data** function that generates data according to certain rules and sets the "$y > 100$" threshold value to classify parts as defective.
- create the dataframe **parts_data** with this function;
- create the **test_set_idx** rule, whereby 80 % of the data is randomly assigned to the training sample, and 20 % to the test sample;
- applying the rule **test_set_idx** to data **parts_data**;
- create training (**«training_set»**) and testing (**«test_set»**) subsamples;
- plot the diagram showing the distribution of observations from the training sample.

To implement the above algorithm, the code from script 3.5 was used.

The result was diagram 3.7. As you can see, if the value of the parameter x is less than 15, all dots are red, which means that the

Listing 3.5: Creation and primary visualization of data on quality and defective parts

```
# Sample of ROC-analysis

# enable libraries
library(ggplot2)
library(dplyr)
library(pROC)

#set seed
set.seed(19190709)

# create own function for ROC
appraiserRoc <- function(labels, scores){
labels <- labels[order(scores, decreasing=TRUE)]
data.frame(TPR=cumsum(labels)/sum(labels),
FPR=cumsum(!labels)/sum(!labels), labels)
}

# create function
sim_parts_data <- function(N, noise=100){
x <- runif(N, min=0, max=100)
y <- 122 - x/2 + rnorm(N, sd=noise)
bad_parts <- factor(y > 100)
data.frame(x, y, bad_parts)
}

# create dataset
parts_data <- sim_parts_data(2000, 10)

# create rule for test subset
test_set_idx <- sample(1:nrow(parts_data),
size=floor(nrow(parts_data)/4))

# create training and test subsets
test_set <- parts_data[test_set_idx,]
training_set <- parts_data[-test_set_idx,]
```



```
# plot graph
test_set %>%
ggplot(aes(x=x, y=y, col=bad_parts)) +
scale_color_manual(values=c("green", "red")) +
geom_point() +
ggtitle("Bad parts related to x")
```

parts are defective. Above 96 are green, which means that the parts are of good quality. If the value is higher than 96, they are green, which means that the parts are of good quality. Between these values is an area of uncertainty, the right side of which is dominated by green dots, and the left side by red dots.

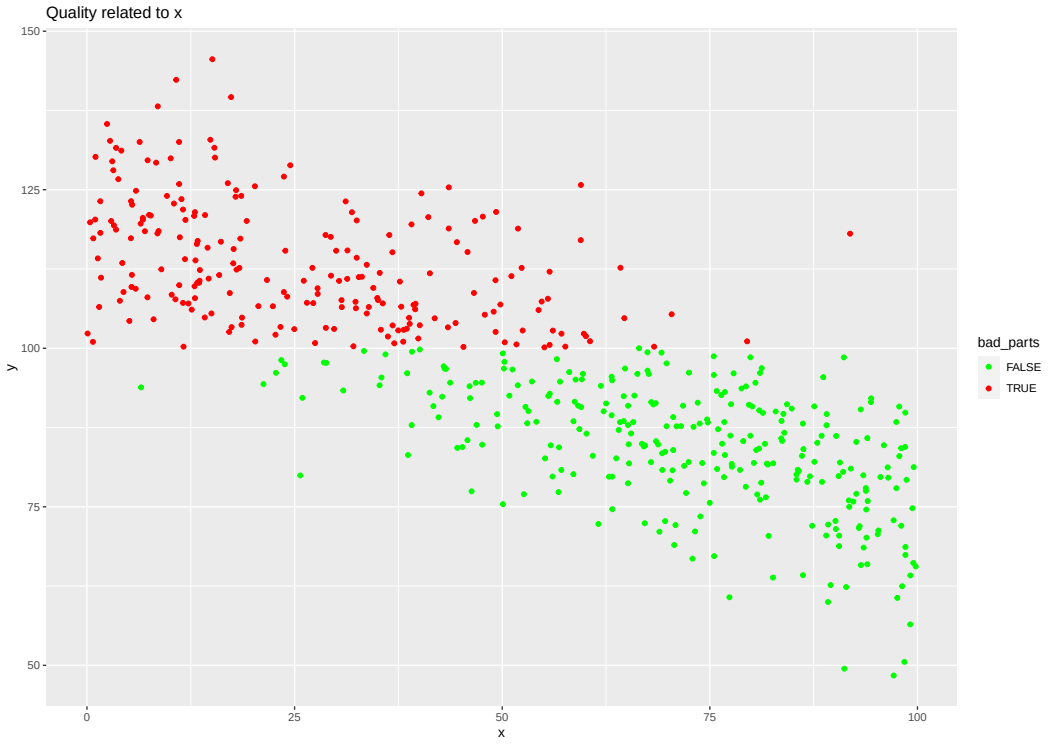

Figure 3.7: Diagram of the distribution of parts with respect to the parameter x.

The training sub-sample will be used to create a logistic regression model based on the values of the attribute x, which allows you to assign a particular part to quality or defective. This model will be used to assign scores to the observations in the training sample. In the future, these scores will be used to construct the ROC curve together with the true labels. Recall that the ROC curve is plotted for observations with known values of parameters x and y. This ROC curve is then applied to the entire set of objects for which x values are known but y values are unknown. The scores themselves as well as the x and y values are

Listing 3.6: Comparing ”link” and ”response” predictions

```
fit_glm <- glm(bad_parts ~ x, training_set,
family=binomial(link="logit"))

glm_link_scores <- predict(fit_glm, test_set, type="link")

glm_response_scores <- predict(fit_glm, test_set,
type="response")

score_data <- data.frame(link=glm_link_scores,
response=glm_response_scores,
bad_parts=test_set$bad_parts,
stringsAsFactors=FALSE)

score_data %>%
ggplot(aes(x=link, y=response, col=bad_parts)) +
scale_color_manual(values=c("green", "red")) +
geom_point() +
geom_rug() +
ggtitle("Both link and response scores put cases in the
same order")
```

not displayed on the graph and are only used for sorting labels. Two different classifiers sorting labels in the same order will give identical ROC curves regardless of the absolute values of the scores. This can be seen by constructing an ROC curve based on ”response” or ”link” predictions from a logistic regression model. The ”response” scores were mapped to a (0, 1) scale using a Sigmoid function[62], the ”link” scores were left untransformed. In this case, the points showing specific observations are ordered in the same way. To test this hypothesis, we use the code 3.6. As you can see in Figure 3.8, the order of the dots is the same for ”link” and ”response”.

Let’s go directly to the construction of the ROC curve. We use

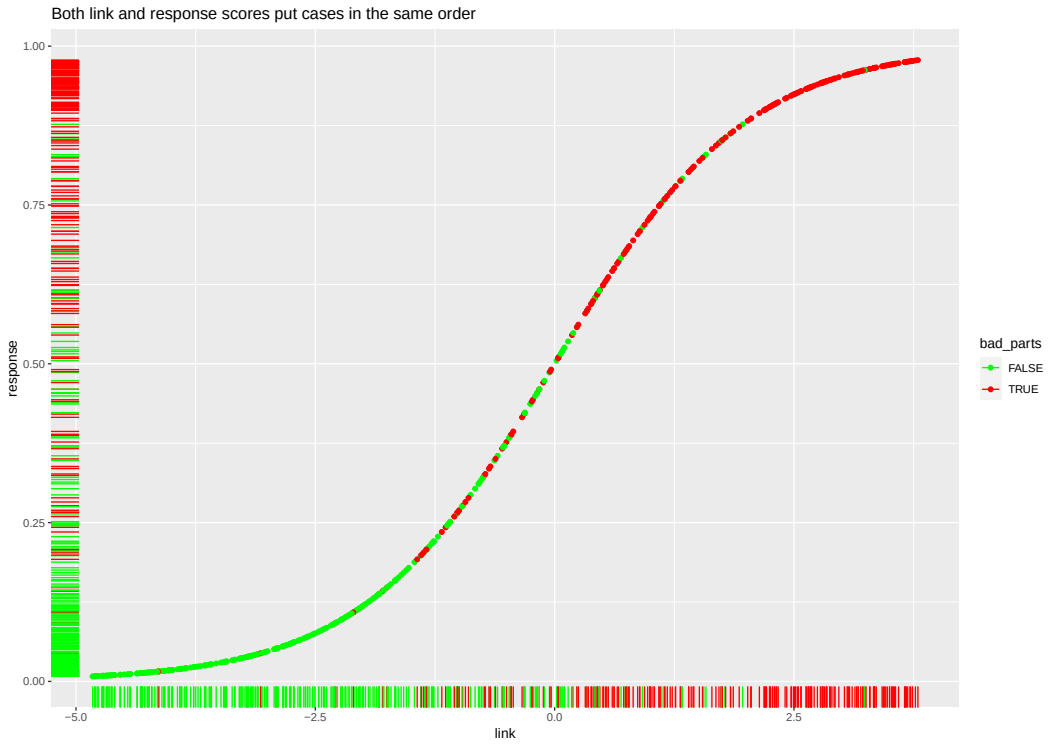

Figure 3.8: Comparison of the order of points for ”link” and ”response”.

both the ready function from the ”pROC” package and the previously created **”appraiserRoc”** function (see script 3.7). The result of the first is represented as an orange curve, the second as circles of red for defective parts and black for quality parts (see Diagram 3.9). It is not difficult to guess that the red dot corresponded to the ”North” command and the black dot to the ”East” command. Since the library function and the own function perform the same actions, the two curves are identical.

Note that the ”Specificity” scale is plotted on the abscissa axis, not the FPR, so the values on the axis are inverted. Since, according to Table 3.4, ”$Specificity = 1 - FPR$” we can talk about the mutual unambiguity of these indicators. Consequently, when plotting the ROC curve any of them can be used. This version of the scale display was self-selected by the **roc** function from the ”pRoc” library. If the user does not set his settings, the function chooses to display

Listing 3.7: Plotting the ROC curve using library and own functions

```
# plot ROC
plot(roc(test_set$bad_parts, glm_response_scores,
direction="<"),
col="orange", lwd=3, main="The turtle finds its way",
xlim = c(1, 0))
glm_simple_roc <-
appraiser_roc(test_set$bad_parts=="TRUE", glm_link_scores)
with(glm_simple_roc, points(1 - FPR, TPR, col=1 + labels))
```

the scale so that the AUC value is always greater than 0.5. This calculation is based on which group (quality parts, defective parts) has a higher median score. Since the **appraiserRoc** function is of course not that smart, a simple subtraction was performed during its use, making it possible to build a joint diagram.

This approach has one limitation: based on the prognostic nature of the ordering of outcomes, it does not allow correct processing of information if the sequence consists of identical estimates. "Turtle" assumes that the order of the labels matters, but there is no meaningful order in the situation of the same scores. These areas should be displayed with a diagonal line, but not the traditional steps.

Consider an example where a diagonal is the only adequate way to plot a ROC curve. To do this, create an extremely unbalanced data set in which only 1 % of the observations are "positive". In this case, the result of the prediction will always be negative. Since all scores will be the same, there is no need for any ordering. The **roc** function from the "pRoc" package correctly recognizes such situations and draws a diagonal line (1,0; 0,1). In doing so, the turtle assumes that the order of scores has some significance, and moves between these points along a random trajectory, alternating between "north" and "east" directions. The code calling the construction of such a ROC curve is given in script **??**. In Diagram **??**, the black diagonal line was plotted by the library function, while the blue dashed line was

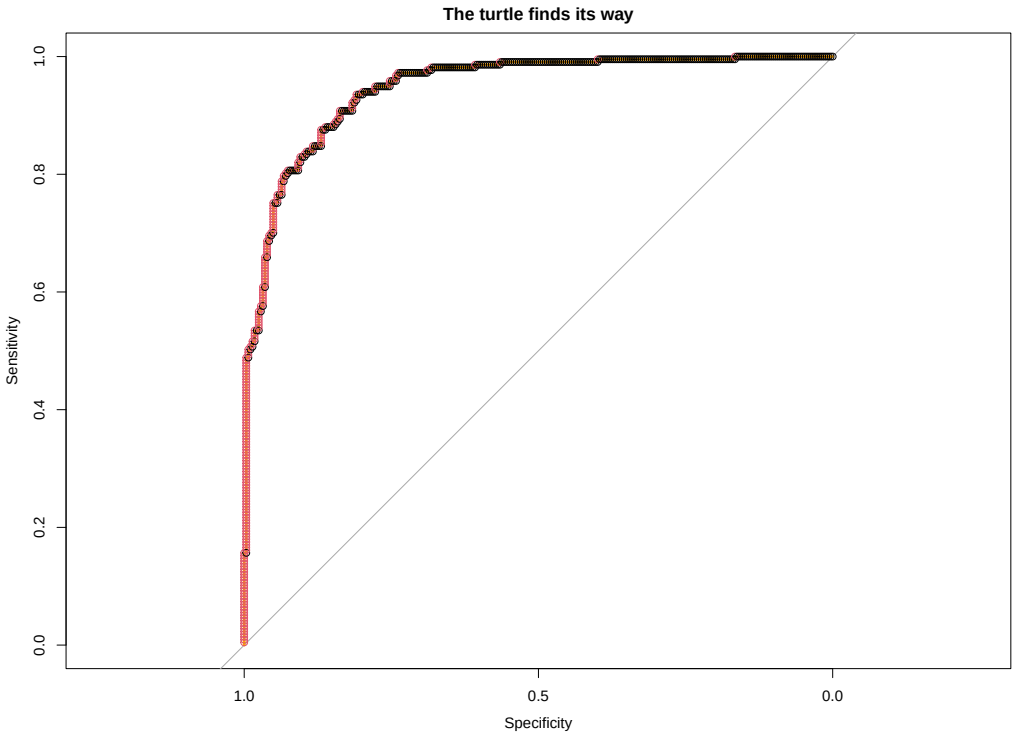

Figure 3.9: Identical ROC curves plotted with library and own functions.

plotted by our own previously written **appraiserRoc** function. As you can see, the library function correctly determined the case of identical estimates, while applying our own function resulted in random turtle wanderings.

The greater the value of N, the closer to the diagonal the turtle will wander. Greater unbalance requires more points in order for the path to run roughly close to the diagonal. In less extreme cases, the emergence of diagonal sections is possible, in particular, in the case of rounding of estimates, leading to equality of some of them.

To further familiarize yourself with the topic of constructing ROC curves, we can recommend studying this theoretical material [8], as well as practice on the online simulator [33].

,æ„å,

,æ„å„æ„å,

Bibliography

[1] Donald R. Mann Henry B.and Whitney. "On a test of whether one of two random variables is stochastically larger than the other". In: *The annals of mathematical statistics* (1947), pp. 50–60.

[2] Morton B. Brown and Alan B. Forsythe. "Robust tests for the equality of variances". In: *Journal of the American Statistical Association* 346.69 (1974), pp. 364–367. DOI: 10.1080/01621459.1974.10482955. URL: https://www.tandfonline.com/doi/abs/10.1080/01621459.1974.10482955 (visited on 06/06/2022).

[3] William J. Conover. *Practical Nonparametric Statistics*. John Wiley and Sons, 1980.

[4] Erich L Lehamnn. "Elements of Large Sample Theory". In: *Springer* (1999), p. 176.

[5] Eiiti Kasuya. "Mann–Whitney U test when variances are unequal". In: *Animal Behaviour* 6.61 (2001), pp. 1247–1249. DOI: 10.1006/anbe.2001.1691. URL: https://www.sciencedirect.com/science/article/abs/pii/S0003347201916914 (visited on 06/06/2022).

[6] Michael C Mozer. *Optimizing classifier performance via an approximation to the Wilcoxon-Mann-Whitney statistic*. English. 2003.

[7] Corinna Cortes and Mehryar Mohri. "AUC optimization vs. error rate minimization". In: *Advances in neural information processing systems* 16.16 (2004), pp. 313–320.

[8] Tom Fawcett. "An introduction to ROC analysis". In: *Pattern Recognition Letters* 27 (2005-12-19), pp. 861–874. URL: file:///home/kaarlahti/Downloads/ROCintro.pdf (visited on 06/25/2022).

[9] . . . . 2006.

[10] Dennis D. Boos and L. A. Stefanski. *Essential Statistical Inference*. English. Springer, 2013.

[11] The IFRS Foundation. *IFRS 13 Fair Value Measurement*. UK, London: The IFRS Foundation, 2016-01-31. URL: http://eifrs.ifrs.org/eifrs/bnstandards/en/IFRS13.pdf (visited on 06/10/2020).

[12] E. Brunner and Arne C Bathke. *Rank and pseudo-rank procedures for independent observations in factorial designs: Using R and SAS*. English. Springer International Publishing, 2018. ISBN: 978-3-030-02912-8.

[13] Julian D. Karch. "Psychologists Should use Brunner-Munzel's instead of Mann-Whitney's U-test as the Default Nonparametric Procedure". In: *Advances in Methods and Practices in Psychological Science* 2.4 (2021). ISSN: 2515-2459. DOI: 10.1177/2515245921999602. URL: https://journals.sagepub.com/doi/10.1177/2515245921999602 (visited on 06/06/2022).

[14] Royal Institution of Chartered Surveyors (RICS). *RICS Valuation — Global Standards*. English. UK, London: RICS, 2021-11-30. URL: https://www.rics.org/uk/upholding-professional-standards/sector-standards/valuation/red-book/red-book-global/ (visited on 05/11/2022).

[15] International Valuation Standards Council. *International Valuation Standards*. 2022-01-31. URL: https://www.rics.org/uk/upholding-professional-standards/sector-standards/valuation/red-book/international-valuation-standards/.

[16] K. A. Murashev. “Practical application of the Mann-Whitney-Wilcoxon test (U-test) in valuation”. English, Spanish, Russian, Interslavic. In: (2022-05-15). URL: https://github.com/Kirill-Murashev/AI_for_valuers_book/tree/main/Parts&Chapters/Mann-Whitney-Wilcoxon (visited on 05/15/2022).

[17] *AUC-Derivation.ipynb*. URL: https://colab.research.google.com/github/unpingco/Python-for-Signal-Processing/blob/master/AUC_Derivation.ipynb#scrollTo=BgrH5C49LqMx (visited on 06/14/2022).

[18] Creative Commons. *cc-by-sa-4.0*. URL: https://creativecommons.org/licenses/by-sa/4.0/ (visited on 01/27/2021).

[19] *F-test*. URL: https://en.wikipedia.org/wiki/F-test (visited on 06/06/2022).

[20] *Fligner-Policello test in R*. URL: https://search.r-project.org/CRAN/refmans/RVAideMemoire/html/fp.test.html (visited on 06/06/2022).

[21] Python Software Foundation. *Python site*. . Python Software Foundation. URL: https://www.python.org/ (visited on 08/17/2021).

[22] The Document Foundation. *LibreOffice Calc*. . URL: https://www.libreoffice.org/discover/calc/ (visited on 08/20/2021).

[23] *GeoGebra official site*. URL: https://www.geogebra.org/ (visited on 08/26/2021).

[24] Bob Horton. *AUC Meets the Wilcoxon-Mann-Whitney U-Statistic*. URL: https://blog.revolutionanalytics.com/2017/03/auc-meets-u-stat.html (visited on 07/14/2022).

[25] Bob Horton. *Calculating AUC: the area under a ROC Curve*. URL: https://blog.revolutionanalytics.com/2016/11/calculating-auc.html (visited on 07/14/2022).

[26] Bob Horton. *ROC Curves in Two Lines of R Code*. URL: https://blog.revolutionanalytics.com/2016/08/roc-curves-in-two-lines-of-code.html (visited on 07/14/2022).

[27] *If p-value is exactly equal to 0.05, is that significant or insignificant?* URL: https://www.researchgate.net/post/If_p-value_is_exactly_equal_to_005_is_that_significant_or_insignificant.

[28] *Jupyter site*. URL: https://jupyter.org (visited on 05/13/2022).

[29] Machinelearning.ru. *U- - .* Russian. URL: http://www.machinelearning.ru/wiki/index.php?title=%D0%9A%D1%80%D0%B8%D1%82%D0%B5%D1%80%D0%B8%D0%B9_%D0%A3%D0%B8%D0%BB%D0%BA%D0%BE%D0%BA%D1%81%D0%BE%D0%BD%D0%B0-%D0%9C%D0%B0%D0%BD%D0%BD%D0%B0-%D0%A3%D0%B8%D1%82%D0%BD%D0%B8 (visited on 05/14/2022).

[30] Machinelearning.ru. . URL: http://www.machinelearning.ru/wiki/index.php?title=%D0%93%D0%B8%D0%BF%D0%BE%D1%82%D0%B5%D0%B7%D0%B0_%D1%81%D0%B4%D0%B2%D0%B8%D0%B3%D0%B0 (visited on 05/15/2022).

[31] Machinelearning.ru. . Russian. URL: http://www.machinelearning.ru/wiki/index.php?title=%D0%9A%D1%80%D0%B8%D1%82%D0%B5%D1%80%D0%B8%D0%B9_%D0%A3%D0%B8%D0%BB%D0%BA%D0%BE%D0%BA%D1%81%D0%BE%D0%BD%D0%B0_%D0%B4%D0%B2%D1%83%D1%85%D0%B2%D1%8B%D0%B1%D0%BE%D1%80%D0%BE%D1%87%D0%BD%D1%8B%D0%B9 (visited on 05/14/2022).

[32] *Ordered logit*. URL: https://en.wikipedia.org/wiki/Ordered_logit (visited on 06/06/2022).

[33] Kennis Research. *Receiver Operating Characteristic (ROC) Curves*. URL: https://kennis-research.shinyapps.io/ROC-Curves/ (visited on 06/25/2022).

[34] *Spyder IDE site*. URL: https://www.spyder-ide.org/.

[35] PBC Studio. *RStudio official site*. . URL: https://www.rstudio.com/ (visited on 08/19/2021).

[36] CTAN team. *TeX official site*. English. CTAN Team. URL: https://www.ctan.org/ (visited on 11/15/2020).

[37] LaTeX team. *LaTeX official site*. English. URL: https://www.latex-project.org/ (visited on 11/15/2020).

[38] *TeXLive official site*. URL: https://www.tug.org/texlive/ (visited on 11/15/2020).

[39] The R Foundation. *The R Project for Statistical Computing*. . The R Foundation. URL: https://www.r-project.org/ (visited on 08/17/2021).

[40] *Turtle graphics*. URL: https://en.wikipedia.org/wiki/Turtle_graphics (visited on 06/22/2022).

[41] *Welch-t-test*. URL: https://en.wikipedia.org/wiki/Welch's_t-test (visited on 06/06/2022).

[42] Wikipedia. *Accuracy and precision*. URL: https://en.wikipedia.org/wiki/Accuracy_and_precision (visited on 08/09/2022).

[43] Wikipedia. *Commom Language Effect Size*. English. URL: https://en.wikipedia.org/wiki/Effect_size#Common_language_effect_size (visited on 05/16/2022).

[44] Wikipedia. *Diagnostic odds ratio*. URL: https://en.wikipedia.org/wiki/Diagnostic_odds_ratio (visited on 08/10/2022).

[45] Wikipedia. *F-score*. URL: https://en.wikipedia.org/wiki/F-score (visited on 08/09/2022).

[46] Wikipedia. *False discovery rate*. URL: https://en.wikipedia.org/wiki/False_discovery_rate (visited on 08/09/2022).

[47] Wikipedia. *False positive rate*. URL: https://en.wikipedia.org/wiki/False_positive_rate (visited on 08/08/2022).

[48] Wikipedia. *Fowlkes–Mallows index*. URL: https://en.wikipedia.org/wiki/Fowlkes%E2%80%93Mallows_index (visited on 08/10/2022).

[49] Wikipedia. *Hit rate*. URL: https://en.wikipedia.org/wiki/Hit_rate (visited on 08/08/2022).

[50] Wikipedia. *Information retrieval*. URL: https://en.wikipedia.org/wiki/Information_retrieval (visited on 08/09/2022).

[51] Wikipedia. *Jaccard index*. URL: https://en.wikipedia.org/wiki/Jaccard_index (visited on 08/09/2022).

[52] Wikipedia. *KISS principle*. URL: https://en.wikipedia.org/wiki/KISS_principle (visited on 11/06/2020).

[53] Wikipedia. *Likelihood ratios in diagnostic testing*. URL: https://en.wikipedia.org/wiki/Likelihood_ratios_in_diagnostic_testing (visited on 08/09/2022).

[54] Wikipedia. *Markedness*. URL: https://en.wikipedia.org/wiki/Markedness (visited on 08/10/2022).

[55] Wikipedia. *Phi coefficient*. URL: https://en.wikipedia.org/wiki/Phi_coefficient (visited on 08/10/2022).

[56] Wikipedia. *Positive and negative predictive values*. URL: https://en.wikipedia.org/wiki/Positive_and_negative_predictive_values (visited on 08/09/2022).

[57] Wikipedia. *Precision and recall*. URL: https://en.wikipedia.org/wiki/Precision_and_recall (visited on 08/08/2022).

[58] Wikipedia. *Prevalence*. URL: https://en.wikipedia.org/wiki/Prevalence (visited on 08/10/2022).

[59] Wikipedia. *Rank-biserial correlation*. English. URL: https://en.wikipedia.org/wiki/Effect_size#Rank-biserial_correlation (visited on 05/16/2022).

[60] Wikipedia. *Receiver operating characteristic*. URL: https://en.wikipedia.org/wiki/Receiver_operating_characteristic (visited on 05/17/2022).

[61] Wikipedia. *Sensitivity and specificity*. URL: https://en.wikipedia.org/wiki/Sensitivity_and_specificity (visited on 08/08/2022).

[62] Wikipedia. *Sigmoid function*. URL: https://en.wikipedia.org/wiki/Sigmoid_function (visited on 06/24/2022).

[63] Wikipedia. *Simple random sample*. URL: https://en.wikipedia.org/wiki/Simple_random_sample (visited on 07/26/2022).

[64] Wikipedia. *Standard score*. URL: https://en.wikipedia.org/wiki/Standard_score (visited on 05/17/2022).

[65] Wikipedia. *Type I and type II errors*. URL: https://en.wikipedia.org/wiki/Type_I_and_type_II_errors (visited on 06/08/2022).

[66] Wikipedia. *Type I and type II errors*. URL: https://en.wikipedia.org/wiki/Type_I_and_type_II_errors (visited on 08/08/2022).

[67] Wikipedia. *Youden's J statistic*. URL: https://en.wikipedia.org/wiki/Youden's_J_statistic (visited on 08/10/2022).

[68] Benito van der Zander. *TeXstusio official site*. URL: https://www.texstudio.org/ (visited on 11/15/2020).