Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Circuit Theory

Classification of Realisations of Random Sets

Diploma thesis

Bogdan Radović

Field of study: Medical Electronics and Bioinformatics
Supervisor: doc. RNDr. Kateřina Helisová, Ph.D.

Prague, 2024

MASTER‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name:	**Radović Bogdan**	Personal ID number:	**483887**
Faculty / Institute:	**Faculty of Electrical Engineering**		
Department / Institute:	**Department of Circuit Theory**		
Study program:	**Medical Electronics and Bioinformatics**		
Specialisation:	**Medical Instrumentation**		

II. Master’s thesis details

Master’s thesis title in English:

Classification of Realisations of Random Sets

Master’s thesis title in Czech:

Klasifikace realizací náhodných množin

Guidelines:

1. Study classification methods of multidimensional and functional data from existing literature.
2. Suggest a suitable classifier of realisations of random sets and describe its properties.
3. Develop a program for application of the suggested classifier to binary images.
4. Perform a simulation study.
5. Show an application of the procedure to real medical data.

Bibliography / sources:

[1] Molchanov I. (2005): Theory of random sets. Springer, New York.
[2] Ferraty F., Vieu P. (2006) Nonparametric functional data analysis. Theory and practice. Springer-Verlag, New York.
[3] Řezanková H., Húsek D., Snášel V. (2007): Shluková analýza dat. Professional publishing, Praha.
[4] Gotovac Đogaš V., Helisová K., Radović B., Staněk J., Zikmundová M., Brejchová K. (2021): Two-step method for assessing similarity of random sets. Image Analysis and Stereology 40, 127–140.
[5] Pawlasová K., Dvořák J. (2022): Supervised nonparametric classification in the context of replicated point patterns. Image Analysis and Stereology 41, 57–74.

Name and workplace of master’s thesis supervisor:

doc. RNDr. Kateřina Helisová, Ph.D. Department of Mathematics FEE

Name and workplace of second master’s thesis supervisor or consultant:

Date of master’s thesis assignment: **31.08.2023** Deadline for master's thesis submission: **09.01.2024**

Assignment valid until: **16.02.2025**

doc. RNDr. Kateřina Helisová, Ph.D. Supervisor’s signature	doc. Ing. Radoslav Bortel, Ph.D. Head of department’s signature	prof. Mgr. Petr Páta, Ph.D. Dean’s signature

III. Assignment receipt

The student acknowledges that the master’s thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the master’s thesis, the author must state the names of consultants and include a list of references.

.

Date of assignment receipt　　　　Student’s signature

Declaration

I hereby declare that the presented thesis was developed independently and that I have listed all sources of information in accordance with the Methodological Instructions for observing the ethical principles in the preparation of university theses.

In Prague, 2024

....
Bogdan Radović

Acknowledgements

I would like to express my gratitude to the supervisor of this thesis, doc. RNDr. Kateřina Helisová, Ph.D., for such an interesting assignment and for all the help she provided, especially for keeping me on the right course during this thesis. Also, I'd like to thank Ing. Jan Sláma, without whose help the program would take eight times more time to complete. Finally, gratitude has to be expressed to my greatest support and inspiration, my beloved family.

Abstract

Random sets have gained significant importance in recent years as a valuable tool for modelling a wide range of phenomena in fields such as biology, geology, medicine, or material sciences. However, to the best of our knowledge, classification of their realisations has not yet been studied. In the presented work, a link between methods for random sets and functional data analysis is built that focusses on evaluating functional characteristics from individual components in the realisations based on their shape. Such obtained functional data is then used for nonparametric classification using both supervised and unsupervised approach based on k-nearest neighbours and k-means algorithms, respectively. The proposed procedures have been justified through a simulation study. Finally, the procedure is applied to medical data to show its applicability in practice.

Keywords: Convex compact set, Curvature, k-means, k-nearest neighbours, N-distance, Nonparametric functional data analysis, Random set, Stochastic geometry, Supervised classification, Unsupervised classification.

Abstrakt

Náhodné množiny se v posledních letech staly cenným nástrojem pro modelování široké škály jevů v různých oborech jako např. biologie, geologie, medicína či materiálové vědy. Nicméně, pokud je nám dobře známo, klasifikace jejich realizací je zatím neprozkoumané téma. V předkládané práci jsou propojeny metody vyvinuté pro náhodné množiny s metodami analýzy funkcionálních dat. Celá procedura se pak zaměřuje na klasifikaci podle funkcionálních charakteristik jednotlivých komponent v realizaci náhodné množiny na základě jejich tvaru. Použita je přitom neparametrická klasifikace jak tzv. s učitelem, tak bez učitele, jmenovitě algoritmy k-nejbližších sousedů, resp. k-průměrů. Navržené postupy byly nejprve ověřeny simulační studií a nakonec byly postupy aplikovány na lékařská data, aby se ukázala jejich použitelnost v praxi.

Klíčová slova: Konvexní kompaktní množina, křivost, k-průměry, k-nejbližších sousedů, N-vzdálenost, neparametrická funkcionální analýza dat, náhodná množina, stochastická geometrie, klasifikace s učitelem, klasifikace bez učitele.

Contents

List of Figures

List of Tables

Nomenclature

$(a,b], [a,b)$	left-open and right-open semi-open intervals in $\mathbb{R}$
$(a,b), [a,b]$	open and closed interval in $\mathbb{R}$
$\#(\mathbf{X})$	number of elements in $\mathbf{X}$
$\lfloor . \rfloor$	floor function
$\mathbb{C}$	set of complex numbers
$\mathbb{E}$	generic functional (infinite dimensional) space
$\mathbb{F}$	system of all closed subsets of $\mathbb{R}$
$\mathbb{K}$	system of all compact subsets of $\mathbb{R}$
$\mathbb{R}^d$	d-dimensional set of real numbers
$\mathbb{1}_{[\mathbf{A}]}$	indicator function of $\mathbf{A}$
$\mathbb{M}$	system of locally finite subsets of $\mathbb{R}^d$
$\mathbf{C}$	generic convex compact set
$\mathbf{D}(x_o, r)$	ball centered at x_o with radius r
$\mathbf{G}$	set of all possible group (class) labels
$\mathbf{K}$	generic compact subset of $\mathbb{R}$
$\mathbf{S}, \mathsf{S}$	generic random set and its digitised two-dimensional approximation
$\mathbf{X}, \mathbf{A}$	generic sets
$\mathcal{A}$	$\sigma-$algebra of $\mathbf{A}$
$\mathcal{B}$	Borel $\sigma-$algebra
$\mathcal{C}$	system of all convex bodies
$\mathcal{F}$	Effros (hitting) $\sigma-$algebra
$\mathcal{K}$	positive definite kernel
$\mathcal{L}$	strongly negative definite kernel
$\mathcal{X}, \chi$	(functional) random variable, its observation
$\mathcal{M}$	σ-algebra of locally finite subsets of $\mathbb{R}^d$
$\mathscr{B}_{\mathbf{X}}$	boundary of the set $\mathbf{X}$
$\lvert . \rvert$	absolute value
μ, ν	measure
v	Lebesgue measure
$d(.,.)$	metrics (distance)
	$d_E(.,.)$ – Euclidean
	$d_M(.,.)$ – Manhattan
	$d_s(.,.)$ – semi-metrics
P	probability measure
$p_g(.), \hat{p}_g(.)$	posterior probability of a group $g \in \mathbf{G}$ and its estimate
$P_{\mathbf{S}}$	Probability distribution of a random set $\mathbf{S}$
$\mathrm{E}(.)$	Expected value

Chapter 1

Introduction

The randomness and variability of geometrical patterns occurring in nature constantly motivate scientists to develop new methods for processing and analysing the data that reflect these patterns. The problem when working with such data lies in the fact that usually there is only one realisation of each process to consider. For that reason, statisticians and data scientists put emphasis on statistical modelling.

Over the past few decades, random sets have been proven to be a very potent tool for modelling various phenomena in ecology [1], biology [2], material science [3], etc. Most importantly, they have recently been applied in biomedicine for modelling different cell patterns and tissues, see e.g. [4], [5] and [6]. (For a broader list of applications, the reader is referred to books by Illian et al. [7], Baddeley and Jensen [8], and Chiu et al. [9].) Due to the great diversity of their application, modelling and statistical analysis of random sets have been rapidly developing due to the fact that neither traditional methods for comparing random sets nor technologically advanced image processing tools are suitable for the problem of distinguishing between different natural processes.

Classification problems motivate a great number of scientific works since classification is a fundamental process in the study of various phenomena. Its purpose is to categorize new data based upon its relevance to already available, known data. The classification domain can be divided into two main subcategories: supervised classification (or discrimination, where the class structure is known a priori) and unsupervised classification (or clustering, where classes have to be defined) [10]. Class assignment is usually done using a decision rule which is expressed in terms of a set of random variables (in computer science literature usually referred to as *attributes* or *case features*).

In order to access classification of realisations of random sets, we establish a link between methods used for random sets and methods used for functional data analysis, which were extensively studied in [11]. This means that instead of directly comparing the realisations of random sets, we facilitate the problem by comparing the functional data derived from the individual realisations. A similar link between methods for point patterns and functional data using functional summary characteristics (namely, the pair correlation function and the contact distribution function) was developed in [12] for the supervised classification case. The authors based their approach on kernel-regression classifier, which proved to be suitable for accessing the given problem.

In this thesis, my goal will be to suggest a suitable classification procedure that will correctly classify realisations of random sets and to implement the proposed procedure, which will be verified using simulated data. Consequently, the procedure will be applied to images of two types of mammary tissue in order to test its applicability to medical data.

The presented thesis is organised as follows. In Chapter 2, theoretical background is introduced. In Chapter 3, we suggest two classifiers, one for supervised classification and one for unsupervised classification. The procedures are based on functional non-parametric classification, where the functional data are derived from the shape of the components of random sets. In Chapter 4, we validate the procedure using simulated data. In Chapter 5, we apply the procedure to real data, that is, to different types of mammary tissue. In Chapter 6, we summarise the results and suggest possible topics for future research.

Chapter 2

Theoretical Background

In this chapter, some basic definitions from general random set theory and stochastic geometry are introduced. However, the main focus is on the basic concepts of functional data analysis which will help us build the apparatus for achieving the goals of this work.

2.1 Basic terms

All definitions in this section can be found, with slightly different notation, in the book [9], unless stated otherwise.

Definition 2.1.1 (*Metric space, metrics*)**.** A *metric space* is an ordered pair $(\mathbf{X}, d)$, where $\mathbf{X}$ is a set, usually $\mathbf{X} \subseteq \mathbb{R}^d$ and d is a mapping $d : \mathbf{X} \times \mathbf{X} \longrightarrow \mathbb{R}$ which satisfies the following conditions:

- $d(x, y) \geq 0$ (non-negativity),
- $d(x, y) = 0$ iff $x = y$ (separation),
- $d(x, y) = d(y, x)$ (symmetry),
- $d(x, z) \leq d(x, y) + d(y, z)$ (triangle inequality),

for any $x, y, z \in \mathbf{X}$. The function d is called *metrics* on $\mathbf{X}$ or simply *distance* [13].

Example (*Euclidean and Manhattan distance*)*.* Let us consider the Euclidean plane with two points $p = [p_1, p_2]$ and $q = [q_1, q_2]$. The *Euclidean distance* between p and q is then defined by

$$d_E(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}. \tag{2.1}$$

For the same points, their *Manhattan distance* is given by

$$d_M(p, q) =\mid p_1 - q_1 \mid + \mid p_2 - q_2 \mid . \tag{2.2}$$

Definition 2.1.2 (*Ball*)**.** Suppose $(\mathbf{X}, d)$ is a metric space, and let x_o be a point in $\mathbf{X}$. For each $r \in \mathbb{R}^+$ we define

- the *closed ball* in $\mathbf{X}$ centered at x_o with radius r as

$$\mathbf{D}(x_o, r) = \{x \in \mathbf{X} : d(x_o, x) \leq r\}, \tag{2.3}$$

- the *open ball* in $\mathbf{X}$ centered at x_o with radius r as

$$\mathbf{D}^{int}(x_o, r) = \{x \in \mathbf{X} : d(x_o, x) < r\}, \tag{2.4}$$

- the *sphere* as the difference between a closed and a concentric open ball

$$\mathbf{D}^{sph}(x_o, r) = \{x \in \mathbf{X} : d(x_o, x) = r\}. \tag{2.5}$$

Definition 2.1.3 (*Bounded set*)**.** A set $\mathbf{A} \subset \mathbb{R}^d$ is said to be *bounded* if there exists a ball $\mathbf{D}(x_o, r) \subset \mathbb{R}^d$, such that $\mathbf{A} \subset \mathbf{D}(x_o, r)$.

Definition 2.1.4 (*Open and closed sets*)**.** A set $\mathbf{X}$ is said to be *open* if $\forall x \in \mathbf{X}$ there exists a positive number ε such that $\mathbf{D}(x, \varepsilon) \subset \mathbf{X}$. A set $\mathbf{X}$ is said to be *closed* if its complement $\mathbf{X}^{\mathbf{c}}$ in $\mathbb{R}^d$ is open. The system of all closed subsets of $\mathbb{R}^d$ will be denoted as $\mathbb{F}$.

Definition 2.1.5 (*Interior, closure, and boundary*)**.** The *interior* $\mathbf{X}^{\mathbf{int}}$ of the set $\mathbf{X}$ is the union of all open sets contained in $\mathbf{X}$. The *closure* $\mathbf{X}^{\mathbf{cl}}$ of the set $\mathbf{X}$ is the intersection of all closed sets containing $\mathbf{X}$. The difference $\partial\mathbf{X} = \mathcal{B}_{\mathbf{X}} = \mathbf{X}^{\mathbf{cl}} - \mathbf{X}^{\mathbf{int}}$ is called the *boundary* of $\mathbf{X}$.

Definition 2.1.6 (*Compact set*)**.** A set $\mathbf{K} \subset \mathbb{R}^d$ is said to be *compact* if it is both closed and bounded. The system of all compact subsets of $\mathbb{R}^d$ shall be denoted as $\mathbb{K}$.

Definition 2.1.7 (*Topology*)**.** Let $(\mathbf{T}, \mathcal{T})$ be an ordered pair, where $\mathbf{T}$ is a set and $\mathcal{T}$ is a collection of open subsets of $\mathbf{T}$ satisfying:

- $\emptyset \in \mathcal{T}$and $\mathbf{T} \in \mathcal{T}$,
- $\bigcup_i \mathbf{T}_i \in \mathcal{T}$, for any sets $\mathbf{T}_i \in \mathcal{T}, i \in \mathbb{N}$,
- $\bigcap_i \mathbf{T}_i \in \mathcal{T}$, for any sets $\mathbf{T}_i \in \mathcal{T}$where i is finite.

Then the couple $(\mathbf{T}, \mathcal{T})$ is called *topological space* and the collection $\mathcal{T}$ is called the *topology* on $(\mathbf{T}, \mathcal{T})$.

Definition 2.1.8 (*Connected set*)**.** Let $(\mathbf{T}, \mathcal{T})$ be a topological space. A subset $\mathbf{X} \subset \mathbf{T}$ is called a *connected set* if it cannot be separated into two nonempty subsets such that each subset has no common points with the set closure of the other [14].

Definition 2.1.9 (*Convex set*)**.** A set $\mathbf{X} \subset \mathbb{R}^d$ is said to be *convex* if for every $x, y \in \mathbf{X}$ and every $0 < c < 1$ we have $cx + (1-c)y \in \mathbf{X}$. Convex sets, which are also compact, are called *convex bodies.*

Definition 2.1.10 (*Convex body functional*)**.** A *convex body functional* assigns a real value $h(\mathbf{C})$ for every $\mathbf{C} \in \mathcal{C}$, where $\mathcal{C}$ denotes the system of all convex bodies.

Example (*Convex body functionals*). Some of the most important convex body functionals of a set $\mathbf{K} \in \mathcal{C}$ in different dimensions are length of a curve, boundary length or area of a planar set, surface area or volume of a 3D body, etc.

Definition 2.1.11 (*$\sigma-$algebra, Borel and Effros $\sigma-$algebras*)**.** For each set $\mathbf{A}$, a system $\mathcal{A}$ of its subsets is called $\sigma-$algebra if it satisfies the following:

- $\mathbf{A} \in \mathcal{A}$,
- if $\mathbf{X} \in \mathcal{A}$, then $\mathbf{X^c} \in \mathcal{A}$,
- if $\mathbf{X_1}, \mathbf{X}_2, \ldots \in \mathcal{A}$, then $\bigcup_{i=1}^{\infty} \mathbf{X_i} \in \mathcal{A}$.

The smallest $\sigma-$algebra on $\mathbb{R}^d$ containing all open subsets of $\mathbb{R}^d$ is called *Borel $\sigma-$algebra* and is denoted by $\mathcal{B}$. Elements of a Borel $\sigma-$algebra are called Borel sets.

The smallest $\sigma-$algebra of subsets of $\mathbb{F}$ containing all 'hitting' sets

$$\mathbb{F}_{\mathbf{K}} = \{\mathbf{F} \in \mathbb{F} : \mathbf{F} \cap \mathbf{K} \neq \emptyset\}, \forall \mathbf{K} \in \mathbb{K} \tag{2.6}$$

is called *Effros $\sigma-$algebra* and is denoted by $\mathcal{F}$.

Definition 2.1.12 (*Measurable space, measurable set and measurable function*)**.** The pair $(\mathbf{A}, \mathcal{A})$ formed by a set $\mathbf{A}$ and $\sigma-$algebra $\mathcal{A}$ of the subsets of $\mathbf{A}$ is called a *measurable space* and the $\mathbf{X}$ in $\mathcal{A}$ are called *measurable sets.* A function $f : \mathbf{X} \longrightarrow \mathbb{R}$ is said to be $\mathcal{A}-$ measurable if for each Borel set $\mathbf{B} \in \mathcal{B}$ the inverse image $f^{-1}(\mathbf{B})$ belongs to $\sigma-$algebra $\mathcal{A}$ associated with $\mathbf{A}$.

Definition 2.1.13 (*Measure, measure space*)**.** Suppose that $(\mathbf{A}, \mathcal{A})$ is a measurable space. A function $\mu : \mathcal{A} \longrightarrow [0, \infty)$ satisfying:

- $\mu(\emptyset) = 0$,
- $\mu(\bigcup_{k=1}^{\infty} \mathbf{X}_k) = \sum_{k=1}^{\infty} \mu(\mathbf{X}_k)$

for all $\mathbf{X}_k \in \mathcal{A}$ with $\mathbf{X}_i \cap \mathbf{X}_j \neq \emptyset$ whenever $i \neq j$ is called a *measure* on $(\mathbf{A}, \mathcal{A})$. The triplet $(\mathbf{A}, \mathcal{A}, \mu)$ is called a *measure space.*

Definition 2.1.14 (*Finite, σ-finite, locally-finite measure*)**.** Let $(\mathbf{A}, \mathcal{A}, \mu)$ be a measure space. Both $(\mathbf{A}, \mathcal{A}, \mu)$ and μ are called *totally finite* if $\mu(\mathbf{A})$, and *σ-finite* if $\mathbf{A}$ can be split into countably many sets of finite measure, that is, $\mathbf{A} = \cup_{i=1}^{n} \mathbf{A}_i$ for some $n \in \mathbb{N}$ such that $\forall i, \mu(\mathbf{A}_i) < \infty$. The totally finite measures are also σ-finite. A measure μ is called *locally finite* if it is finite on bounded sets. [15]

Definition 2.1.15 (*Borel measure*)**.** A Borel measure is any measure defined on the $\sigma-$ algebra of Borel sets.

Definition 2.1.16 (*Lebesgue measure*)**.** For $\mathbf{Q} = [u_1, w_1] \times ... \times [u_d, w_d] \subset \mathbb{R}^d$ Lebesgue measure is defined by

$$v_d(\mathbf{Q}) = (u_1 - w_1) \cdot ... \cdot (u_d - w_d), \tag{2.7}$$

i.e. it is characterised by the volume of a d-dimensional hypercube.

Definition 2.1.17 (*Probability space, random variable*)**.** A measure space (Ω, Σ, P) is called *probability space* if it holds that $P(\Omega) = 1$. In that case, the measure P is called a *probability measure*, the Ω is called the *sample space* and its elements are called *sample points*, the subsets of Ω that belong to Σ are called *events.* Real-valued $\Sigma-$measurable functions defined on Ω are called *random variables.*

2.2 Random Set Theory

Definitions in this section come from [9], unless otherwise stated.

Definition 2.2.1 (*Random closed set*)**.** Let (Ω, Σ, P) be a probability space. A measurable mapping $\mathbf{S} : (\Omega, \Sigma, P) \longrightarrow (\mathbb{F}, \mathcal{F})$ is a *random closed set* if for every compact $\mathbf{K} \in \mathbb{K}$ we have $\{\omega \in \Omega : \mathbf{S} \cap \mathbf{K} \neq \emptyset\} \in \Sigma$.

If we replace $\mathbb{K}$ by the system of convex bodies $\mathcal{C}$ in Definition 2.2.1, we get the definition of a *random convex compact set.*

Definition 2.2.2 (*Probability distribution of a random set*)**.** The *probability distribution* $P_{\mathbf{S}}$ of a random set $\mathbf{S}$ is defined by

$$P_{\mathbf{S}}(\mathbf{F}) = P(\mathbf{S}^{-1}(\mathbf{F})) = P(\mathbf{S} \in \mathbf{F}), \tag{2.8}$$

for every $\mathbf{F} \in \mathcal{F}$.

Definition 2.2.3 (*Independent random sets*)**.** Two random sets $\mathbf{S}_1$ and $\mathbf{S}_2$ are independent if and only if for any $\mathbf{F}_1$ and $\mathbf{F}_2$ in $\mathcal{F}$ we have

$$P(\mathbf{S}_1^{-1}(\mathbf{F}_1) \cap \mathbf{S}_2^{-1}(\mathbf{F}_2)) = P(\mathbf{S}_1^{-1}(\mathbf{F}_1)) \cdot P(\mathbf{S}_2^{-1}(\mathbf{F}_2)). \tag{2.9}$$

We can find this definition in [16].

Definition 2.2.4 (*Stationarity, isotropy*)**.** A random closed set $\mathbf{S}$ is *stationary* if its distribution $P_{\mathbf{S}}(\mathbf{F}) = P(\omega \in \Omega : \mathbf{S}(\omega) \in \mathbf{F})$ for $\mathbf{F} \in \mathcal{F}$ is invariant under translation. A random closed set $\mathbf{S}$ is *isotropic* if its distribution is invariant under rotation. If a random closed set is both stationary and isotropic, it is called *motion invariant.*

Definition 2.2.5 (*Neighbourhood*)**.** Consider a finite union of disjoint random sets $\{\mathbf{S}_1, ..., \mathbf{S}_n\}$ within an observation window $\mathbf{W} \subset \mathbb{R}^d$. Every set $\mathbf{S}_i$ generates a *neighbourhood*

$$\mathbf{H}_M^i = \{z \in \mathbf{W} : d_M(\{z\}, \mathbf{S}_i) \leq d_M(\{z\}, \mathbf{S}_j) \text{ for all } i \neq j\}. \tag{2.10}$$

2.3 Functional Nonparametric Statistics

The following definitions are mainly taken from [11], unless stated otherwise.

Definition 2.3.1 (*Functional random variable*)**.** A random variable $\mathcal{X}$ is called a functional random variable (f.r.v.) if it takes values in an infinite-dimensional space $\mathbb{E}$ (or functional space). An observation χ of $\mathcal{X}$ is called functional data.

Example. Some of the examples of f.r.v. are a random curve, a random surface, etc.

Definition 2.3.2 (*Functional dataset*)**.** A functional dataset $\chi_1, ..., \chi_n$ is the observation of n functional variables $\mathcal{X}_1, ..., \mathcal{X}_n$ identically distributed as $\mathcal{X}$.

Definition 2.3.3 (*Mean*)**.** Let $\mathcal{S} = \{\mathcal{X}_1, \ldots, \mathcal{X}_n\}$ be a sample on n functional random variables identically and independently distributed as $\mathcal{X}$ taking values in $\mathbb{E}$, $\chi_1, \ldots, \chi_n$ the functional data set associated with $\mathcal{S}$ and let (Ω, Σ, P) be a probability space. The *mean* of a (functional) random variable $\mathcal{X}$ is defined as

$$E(\mathcal{X}) = \int_\Omega \mathcal{X}(\omega) dP(w) \tag{2.11}$$

and its estimator (known as empirical mean) by

$$\overline{\mathcal{X}_\mathcal{S}} = \frac{1}{n} \sum_{i=1}^{n} \mathcal{X}_i \tag{2.12}$$

Definition 2.3.4 (*Functional parametric and nonparametric model*)**.** Let $\mathcal{X}$ be a random variable valued in some infinite dimensional space $\mathbb{E}$ and let f be a mapping defined on $\mathbb{E}$ and dependent on the distribution of $\mathcal{X}$. The model is called a functional parametric model for the estimation of f if it is evaluated in a finite number of elements of $\mathbb{E}$, or a functional nonparametric model otherwise.

Since the dimension of the space in which a random variable takes values dictates the sparseness of the data, it is easily deduced that in the case of *functional* random variable, it will be harder to measure the distance between two realisations. In finite dimensional space $\mathbb{R}^d$, we can use norms defined by

$$|| \, x \, ||^2_{\mathbf{M}} = \sum_{i=1}^{d} (x_i)^2 = x^T \mathbf{M} x, \tag{2.13}$$

where $\mathbf{M}$ is a definite positive matrix. From 2.13 we see that all the norms in $\mathbb{R}^d$ are equivalent. However, in high-dimensional spaces the notion of equivalence between norms does not hold. Thus, the choice of the norm becomes crucial. It has been shown [17] that usage of normed or metric spaces in such settings is restrictive. For this reason, some of the restrictions in Definition 2.1.1 have to be relaxed.

Definition 2.3.5 (*Semi-metrics*)**.** Let $\mathbb{E}$ be an infinite dimensional space, and let d_s be a mapping $d_s : \mathbb{E} \times \mathbb{E} \longrightarrow \mathbb{R}$ which for any $x, y, z \in \mathbb{E}$ satisfies:

- $d_s(x, y) \geq 0$ (non-negativity),
- $d_s(x, y) = d_s(y, x)$ (symmetry),
- $d_s(x, z) \leq d_s(x, y) + d_s(y, z)$ (triangle inequality).

Then d_s is called semi-metrics and space $(\mathbb{E}, d_s)$ semi-metric[1].

[1] In literature, names pre-metrics, pseudo-metrics and quasi-metrics are also used, see [18], [19].

2.3.1 Supervised classification

In the setting for supervised classification, we observe a functional random variable $\mathcal{X}$ and a categorical random variable Y which, for each realisation χ of $\mathcal{X}$, indicates the class (synonymously, group) membership y (also called label). The classification task could be summarised as: given a new functional data χ predict its label y. In the sense of functional data, it is important to stress that the classical linear discriminant analysis fails, since the highly correlated predictors degenerate the within-class covariance matrix.

Let $(\mathcal{X}_i, Y_i), i = 1, ..., n$ be a sample of n independent identically distributed pairs from $(\mathcal{X}, Y)$ taking values in $\mathbb{E}\times\mathbf{G}$, where $(\mathbb{E}, d_s)$ is a semi-metric vector space and $\mathbf{G} = \{1, 2, ..., G\}$ a set of all possible groups. The notation (χ_i, y_i) will be used in the rest of the text for an observation of the pair $(\mathcal{X}_i, Y_i)$ for $i = 1, ..., n$.

Definition 2.3.6 (*Bayes rule*)**.** Given a functional object χ in $\mathbb{E}$ estimate G posterior probabilities

$$p_g(\chi) = P(Y = g \mid \mathcal{X} = \chi), g \in \mathbf{G} \tag{2.14}$$

for each group $g \in \mathbf{G}$. The Bayes classification rule consists of assigning the observation χ to the class with the highest estimated posterior probability:

$$\hat{y}(\chi) = \arg\max_{g\in\mathbf{G}} p_g(\chi). \tag{2.15}$$

It is important to note that Definition 2.14 can be rewritten as:

$$p_g(\chi) = E(\mathbb{1}_{[Y=g]} \mid \mathcal{X} = \chi), \tag{2.16}$$

which means that we can express posterior probabilities using conditional expectations[2].

Definition 2.3.7 (*Kernel estimator of posterior probabilities*)**.** Let $\mathcal{K}$ be an asymmetric kernel with bandwidth (a smoothing parameter) $h, h > 0$. Then the kernel estimator of posterior probability $p_g(\chi)$ is defined by

$$\hat{p}_g(\chi) = \hat{p}_{g,h}(\chi) = \frac{\sum_{i=1}^{n} 1_{[Y=g]}\mathcal{K}(h^{-1}d_s(\chi, \mathcal{X}_i))}{\sum_{i=1}^{n} \mathcal{K}(h^{-1}d_s(\chi, \mathcal{X}_i))}. \tag{2.17}$$

Setting

$$w_{i,h}(\chi) = \frac{\mathcal{K}(h^{-1}d_s(\chi, \mathcal{X}_i))}{\sum_{i=1}^{n} \mathcal{K}(h^{-1}d_s(\chi, \mathcal{X}_i))}, \tag{2.18}$$

[2] For the definition of tho conditional expectation, see [20].

we get

$$\hat{p}_{g,h}(\chi) = \sum_{i \in \mathbf{I}} w_{i,h}(\chi), \text{where } \mathbf{I} = \{i : Y_i = g\} \cap \{i : d_s(\chi, \mathcal{X}_i) < h\} \tag{2.19}$$

It is important to note that the estimated posterior probabilities $\hat{p}_{g,h}$ form a discrete distribution if $\mathcal{K}$ is nonnegative.

From the above definitions, we can see that choosing a proper semi-metric d_s and the bandwidth h play a vital role for the proper functioning of the kernel estimator. Choosing h is usually simplified to the minimisation of a loss function: $h_{Loss} = \arg\inf_h Loss(h)$, where $Loss$ is usually given by the misclassification rate. Computationally, it would be more efficient to replace the real-valued continuous h that takes values from an infinite set with an integer parameter k that takes values from a finite subset. One way to do that is by using the k-nearest neighbours estimator.

Definition 2.3.8 (*k-nearest neighbours estimator*)**.** Let $x_i = \{\chi_i(p_1), ..., \chi_i(p_J)\}$ be a discretised version of a curve $\chi_i = \{\chi_i(p); p \in \mathbf{P}\}$ measured on a grid of J points $p_1, ..., p_J$, and let y_i be respective class labels. If n identically and independently distributed pairs $(x_i, y_i)_{i=1,...,n}$ are observed, we can rewrite Equation 2.17 as

$$\hat{p}_{g,k}(x) = \frac{\sum_{i:y_i=g}^{n} \mathcal{K}(h_k^{-1} d_s(x, x_i))}{\sum_{i=1}^{n} \mathcal{K}(h_k^{-1} d_s(x, x_i))}, \tag{2.20}$$

where h_k satisfies

$$card\{i : d_s(x, x_i) < h_k\} = k. \tag{2.21}$$

2.3.2 Unsupervised classification

Unsupervised classification is a domain with a wide range of applications. It differs from supervised classification due to the fact that in the setting for unsupervised classification, we do not observe categorical responses, which means that we have to define homogeneity of a class. Since we suppose that the generating distribution is unknown, the approach is based on centrality notions, for example the mean, as defined in Definition 2.3.3.

Definition 2.3.9 (*k-means algorithm*)**.** Let $\mathcal{S} = \{\mathcal{X}_1, \ldots, \mathcal{X}_n\}$ be a sample of n functional random variables identically and independently distributed as $\mathcal{X}$ taking values in $\mathbb{E}$, and let $k, k \in \mathbb{N}$ be a positive integer specifying the number of clusters, $k \leq n$. The goal of the k-means algorithm is to partition the set $\mathcal{S}$ into k subsets $\mathcal{S}^k = \{\mathcal{S}_1, \ldots, \mathcal{S}_k\}, \mathcal{S}_i \subset \mathcal{S}$,

$i = 1, \dots, k$ minimising the within-cluster sum of squares

$$\arg\min_{\mathcal{S}^k} \sum_{i=1}^{k} \sum_{\mathcal{X} \in \mathcal{S}_i} d_s(\mathcal{X} - \overline{\mathcal{X}_{\mathcal{S}_i}})^2, \tag{2.22}$$

where $\overline{\mathcal{X}_{\mathcal{S}_\rangle}}$ is the empirical mean (also called centroid) defined by (2.12). The algorithm can be summarised in 6 steps (out of which the first three are used for initialisation):

1. Choose an initial center c_1 uniformly at random from $\mathcal{S}$
2. Choose the next center c_i selecting $c_i = \mathcal{X}' \in \mathcal{S}$ with probability $\frac{d_s(\mathcal{X}')^2}{\sum_{\mathcal{X} \in \mathcal{S}} d_s(\mathcal{X})^2}$
3. Repeat step 2 until k centers $\{c_1, \dots, c_k\}$ have been chosen
4. For each $i \in \{1, \dots, k\}$, set the cluster $\mathcal{S}_i$ to be the set of points in $\mathcal{S}$ closer to c_i than they are to c_j for all $j \neq i$
5. For each $i \in \{1, \dots, k\}$, set the center c_i to be the centroid (i.e. empirical mean) of points in $\mathcal{S}_i$
6. Repeat steps 4 and 5 until $\{c_1, \dots, c_k\}$ no longer changes. [21]

2.4 $\mathcal{N}$-distance

The following definitions, theorems and examples are taken from [22], unless stated otherwise.

Definition 2.4.1 (*Positive definite kernel*)**.** Let $\mathbf{X}$ be a nonempty set. A map

$$\mathcal{K} : \mathbf{X} \times \mathbf{X} \to \mathbb{C} \tag{2.23}$$

is called *positive definite kernel* if for any $n \in \mathbb{N}$, arbitrary complex numbers $c_1, ..., c_n \in \mathbb{C}$ such that $\sum_{i=1}^{n} c_i = 0$ and arbitrary $x_1, ..., x_n \in \mathbf{X}$ it holds

$$\sum_{i}^{n} \sum_{j}^{n} \mathcal{K}(x_i, x_j) c_i \bar{c_j} \geq 0. \tag{2.24}$$

Example. Let $\mathbf{X} = \mathbb{R}$. An example of a positive definite kernel is

$$\mathcal{K}(s,t) = \begin{cases} 1- \mid s - t \mid, & \mid s - t \mid \leq 1 \\ 0, & \mid s - t \mid > 1. \end{cases} \tag{2.25}$$

Since it is a function of the difference $\mid s - t \mid$ only, we can consider it as a function of one real variable

$$\tilde{\mathcal{K}}(u) = \begin{cases} 1- \mid u \mid, & \mid u \mid \leq 1 \\ 0, & \mid u \mid > 1. \end{cases} \tag{2.26}$$

It is called the triangular kernel and it will be used later in Section 3.1

Definition 2.4.2 (*Negative definite kernel*)**.** Let $\mathbf{X}$ be a nonempty set. A map

$$\mathcal{L} : \mathbf{X} \times \mathbf{X} \to \mathbb{C} \tag{2.27}$$

is called *negative definite kernel* if for any $n \in \mathbb{N}$, arbitrary complex numbers $c_1, ..., c_n \in \mathbb{C}$ such that $\sum_{i=1}^{n} c_i = 0$ and arbitrary $x_1, ..., x_n \in \mathbf{X}$ it holds

$$\sum_{i}^{n} \sum_{j}^{n} \mathcal{L}(x_i, x_j) c_i \bar{c_j} \leq 0. \tag{2.28}$$

Definition 2.4.3 (*Strongly negative definite kernel*)**.** Let $\mathbf{X}$ be a nonempty set and suppose that the map $\mathcal{L}$ is a real continuous function. The *negative definite kernel*

$$\mathcal{L} : \mathbf{X} \times \mathbf{X} \to \mathbb{R} \tag{2.29}$$

is called *strongly negative definite kernel* if for an arbitrary probability measure μ and an arbitrary real function $f : \mathbf{X} \to \mathbb{R}$ such that $\int_{\mathbf{X}} f(x) d\mu(x) = 0$ holds and

$$\int_{\mathbf{X}} \int_{\mathbf{X}} \mathcal{L}(x, y) f(x) f(y) d\mu(x) d\mu(y) \tag{2.30}$$

exists and is finite, the relation

$$\int_{\mathbf{X}} \int_{\mathbf{X}} \mathcal{L}(x, y) f(x) f(y) d\mu(x) d\mu(y) = 0 \tag{2.31}$$

implies that $f(x) = 0$ μ-almost everywhere.

Example. Let $\mathbf{X} = \mathbb{R}^d$. Then an example of a strongly negative definite kernel is the Euclidean distance, i.e.

$$\mathcal{L}(x, y) = d_E(x, y). \tag{2.32}$$

Example. In [6], strongly negative definite kernels for real functions t_1 and t_2 which are evaluated on a finite grid $u_1, \ldots, u_n \in \mathbb{R}$ has been constructed. It is given by the relation

$$\mathcal{L}(t_1, t_2) = \sum_{m=1}^{n} \sum_{\{k_1,\ldots,k_m\}\subseteq\{1,\ldots,n\}} \left(\sum_{l=1}^{m} (t_1(u_{k_l}) - t_2(u_{k_l}))^2 \right)^{1/2} . \tag{2.33}$$

It was also shown that for statistical purposes, we can take $\sum\limits_{m=1}^{d}$ instead of $\sum\limits_{m=1}^{n}$ for an appropriately chosen $d < n$, which makes the calculations less time-consuming ($d = 3$ is recommended, see [6]).

Theorem 2.4.1 (*Klebanov*)**.** Let $\mathcal{L} : \mathbf{X} \times \mathbf{X} \to \mathbb{R}$ be a map satisfying $\mathcal{L}(x, y) = \mathcal{L}(y, x)$. Denote $M_{\mathcal{L}}$ the set of all measures μ such that

$$\int_{\mathbf{S}} \int_{\mathbf{S}} \mathcal{L}(x, y) d\mu(x) d\mu(y) \tag{2.34}$$

exists. Then $\mathcal{N}$-distance of the measures μ and ν is given by equation

$$\begin{aligned} \mathcal{N}(\mu, \nu) =& 2 \int_{\mathcal{X}} \int_{\mathcal{X}} \mathcal{L}(x, y) d\mu(x) d\nu(y) - \int_{\mathcal{X}} \int_{\mathcal{X}} \mathcal{L}(x, y) d\mu(x) d\mu(y) \\ & - \int_{\mathcal{X}} \int_{\mathcal{X}} \mathcal{L}(x, y) d\nu(x) d\nu(y) \geq 0 \end{aligned} \tag{2.35}$$

which holds for all measures $\mu, \nu \in \mathcal{M}_{\mathcal{L}}$ with equality in the case $\mu = \nu$ if and only if $\mathcal{L}$ is a strongly negative definite kernel.

Empirical estimate of $\mathcal{N}$-distance

Assume we have an observation $X_1, ..., X_{m_1}$ from a distribution μ and $Y_1, ..., Y_{m_2}$ from a distribution ν. The $\mathcal{N}$-distance of the measures μ and ν is then estimated as

$$\hat{\mathcal{N}}(\mu, \nu) = \frac{2}{m_1 m_2} \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} \mathcal{L}(X_i, Y_j) - \frac{1}{m_1^2} \sum_{i=1}^{m_1} \sum_{j=1}^{m_1} \mathcal{L}(X_i, X_j) - \frac{1}{m_2^2} \sum_{i=1}^{m_2} \sum_{j=1}^{m_2} \mathcal{L}(Y_i, Y_j). \tag{2.36}$$

2.5 Point Processes

Definitions in this section come from [9], unless stated otherwise.

Multidimensional point processes have been used to model various processes occurring in nature and our surroundings (for example, particles, cells, plants, animals (such as sea urchins) or cellphones), more precisely, their respective geographical locations or centres of mass. They are the elementary structures in stochastic geometry, closely related to random sets.

Definition 2.5.1 (*Point process*)**.** Let (Ω, Σ, P) be a probability space. Consider $\mathbb{M}$, the system of locally finite subsets of $\mathbb{R}^d$, with the σ-algebra $\mathcal{M} = \sigma(\{\mathbf{M} \in \mathbb{M} : \#(\mathbf{M} \cap \mathbf{B}) = m\} : \mathbf{B} \in \mathcal{B}, m \in \mathbb{N}_0)$, where $\mathcal{B}$ denotes the system of bounded Borel sets and $\#(\mathbf{M})$ represents the number of points in the configuration $\mathbf{M}$. A point process Φ defined on $\mathbb{R}^d$ is a measurable mapping from (Ω, Σ) to $(\mathbb{M}, \mathcal{M})$.

Definition 2.5.2 (*Intensity and homogeneity of a point process*)**.** A measure Λ on $\mathcal{B}$ satisfying $\Lambda(\mathbf{B}) = \ \Phi(\mathbf{B})$ for all $\mathbf{B} \in \mathcal{B}$, where $\Phi(\mathbf{B})$ denotes the number of points of Φ in $\mathbf{B}$, is called the *intensity measure.* If there exists a function $\lambda(x)$ for $x \in \mathbb{R}^d$ such that $\Lambda(\mathbf{B}) = \int_{\mathbf{B}} \lambda(x)dx$, then $\lambda(x)$ is called the *intensity function.* If the intensity function $\lambda(x)$ is constant, $\lambda(x) = \lambda$, the point process is called *homogeneous* (or, synonymously, *stationary*) with the *intensity* λ. Otherwise, it is said to be *inhomogeneous* (or, synonymously, *nonstationary*).

Definition 2.5.3 (*Poisson point process*)**.** Let Λ be a locally-finite non-null measure on $\mathbb{R}^d$. The *Poisson point process* Φ of intensity measure Λ is defined using its finite-dimensional distributions:

$$P(\Phi(\mathbf{A}_1) = m_1, ..., \Phi(\mathbf{A}_k) = m_k) = \prod_{i=1}^{k} e^{-\Lambda(\mathbf{A}_i)} \cdot \frac{\Lambda(\mathbf{A}_i)^{m_i}}{m_i!}, \tag{2.37}$$

for every $k = 1, 2, ...$ and all bounded, disjoint sets $\mathbf{A}_i$, $i = 1, 2, ..., k$, such that $\mathbf{A}_i \subset \mathbb{R}^d$. If $\Lambda(\mathbf{A}_i) = \lambda \cdot v_d(\mathbf{A}_i)$, where λ is a constant, then Φ is called a *homogeneous Poisson point process* [23].

Since we work mainly with homogeneous Poisson point process Φ, we can say, in order to summarise, that it is characterised by:

- Poisson distribution of the number of points in each $\mathbf{A} \in \mathcal{B}$ with the parameter $\Lambda(\mathbf{A})$,
- independent scattering, i.e. the numbers of points in disjoint sets are independent random variables.

Definition 2.5.4 (*Boolean model*)**.** Let $\mathbf{\Phi} = \{x_1, x_2, ...\}$ be a stationary Poisson point process in $\mathbb{R}^d$ and $\{\mathbf{K}_1, \mathbf{K}_2, ...\}$ be a sequence of independent identically distributed (i.i.d.) random compact sets in $\mathbb{R}^d$ that are mutually independent and independent of $\mathbf{\Phi}$.
If $E(v_d(\mathbf{K}_1 \oplus \mathbf{K})) < \infty$ for all compact sets $\mathbf{K}$, where $\oplus$ denotes Minkowski-addition, then the random set

$$\mathbf{\Psi} = \bigcup_{i=1}^{\infty} (x_i + \mathbf{K}_i) \tag{2.38}$$

is called the *Boolean model.*

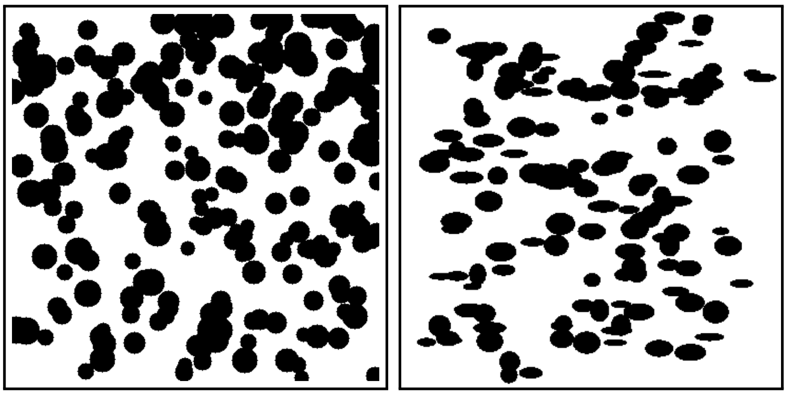


Figure 2.1: Boolean model: random discs (left) and random ellipses (right)

Boolean model is sometimes called Poisson germ-grain model [9]. It can be easily modelled using the Poisson point process with the intensity λ, where around each point of the Poisson process we construct a random geometrical object (e.g. a line segment, a disc, a polygon, a ball etc.). The resulting union is an example of a Boolean model.

The name germ-grain model comes from the point of view that the points of the Poisson process form the *germs*, while the geometrical objects are their corresponding *grains*. The Boolean model is an extremely powerful tool for modelling various natural and artificial phenomena, see [9]. However, it is not a sufficient model for all situations. Here we consider the following, more sophisticated model, the Quermass-interaction model, first defined in [24], which can be used for modelling repulsive and clustering interactions, see Figure 4.1.

Definition 2.5.5 (*Random disc Quermass-interaction process*)**.** Consider a planar random disc Boolean model. The *random disc Quermass-interaction process* is a random set whose probability measure is absolutely continuous with respect to the probability measure of the given Boolean model and the density of its probability measure is given by

$$f_\theta(\mathbf{D}) = \frac{1}{c_\theta} exp\{\theta_1 A(U_\mathbf{D}) + \theta_2 L(U_\mathbf{D}) + \theta_3 \chi(U_\mathbf{D})\}, \tag{2.39}$$

for each finite disc configuration $\mathbf{D} = \{\mathbf{D}_1, \mathbf{D}_2, ..., \mathbf{D}_n\}$, where A, L and χ are, respectively, the area, the perimeter and the Euler–Poincaré characteristic (the number of holes subtracted from the number of connected components) of the union of discs $U_\mathbf{D} = \bigcup_{i=1}^{n} \mathbf{D}_i$, $\theta = (\theta_1, \theta_2, \theta_3)$ is a three-dimensional vector of parameters, and c_θ is the normalising constant [25].

2.6 Curvature of a Planar Curve

The following two definitions come from [26].

Definition 2.6.1 (*Curvature of a curve*)**.** Let $\mathcal{C}$ be a smooth twice differentiable 2D curve that is properly parameterised by a parameter $s \in [0, s_{max}] \subset \mathbb{R}$, that is, $\mathcal{C}(s) = (x(s), y(s))$. The *curvature* of the curve $\mathcal{C}$ at the point $\mathcal{C}(s)$ is then defined by

$$\kappa(\mathcal{C}(s)) = \frac{x'(s)y''(s) - x''(s)y'(s)}{(x'^2(s) + y'^2(s))^{3/2}}, \tag{2.40}$$

where $'$ denotes derivative with respect to s.

In other words, if $r(s)$ is the radius of the osculating circle that touches the curve at the point $[x(s), y(s)]$, then the curvature is given by $\kappa(s) = \pm 1/r(s)$, where the choice between "$+$" and "$-$" is dictated by the local convexity convention.

Let $\mathcal{C}$ be a continuous, closed (i.e. $\mathcal{C}(0) = \mathcal{C}(s_{max})$), and non-self-intersecting curve (i.e. if $\mathcal{C}(s_1) = \mathcal{C}(s_2)$ then $s_1 = s_2$). Suppose that $\mathbf{S}$ is a planar (connected) set whose boundary is determined by $\mathcal{C}$ (with appropriately chosen orientation to ensure the right sign +/-). Curvature $\kappa(z)$, at the point $z \in \mathcal{C}$ and for r small enough, is then given by

$$\kappa(z) \approx \frac{3A^*(\mathbf{D}(z,r))}{r^3} - \frac{3\pi}{2r} = \frac{3\pi}{r}\left(\frac{A^*(\mathbf{D}(z,r))}{A(\mathbf{D}(z,r))} - \frac{1}{2}\right), \tag{2.41}$$

where $A(\mathbf{D}(z,r))$ is the area of the disc $\mathbf{D}(z,r)$ centred at z and $A^*(\mathbf{D}(z,r))$ is the area of $\mathbf{D}(z,r) \cap \mathbf{S}$ [26].

2.6.1 Implementation

Note that this procedure was already implemented in [27].

The starting point for our algorithm is a binary (i.e., consisting solely of black and white pixels) image $\mathbf{W}$ containing a digital approximation S of a planar random set $\mathbf{S}$, such that there are n black disjoint connected components $\mathsf{S}_k, k = 1, 2, \ldots, n$ inside S. Note that our attention will be directed to individual components. In the initial stage, certain terms need to be redefined in order to make them suitable for working with binary images. Note that in the reminder of the text, the terms *point* and *pixel* will be used interchangeably, where *pixel* z will be understood as a square of the unit area that is centred at point z.

Definition 2.6.2 (*4-neighbourhood*)**.** Let z be a point in a binary image $\mathbf{W}$. *4-neighbourhood* of the pixel z is then defined as

$$\mathbf{H}_4(z) = \{\bigcup_i z_i \in \mathbf{W} : d_M(z, z_i) \leq 1\}. \tag{2.42}$$

Since we are working with the digital approximation S of the set $\mathbf{S}$, we have to discretise the function (2.41) in such a way that the area $A(\mathbf{D}(z, r))$ represents the number of pixels inside the disc $\mathbf{D}(z, r)$ centred at the boundary of S, and $A^*(\mathbf{D}(z, r))$ is the number of pixels of $\mathbf{D}(z, r)$ inside S.

Definition 2.6.3 (*Boundary pixel, boundary*)**.** Let $\mathsf{S}_k \subset \mathsf{S}$ be a digital approximation of a connected random set $\mathbf{S}_k \subset \mathbf{S}$, consisting of black pixels. A pixel $z \in \mathsf{S}_k$ is called a *boundary pixel* if and only if at least one of its neighbouring pixels in its 4-neighbourhood $\mathbf{H}_4$ is white. The union of all boundary pixels of the same component is called *boundary* and denoted by $\mathscr{B}_{\mathsf{S}_k}$.

Let S_k be a connected component with boundary $\mathscr{B}_{\mathsf{S}_k}$. Then

- the boundary length $L(\mathscr{B}_{\mathsf{S}_k})$ is called *perimeter* and calculated by

$$L(\mathscr{B}_{\mathsf{S}_k}) = \#\{z : z \in \mathscr{B}_{\mathsf{S}_k}\}, \tag{2.43}$$

- the area $A(\mathsf{S}_k)$ is calculated as

$$A(\mathsf{S}_k) = \#\{z : z \in \mathsf{S}_k\}, \tag{2.44}$$

- for each component, we define a *ratio of its perimeter and area* as

$$R_{\mathsf{S}_k} = \frac{L(\mathscr{B}_{\mathsf{S}_k})}{A(\mathsf{S}_k)} = \frac{\#\{z : z \in \mathscr{B}_{\mathsf{S}_k}\}}{\#\{z : z \in \mathsf{S}_k\}}. \tag{2.45}$$

Example. An illustration of the algorithm is shown in Figure 2.2 where the ellipse-shaped component S_k is given and a disc $\mathbf{D}$ has been constructed with the centre at the boundary point of the set $\mathbf{X}$. The resulting estimate of the curvature is then $\frac{\#C}{\#C+\#D} = \frac{5}{5+8} = \frac{5}{13}$, while the respective ratio of the perimeter and area is $\frac{\#B}{\#E} = \frac{12}{19}$.

After identifying all the boundary points of the connected components S_k, it is necessary to compute the curvature $\kappa_k(z)$ at each point $z \in \mathscr{B}_{\mathsf{S}_k}$. From equation (2.41), we can see

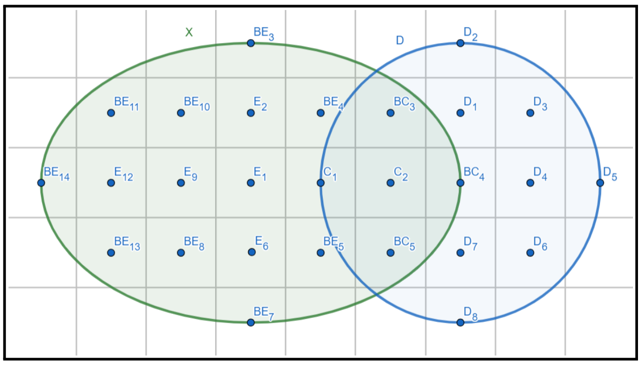


Figure 2.2: An illustration of the algorithm for estimating the curvature and the ratio of the perimeter and the area

that $\kappa_k(z)$ is proportional to

$$\kappa_k(z) \simeq \frac{A^*(\mathbf{D}(z,r))}{A(\mathbf{D}(z,r))} = O_{k,\mathbf{D}(z,r)}, \tag{2.46}$$

for appropriately chosen r. This fact will be used as a guideline for devising a testing characteristic.

Definition 2.6.4 (*Distribution of curvature*)**.** Let $O_{k,\mathbf{D}(z,r)}$ be the ratio as defined by equation (2.46). Define by

$$\tilde{\kappa}_{k,\mathbf{D}(.,r)}(u) = \frac{1}{L(\mathcal{B}_{\mathsf{S}_k})} \int_{\mathcal{B}_{\mathsf{S}_k}} \mathbb{1}_{[O_{k,\mathbf{D}(z,r)} \le u]} dz, \quad u \in \langle 0, 1 \rangle. \tag{2.47}$$

It is an analogy of the distribution function of the curvature at points on the boundary, with the difference that we work with highly dependent values here. From this function, an analogy to the density function can be defined as

$$t_{k,\mathbf{D}(.,r)}(u) = \tilde{\kappa}'_{k,\mathbf{D}(.,r)}(u) \tag{2.48}$$

which will be used as a functional characteristic describing the curvature.

Since we are working with binary pictures, i.e. with discrete values, we have to approximate the distribution function of the curvature.

Let $\mathbf{W}$ be a binary image containing a digitised realisation of a connected random set $\mathbf{X}_k$. For each boundary pixel z_i and a fixed radius $r \in \mathbb{N}$ we approximate

$$\kappa_{\hat{\mathbf{X}}_k}(z_i) = \frac{\#\{z_j \in \mathbf{W} : z_j \in \mathbf{D}(z_i, r) \cap \mathbf{X}_k\}}{\#\{z_j \in \mathbf{W} : z_j \in \mathbf{D}(z_i, r)\}}. \tag{2.49}$$

Using this approximation, we can further set

$$t_{\mathbf{X}_k}(u) = \frac{\#\{i \in \{1, \ldots, n\} : \hat{\kappa}(z_i) \in [u - 1/l, u)\}}{n}, \quad u = \frac{1}{l}, \frac{2}{l}, \ldots, 1, \tag{2.50}$$

where l is the number of pixels forming the disk $\mathbf{D}(., r)$.

Chapter 3

Classification of Realisations of Random sets

In this chapter, we will build our classifiers, using the tools provided in Chapter 2.

Consider a binary image S of a random set $\mathbf{S}$. For each $k = 1, ..., m$, where m represents the number of connected components $\mathsf{S}_1, ...\mathsf{S}_m$ inside the realisation S, we evaluate ratios $R_k = R_{\mathsf{S}_k}$ and functions describing the curvature $t_k = t_{\mathsf{S}_k}$ using (2.45) and (2.50), respectively.

3.1 Supervised classification

Let $(\mathcal{X}_i, Y_i), i = 1, ..., n$ be a sample of n independent pairs as mentioned in Section 2.3.1, where $\mathcal{X}$ denotes the functional random variable and Y_i denotes categorical response, and let (x, y) be an observation of the pair $(\mathcal{X}_i, Y_i)$, for $i = 1, ...n$.

For supervised classification, we will use a version of *k-nearest neighbours classifier* adapted to work with disretised functional data. Let us recall that it was defined in 2.3.8 as

$$\hat{p}_{g,k}(x) = \frac{\sum\limits_{i:\mathsf{y}_i=g}^{n} \mathcal{K}(h_k^{-1} d_s(x, \mathsf{x}_i))}{\sum\limits_{i=1}^{n} \mathcal{K}(h_k^{-1} d_s(x, \mathsf{x}_i))} \tag{3.1}$$

As the semi-metric d_s we use the $\mathcal{N}$-distance (2.36) with negative defined kernels given by (2.32) and (2.33). As the kernel $\mathcal{K}$ in (3.1) we decided to use the triangular kernel (2.26). The last step is to choose its tuning parameter k. Since we base our approach on [11], we use the same loss function

$$LCV(k, i_0) = \sum_{g=1}^{G} \left(\mathbb{1}_{[\mathsf{y}_{i_0}=g]} - p_{g,k}^{(-i_0)}(\mathsf{x}_{i_0}) \right)^2, \tag{3.2}$$

where

$$i_0 = \arg \min_{i=1,\dots,n} \hat{\mathcal{N}}(x, \mathsf{x}_i), \tag{3.3}$$

$$p_{g,k}^{(-i_0)}(\mathsf{x}_{i_0}) = \frac{\sum\limits_{i:\mathsf{y}_i=g, i\neq i_0}^{n} \mathcal{K}(h_{k(\mathsf{x}_{i_0})}^{-1} \hat{\mathcal{N}}(\mathsf{x}_i, \mathsf{x}_{i_0}))}{\sum\limits_{i=1}^{n} \mathcal{K}(h_{k(\mathsf{x}_{i_0})}^{-1} \hat{\mathcal{N}}(\mathsf{x}_i, \mathsf{x}_{i_0}))} \tag{3.4}$$

and obtain optimal number of nearest neighbours k_{LCV} at x_{i_0} as

$$k_{LCV}(\mathsf{x}_{i_0}) = \arg \min_{k} LCV(k, i_0). \tag{3.5}$$

Finally, we have

$$\hat{p}_g^{(LCV)}(x) = \frac{\sum\limits_{i:\mathsf{y}_i=g}^{n} \mathcal{K}(h_{LCV}^{-1}(\mathsf{x}_{i_0}) \hat{\mathcal{N}}(x, \mathsf{x}_i))}{\sum\limits_{i=1}^{n} \mathcal{K}(h_{LCV}^{-1}(\mathsf{x}_{i_0}) \hat{\mathcal{N}}(x, \mathsf{x}_i))}, \tag{3.6}$$

where h_{LCV} is the bandwidth corresponding to $k_L CV$ which depends on the functional point at which $\hat{p}_{g,k}^{(LCV)}(x)$ is evaluated. The classification rule is then given by

$$y = \arg \max_{g \in \{1,\dots,G\}} \hat{p}_g^{(LCV)}(x). \tag{3.7}$$

3.2 Unsupervised classification

For unsupervised classification, we will use a version of the *k-means classifier* described in Section 2.3.2 adapted to work with discretised functional data. As $\mathcal{X}_1, \dots, \mathcal{X}_n$ we take the ratios of the components $\mathsf{S}_1, \dots, \mathsf{S}_n$ as points in the 1-dimensional space, the functions describing the curvature as points in l-dimensional space, where l is the number of points in which the function describing the curvature is evaluated, and when we consider both ratio and curvature, we consider this couple as a point in $(l+1)$-dimensional space, where a special weight is given to the ratio, see the following chapter. The output of the algorithm is the set of subsets $\mathcal{S}_i, i = 1, \dots, k$, as introduced in Section 2.3.2, which we consider as the groups of $\mathcal{X}_j$'s belonging to the i-th class after the algorithm 2.3.9. This means that for each realisation of a random set, after evaluating the respective characteristic x for each component, we calculate the mean as defined by (2.12). Since our approach is based on $\mathcal{N}$-distance, we will use the same kernel $\mathcal{L}$ as defined by (2.32) and (2.33) to calculate $\mathcal{N}$-distance using (2.36). The classification rule is then given by

$$\arg\min_{\mathcal{S}^k} \sum_{i=1}^{k} \sum_{x \in \mathcal{S}_i} \hat{\mathcal{N}}(x - \overline{\mathsf{x}_{\mathcal{S}_i}})^2. \tag{3.8}$$

Chapter 4

Simulation Study

The primary objective of this chapter is to demonstrate the functionality of the classifiers introduced in Chapter 3 using simulated data. Specifically, we will focus on the three processes outlined in Section 2.5.

4.1 Simulated Data

The first step in the validation procedure is to simulate the data of the random sets. We will focus mainly on models that have already been studied by different authors, see Figure 4.1. Namely, we will use the Boolean model, see Definition 2.5.4, which is widely studied, the second and the third model are the cluster (studied in [25] and [6]) and the repulsive model (studied in [28], [25] and [29]), both simulated using Quermass-interaction process as described in Definition 2.5.5 with suitably chosen parameters using the algorithm from [30]. The simulated data were kindly provided by the authors of [28] and [25]. For each model we take 200 realisations.

Due to the fact that the models mentioned above significantly differ by the number of components in their respective realisations, we had to determine the optimal number of components that we will consider. Analogously to [31], where the same models and the same functional characteristic were studied, we decided to use samples of size 10, 20 and 'all' (where 'all' marks the number that is equal to the number of components in the realisation with smaller number of components between the two from which we calculate semi-metrics).

Once we have defined the appropriate classifier for the supervised classification, see (3.6), we should apply the procedure to the simulated data.

To obtain functional characteristics, we have to choose the appropriate value for the radius r that is used for calculating the curvature at the boundary point, see (2.49).

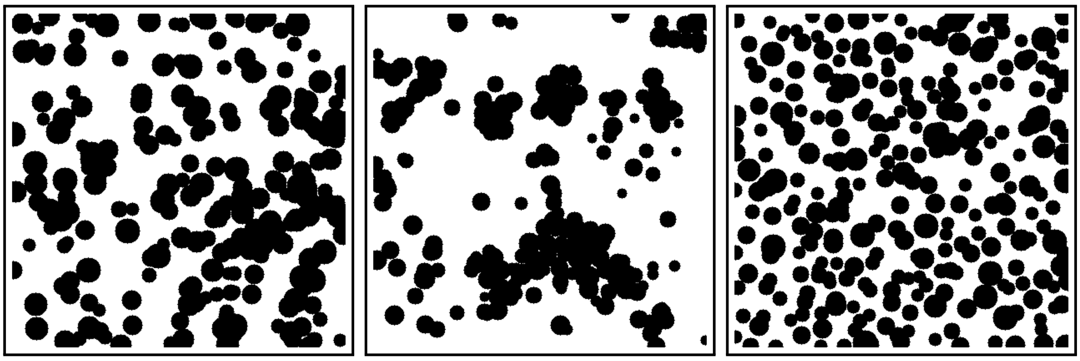


Figure 4.1: Previously studied models: the Boolean, the cluster and the repulsive model, respectively

The area of the disc with radius r (measured in pixels) is given [32] by

$$A(\mathbf{D}(.,r)) = 1 + 4 \cdot \sum_{j \geq 0} \left(\left\lfloor \frac{r^2}{4j+1} \right\rfloor - \left\lfloor \frac{r^2}{4j+3} \right\rfloor \right). \tag{4.1}$$

A list with values for $r = 1, ..., 10000$ can be found in [33]. For our study, we will use $r = 3$ and $r = 5$, which correspond to $A(\mathbf{D}(.,3)) = 29$ and $A(\mathbf{D}(.,5)) = 81$, as they were used in [31], because choosing a disc with a large area would lead to a great mistake, since the disc would not be able to detect local changes in curvature due to discretisation. Note that the values $A(\mathbf{D}(.,3))$ and $A(\mathbf{D}(.,3))$ are the values used for l in 2.50.

Once we have the input data, we estimate the ratios and curvatures. In this way, we obtain functional data which is then passed to a classifier.

4.2 Supervised classification

Data are split into train set and test set with a 3:1 ratio (which means that 75% of the realisations is used for training, while 25% is used for testing the performance of the classifier). We decided to use three settings in order to study the influence of the number of realisations on the classification:

- in the first setting we used a sample of 20 randomly chosen realisations from each model (further 'class'), Boolean (class 'B'), cluster (class 'C') and repulsive (class 'R'), meaning that in the training set we have 45 realisations (15 of each class, 'B', 'C' and 'R') in the training set and 15 realisations for testing purpose (5 of each class)
- in the second setting we used a sample of 50 randomly chosen realisations from each class, meaning that in the training set we have 111 realisations (37 of each class) in the training set and 39 realisations for testing purpose (13 of each class)

- in the third setting we used a sample of 100 randomly chosen realisations from each class, meaning that we have 225 realisations (75 of each class) in the training set and 75 realisations for testing purpose (25 of each class).

Each of the settings mentioned above is then split into three subsettings according to the characteristic which is used for discrimination, namely 'Ratio' (using only the ratio), 'Curvature' (using only the curvature) and 'Both' (using both the ratio and the curvature). After that, the classifier is learnt three times for different numbers of components, which we use for calculating the $\mathcal{N}$-distance (i.e. 10, 20 and 'all', as mentioned above). After the learning stage, we use the test set and predict the labels using the posterior probabilities calculated for each class, as defined in (3.6). The classification results for each setting are shown in Figures 4.2 (20 realisations), 4.3 (50 realisations) and 4.4 (100 realisations) for data obtained using the osculating circle with radius $r = 3$, and Figures 4.6 (20 realisations), 4.7 (50 realisations) and 4.8 (100 realisations) for data obtained using the osculating circle with radius $r = 5$.

Focussing on the results when considering only 20 realisations shown in Figure 4.2, we observe that the highest overall misclassification rate was for the smallest sample size (of 10 components) as expected, it drops for a larger sample size (of 20 components), while the best performance was when considering 'all' components. Furthermore, we observe that the most problematic part was the classification of the cluster model. Similar problems occurred in the simulation study in [31]. This is probably due to the fact that the cluster model contains a few larger components, a number of (Boolean-like) 2-to-10-disc components, and a greater amount of (repulsive-like) single-disc components. Among the three subsettings, the lowest misclassification rate was when considering both characteristics. This reflects the results obtained in [31] where it was concluded that both characteristics were necessary to correctly discriminate between different processes.

Taking a look at the results when considering 50 realisations shown in Figure 4.3, we can see that the classifier behaves in the expected way: the misclassification rate is highest when taking into account the smallest sample size of 10 components, it drops for a higher sample size of 20 components, and it is the lowest when considering 'all' components. Comparing the results with the previous setting (for only 20 realisations), we can see that the maximum misclassification rate for each characteristic separately is higher, while it decreases when considering both characteristics, which is in accordance with the results from [31].

The results for 100 realisations shown in Figure 4.4 indicate that the amount of data higher than some threshold does not make the classifier significantly more precise, as the highest misclassification rate is comparable to the setting working with 50 realisations, but also indicates the dependence of the classification precision on the number of components considered.

Each setting is run 50 times in order to obtain box plots of the misclassification rate shown in Figure 4.5. The maximum and minimum misclassification rates for each setting are shown in Table 4.1.

Number of realisations	20			50			100		
Number of components	10	20	'all'	10	20	'all'	10	20	'all'
Characteristics considered	Misclassification rate [%]								
Both	53.3	26.7	13.3	30.8	12.8	5.1	30.7	10.7	2.7
Curvature	66.7	53.3	20	53.8	30.8	12.8	54.7	28	5.3
Ratio	46.7	26.7	20	30.8	18	12.8	25.3	12	5.3
Both	0	0	0	5.1	0	0	10.7	0	0
Curvature	26.7	6.7	0	25.6	10.3	0	30.7	8	0
Ratio	6.7	0	0	2.6	2.6	0	9.3	1.3	0

Table 4.1: Maximum and minimum (underlined) misclassification rates obtained after 50 runs of k-nearest neighbours algorithm for different settings (20, 50 and 100 realisations) and respective subsettings (Both, Curvature and Ratio) when using samples of 10, 20 and 'all' components, respectively. Note that the data used are the data obtained using an osculating circle of radius $r = 3$.

To illustrate how the choice of the radius of the osculating circle affects the performance of the classifier, we performed the same procedures as above for the data obtained using the osculating circle of radius $r = 5$. The highest overall misclassification rate when considering 20 realisations, see Figure 4.6, was again for the smallest sample size of 10 components, as expected. However, comparing the results with the results obtained above (when using $r = 3$), we can see that the misclassification rate for all three characteristics is equal or lower. The best performance was again when considering 'all' components. The unexpected increase in the misclassification rate with a growing sample size when considering only the ratio is again present, since the results shown in Figure 4.6 were obtained for the same realisations and components (that is, the same seed was used for randomly choosing) and since the size of the osculating circle does not affect the value of the ratio. Further, we see that classification based on only the curvature performs better in all three cases (for 10, 20 and 'all' components). It is due to the fact that the curvature is evaluated in more positions, leading to greater versatility between classes.

The results when considering 50 realisations shown in Figure 4.7, suggest that the classifier behaves in the expected way: the misclassification rate is highest when taking into account the smallest sample size of 10 components, it drops for a higher sample size of 20 components, and it is the lowest when considering 'all' components. Comparing the results with the previous ones (for $r = 3$), we can see that, again, classification based on only the curvature gives slightly better results.

The results for 100 realisations shown in Figure 4.8 are the best obtained since the misclassification rate obtained drops in all three cases.

Each setting is run, as above, 50 times to obtain box plots. The results are shown in Figure 4.9. The maximum and minimum misclassification rates are shown in Table 4.2.

Number of realisations	20			50			100		
Number of components	10	20	'all'	10	20	'all'	10	20	'all'
Characteristics considered	Misclassification rate [%]								
Both	53.3	13.3	6.7	30.8	15.4	2.6	28.2	9.3	1.3
Curvature	60	40	6.7	51.3	23.1	2.6	45.3	20.5	2.6
Ratio	46.7	26.7	20	30.8	18	12.8	25.6	12	5.3
Both	$\underline{6}.7$	$\underline{0}$	$\underline{0}$	$\underline{7}.7$	$\underline{0}$	$\underline{0}$	$\underline{9}.3$	$\underline{0}$	$\underline{0}$
Curvature	$\underline{2}0$	$\underline{0}$	$\underline{0}$	$\underline{2}8.2$	$\underline{2}.6$	$\underline{0}$	$\underline{2}4$	$\underline{5}.3$	$\underline{0}$
Ratio	$\underline{6}.7$	$\underline{0}$	$\underline{0}$	$\underline{2}.6$	$\underline{2}.6$	$\underline{0}$	$\underline{9}.3$	$\underline{1}.3$	$\underline{0}$

Table 4.2: Maximum and minimum (underlined) misclassification rates obtained after 50 runs of k-nearest neighbours algorithm for different settings (20, 50 and 100 realisations) and respective subsettings (Both, Curvature and Ratio) when using samples of 10, 20 and 'all' components, respectively. Note that the data used are the data obtained using an osculating circle of radius $r = 5$.

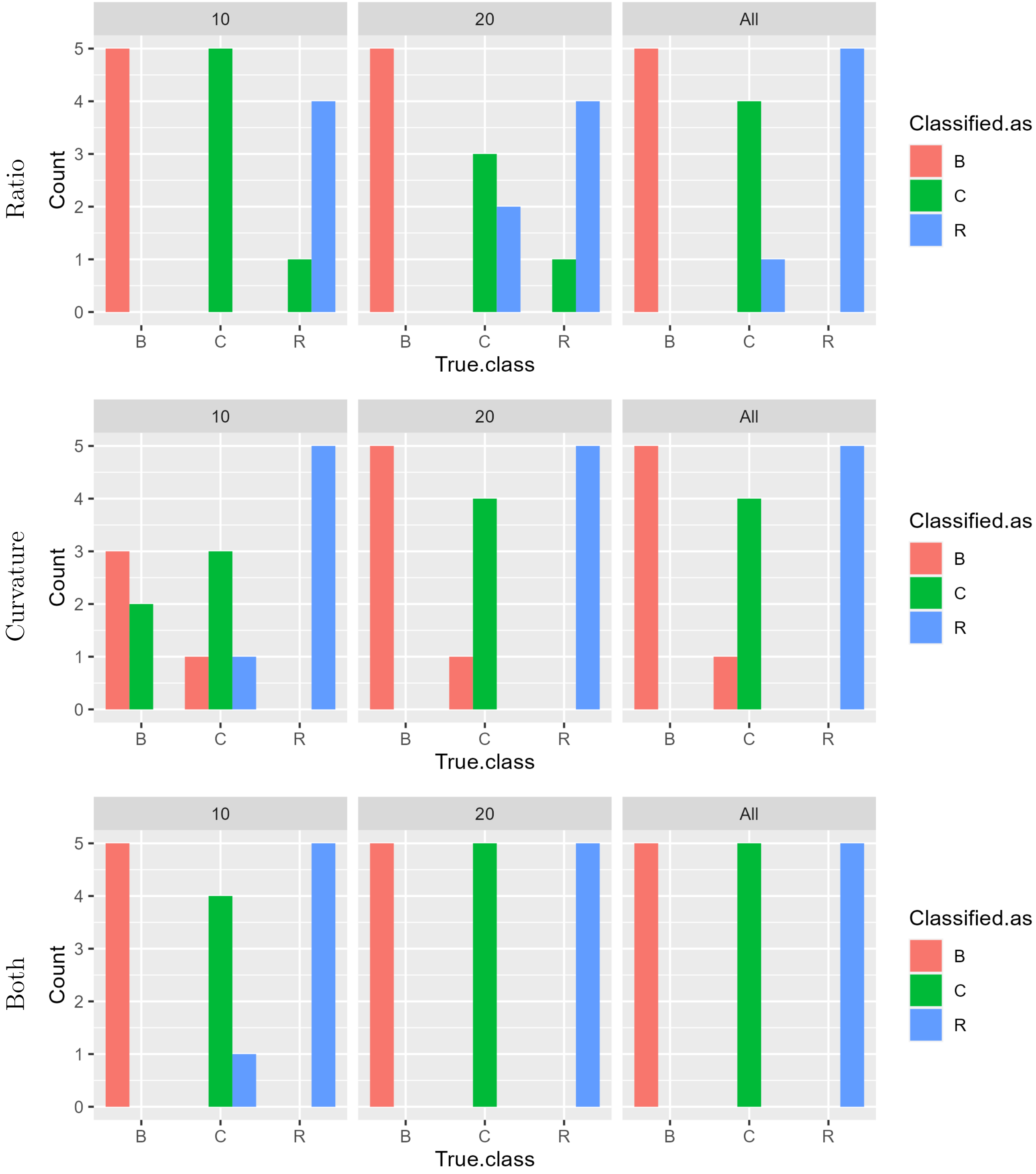


Figure 4.2: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20 and 'all' components, respectively. Misclassification rates are 6. 7%, 20% and 6. 7% for 10, 20 and all components, respectively, when using only the ratio, 26.7%, 6.7% and 6.7% when using only the curvature, and 6.7%, 0% and 0% when using both characteristics for a sample of 20 realisations that were osculated by a disc of radius $r = 3$.

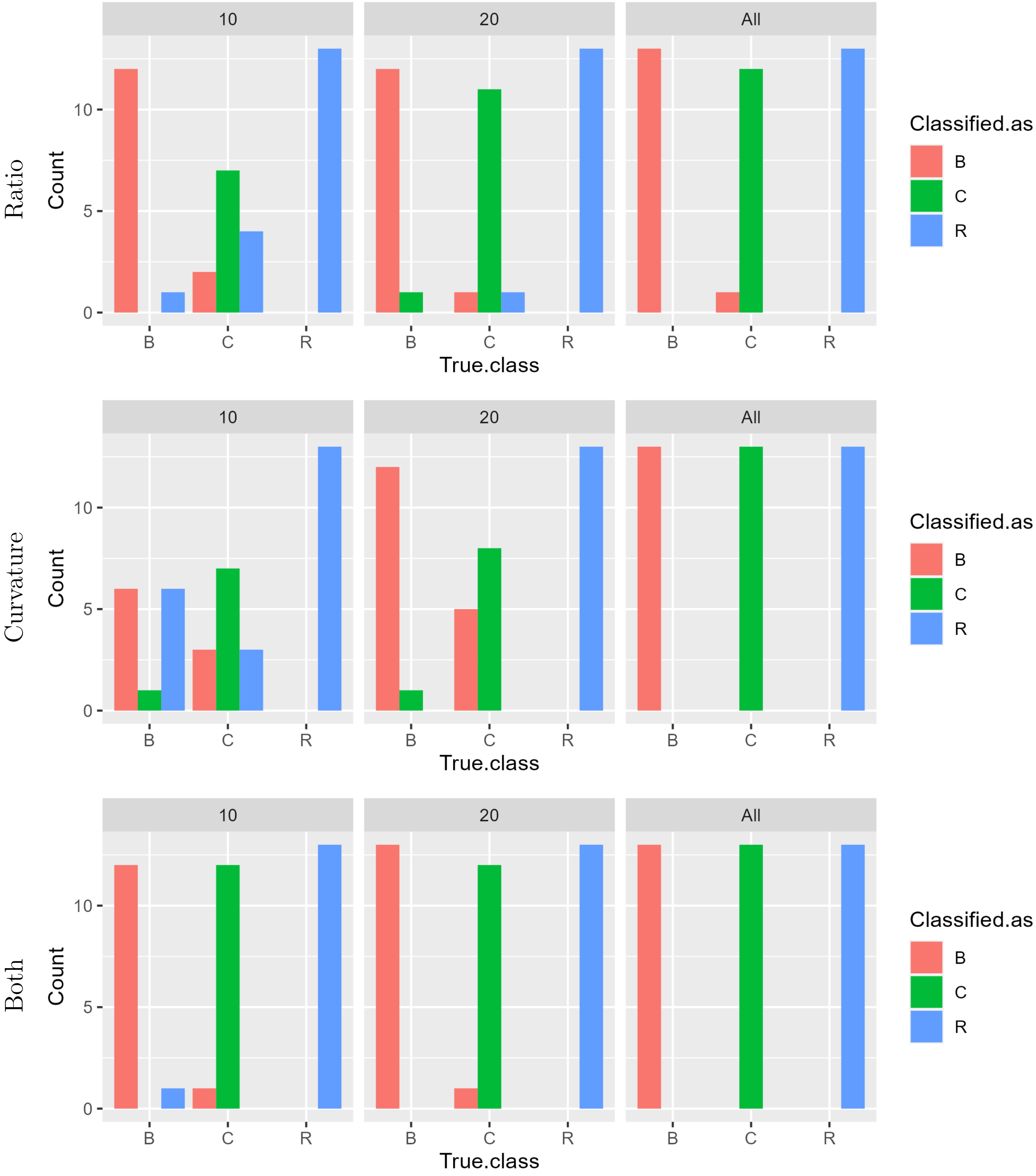


Figure 4.3: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20 and 'all' components, respectively. Misclassification rates are 17.9%, 7.7% and 2.6% for 10, 20 and 'all' components, respectively, when using only the ratio, 33.3%, 15.4% and 0% when using only the curvature, and 5.1%, 2.6% and 0% when using both characteristics for a sample of 50 realisations that were osculated by a disc of radius $r = 3$.

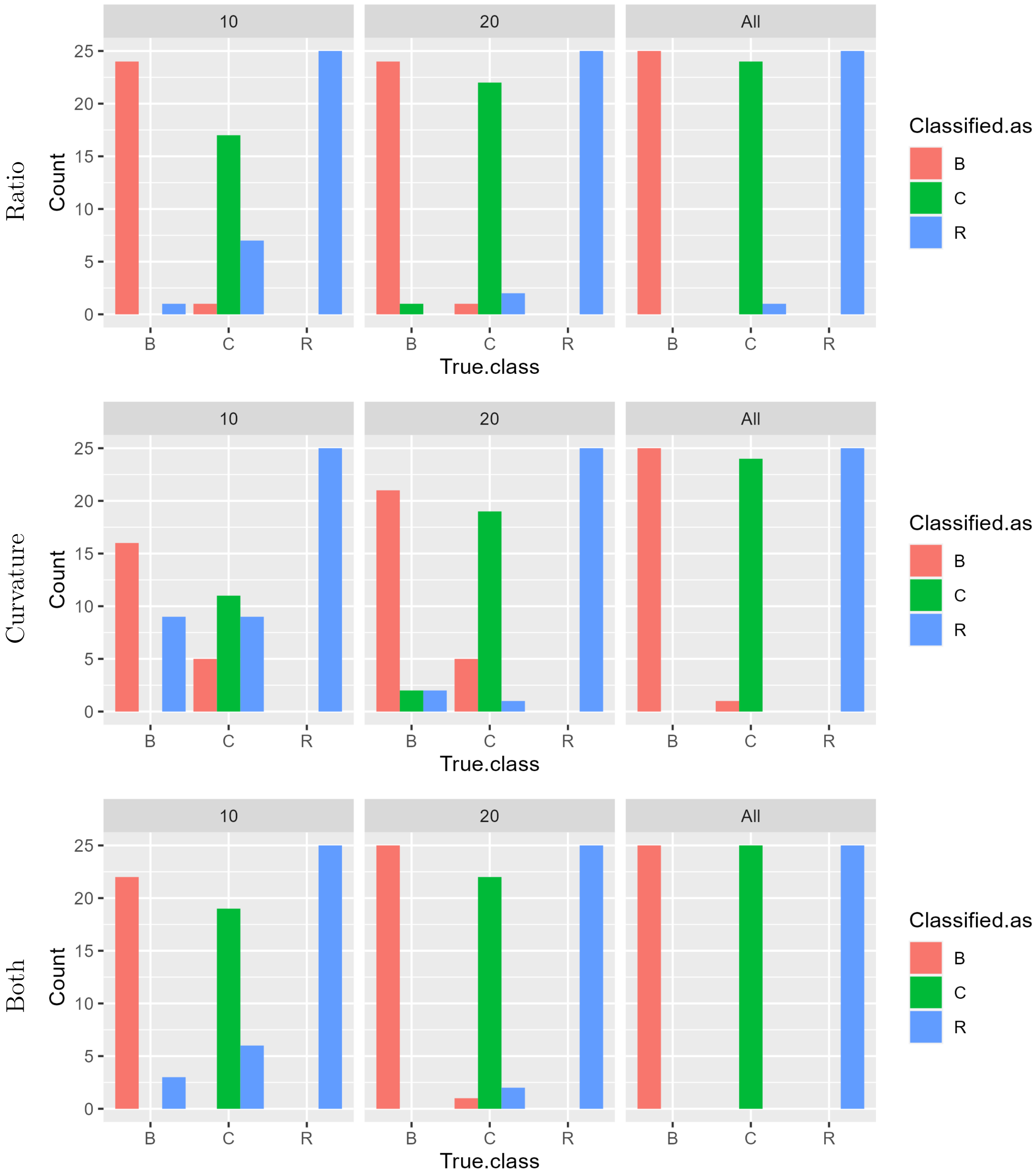


Figure 4.4: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20 and 'all' components, respectively. Misclassification rates are 12%, 5.3% and 1.3% for 10, 20 and 'all' components, respectively, when using only the ratio, 30.7%, 13.3% and 1.3% when using only the curvature, and 12%, 4% and 0% when using both characteristics for a sample of 100 realisations that were osculated by a disc of radius $r = 3$.

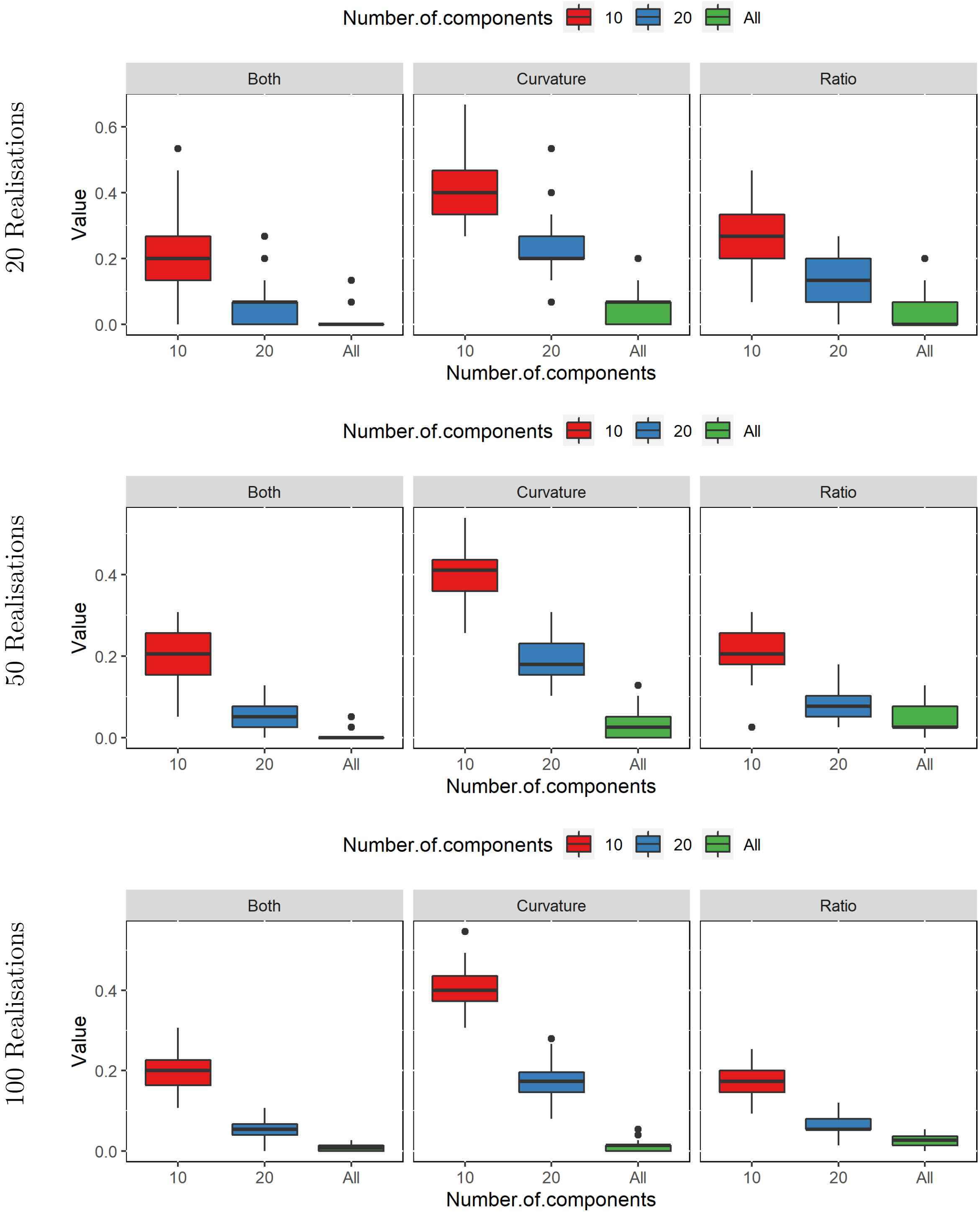


Figure 4.5: Boxplots of misclassification rate for 50 runs of k-nearest neighbours algorithm when considering samples of 20 (top), 50 (central) and 100 (bottom) realisations using both ratio and curvature, only the curvature and only the ratio for discrimination, respectively. For each setting, misclassification rates for different number of components considered (namely 10, 20 and 'all') are shown. Note that the characteristics were obtained using an osculating disc of radius $r = 3$.

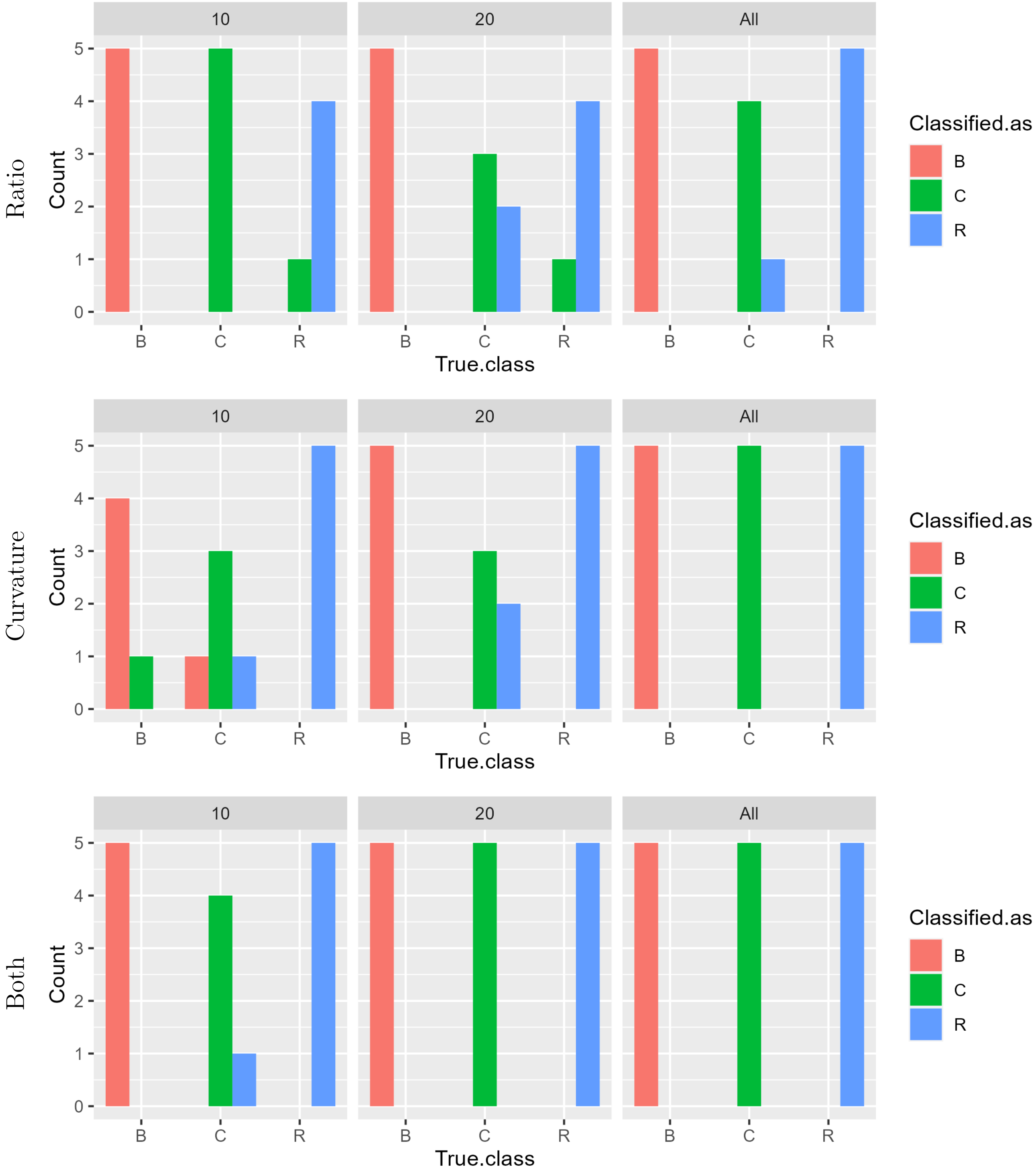


Figure 4.6: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 6.7%, 20% and 6.7% for 10, 20 and 'all' components, respectively, when using only the ratio, 20%, 13.3% and 0% when using only the curvature, and 6.7%, 0% and 0% when using both characteristics for a sample of 20 realisations that were osculated by a disc of radius $r = 5$.

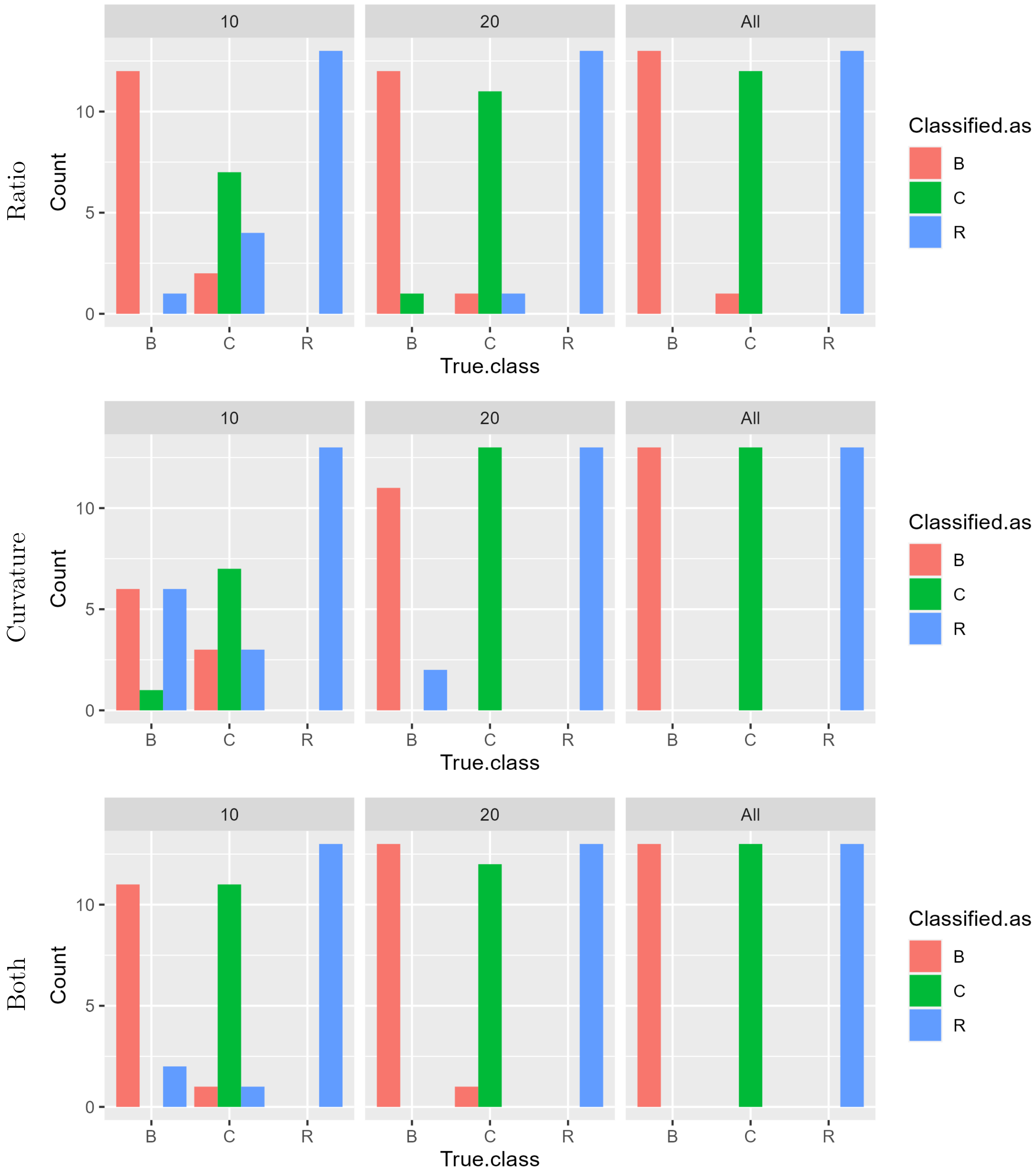


Figure 4.7: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 17.9%, 7.7% and 2.6% for 10, 20 and 'all' components, respectively, when using only the ratio, 33.3%, 5.1% and 0% when using only the curvature, and 10.3%, 2.6% and 0% when using both characteristics for a sample of 50 realisations that were osculated by a disc of radius $r = 5$.

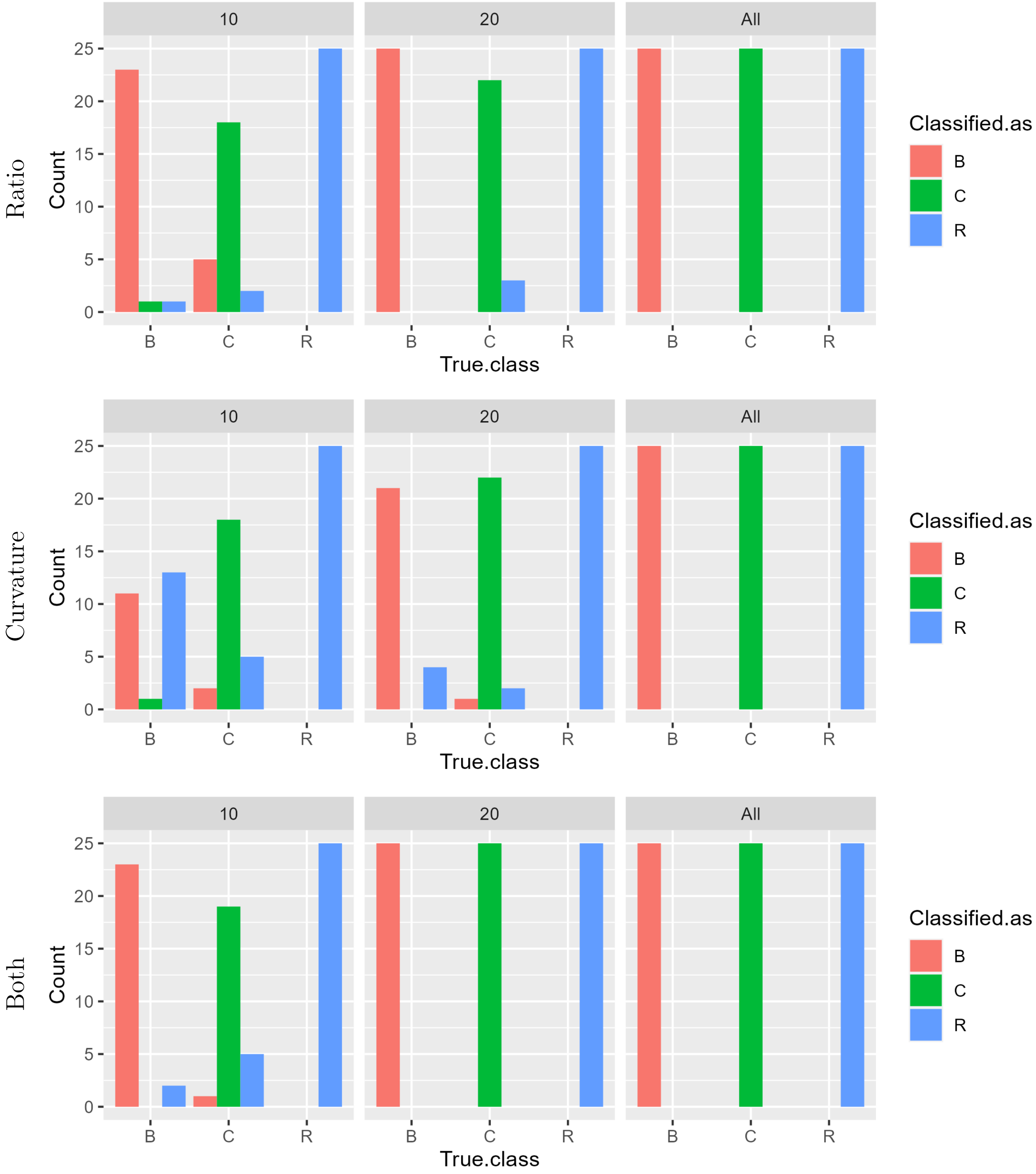


Figure 4.8: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 12%, 4% and 0% for 10, 20 and 'all' components, respectively, when using only the ratio, 28%, 9.3% and 0% when using only the curvature, and 10.7%, 0% and 0% when using both characteristics for a sample of 100 realisations that were osculated by a disc of radius $r = 5$.

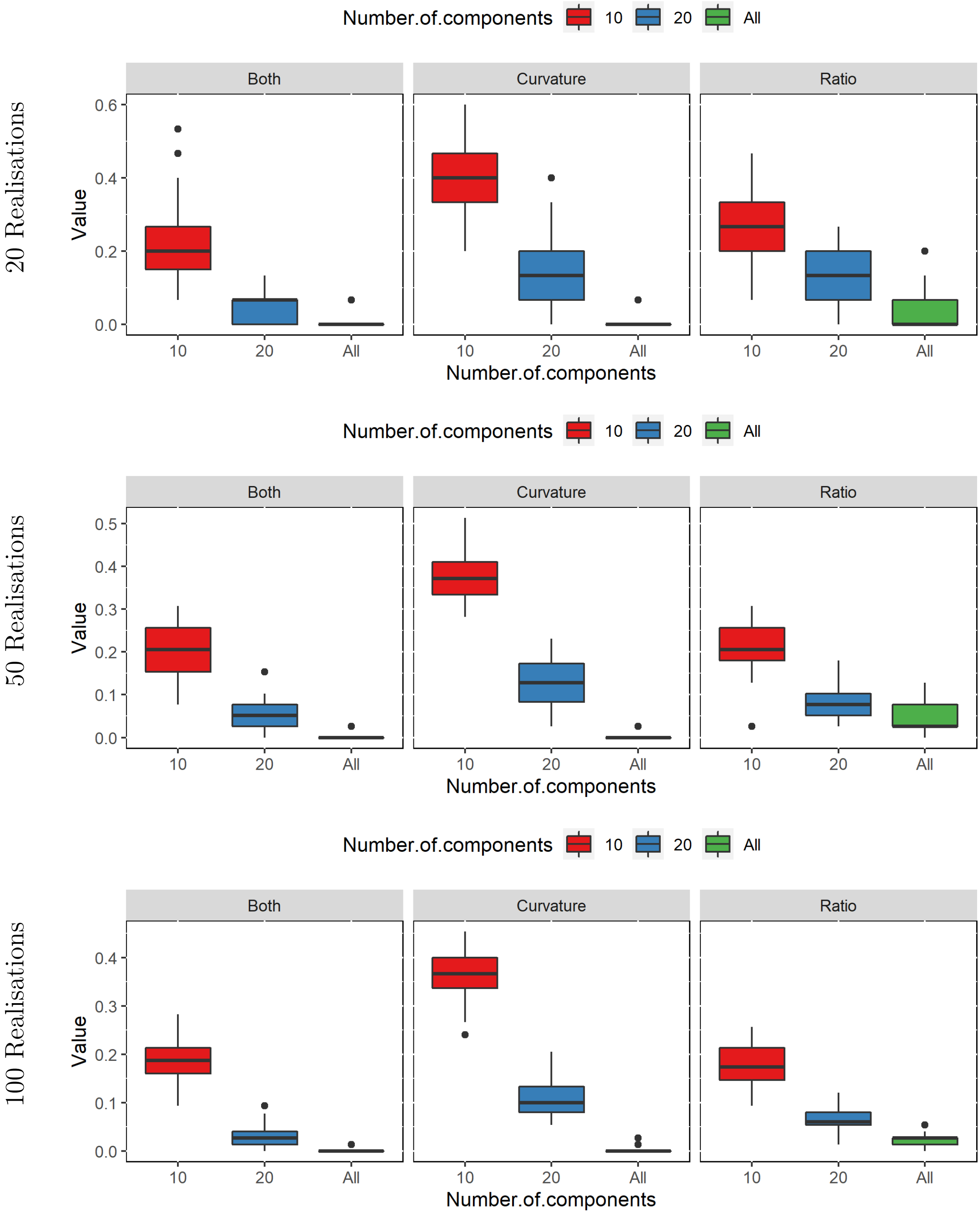


Figure 4.9: Boxplots of misclassification rate for 50 runs of k-nearest neighbours algorithm when considerring samples of 20 (top), 50 (central) and 100 (bottom) realisations using both ratio and curvature, only the curvature and only the ratio for discrimination, respectively. For each setting, misclassification rates for different number of components considered (namely 10, 20 and 'all') are shown. Note that the characteristics were obtained using an osculating disc of radius $r = 5$ on the simulated data.

4.3 Unsupervised classification

In the unsupervised classification case, we use the k-means algorithm, which randomly initialises the cluster centres n_{init} times and uses the set of clusters with the best inertia [34]. For our study, we used $n_{init} = 20$ for all settings listed below.

Similarly as for the supervised classification, the data are again split into train set and test set with a 3:1 ratio (which means that 75% of the realisations is used for training, while 25% is used for testing the performance of the classifier). Since we wanted to test how much the amount of data at our disposal affects the performance of the classifier, we decided to use 3 sizes of samples, namely the sample of 20, 50 and 100 randomly chosen realisations from each class.

Each of the above-mentioned settings is then split into three subsettings according to the characteristic which is used for discrimination, i.e. 'Ratio', 'Curvature', and 'Both'. Note that we set the weight for the ratio to be 1000 to compensate for the length and the values describing the curvature. After that, the classifier performance is tested for different numbers of components, which we use to calculate the mean. The classification results for each setting are shown in Figures 4.10 (20 realisations), 4.11 (50 realisations) and 4.12 (100 realisations) for data obtained using osculating circle with radius $r = 3$, and Figures 4.14 (20 realisations), 4.15 (50 realisations) and 4.16 (100 realisations) for data obtained using osculating circle with radius $r = 5$.

Focussing on Figure 4.10 we can see that the highest overall misclassification rate was for the smallest sample size (of 10 components) as expected, while the best performance was when considering 'all' components. Furthermore, we observe that for a smaller data set (of only 20 realisations, out of which 15 are used for training), the classifier again had the greatest problem correctly classifying the cluster model. We justify it by the same fact as in the unsupervised classification case.

Taking a look at the results when considering 50 realisations shown in Figure 4.11, we can see that the highest overall misclassification rate was for the smallest sample size (of 10 components) as expected, while the best performance was when considering 'all' components. Compared to the results when only 20 realisations are considered, the misclassification rate when only the ratio is considered decreases, which is expected. However, we observe an increase in the misclassification rate when using only the curvature and both the ratio and the curvature, which leads to a situation where the results when considering both characteristics and 'all' components are not the best. This is unexpected since we assume that, with the growing sample, the estimate of the mean should better reflect the population (i.e., class) mean. We can see that the increase in the misclassification rate is due to the decrease in the precision of classifying the Boolean model.

Contrary to the unsupervised classification case, the results for 100 realisations shown in Figure 4.12 indicate that the amount of data at hand significantly affects the performance of the classifier. For a sample of 10 components, the misclassification rate is the highest when considering both characteristics, while for ’all’ components it is the lowest when both characteristics are considered. However, the results obtained are not the best results for any subsetting (’Ratio’, ’Curvature,’ or ’Both’). The highest overall misclassification rate was again for the smallest sample size, while the best performance was when considering ’all’ components, as expected.

Each setting is run 50 times in order to obtain box plots of the misclassification rate, and the results are shown in Figure 4.13. The maximum and minimum misclassification rates after 50 runs for each setting are shown in Table 4.3.

Number of realisations	20			50			100		
Number of components	10	20	’all’	10	20	’all’	10	20	’all’
Characteristics considered	Misclassification rate [%]								
Both	86.7	93.3	73.3	89.7	92.3	48.7	82	84	36
Curvature	86.7	73.3	86.7	89.7	84.6	51.3	86.7	88	49.3
Ratio	86.7	87.2	86.7	89.7	87.2	84.6	86.7	85.5	81.3
Both	26.7	13.3	0	33.3	23.1	7.7	41.3	33.3	17.3
Curvature	26.7	13.3	6.7	30.8	25.6	15.4	40	26.7	14.7
Ratio	13.3	28.3	13.3	28.2	20.5	12.8	26.7	29.3	20

Table 4.3: Maximum and minimum (underlined) misclassification rates obtained after 50 runs of k-means algorithm for different settings (20, 50 and 100 realisations) and respective subsettings (Both, Curvature and Ratio) when using samples of 10, 20 and ’all’ components, respectively. Note that the data used are the data obtained using an osculating circle of radius $r = 3$.

To illustrate how the choice of the radius of the osculating circle affects the performance of the classifier, we performed the same procedures as above for the data obtained using the osculating circle of radius $r = 5$. In Figure 4.14 we can see that for a data set of only 20 realisations, classifiers based on only the ratio and only the curvature give better results than for the data obtained with $r = 3$. The different results for only the ratio, even for the data sampled with the same seed as in the $r = 3$ case (meaning that the same realisations and the same components in each realisation are chosen to calculate the respective means), are due to the fact that the initial cluster centres in the k-means procedure are chosen differently. This difference probably could be avoided by setting a really high number of initialisations from which the one with the best inertia is chosen (the presented results were obtained for 20 initialisations). When classification is based on both characteristics simultaneously, the misclassification rate for 10 and 20 components is higher, while for ’all’ components it is comparable.

When considering 50 realisations, the results for samples of 10 and 20 components, see Figure 4.15, are comparable to the results obtained for $r = 3$. However, the results obtained when considering ’all’ components are much better, which is justified by the fact that the curvature is evaluated in more positions.

The results for 100 realisations shown in Figure 4.16 are comparable or slightly better to those obtained for $r = 3$ when considering the characteristics separately. However, when both characteristics are considered simultaneously, the results are slightly better for 10 and 20 components, and worse when considering ’all’ components. This means that a case-specific optimal weight for ratio should probably be applied.

Each setting is run, as above, 50 times in order to obtain box plots. The results are shown in Figure 4.17. The maximum and minimum misclassification rates after 50 runs are shown in Table 4.4.

Number of realisations	20			50			100		
Number of components	10	20	’all’	10	20	’all’	10	20	’all’
Characteristics considered	Misclassification rate [%]								
Both	86.7	80	53.3	82.1	87.2	25.6	86.7	84	20
Curvature	86.7	86.7	46.7	89.7	92.3	25.6	85.3	84	20
Ratio	86.7	80	80	87.2	87.2	84.6	84	89.3	80
Both	26.7	20	0	33.3	20.1	0	36	26.7	6.7
Curvature	26.7	6.7	0	33.3	15.4	5.1	38.7	29.3	5.3
Ratio	26.7	13.3	13.3	28.2	15.4	17.9	36	28	21.3

Table 4.4: Maximum and minimum (underlined) misclassification rates obtained after 50 runs of k-means algorithm for different settings (20, 50 and 100 realisations) and respective subsettings (Both, Curvature and Ratio) when using samples of 10, 20 and ’all’ components, respectively. Note that the data used are the data obtained using an osculating circle of radius $r = 5$.

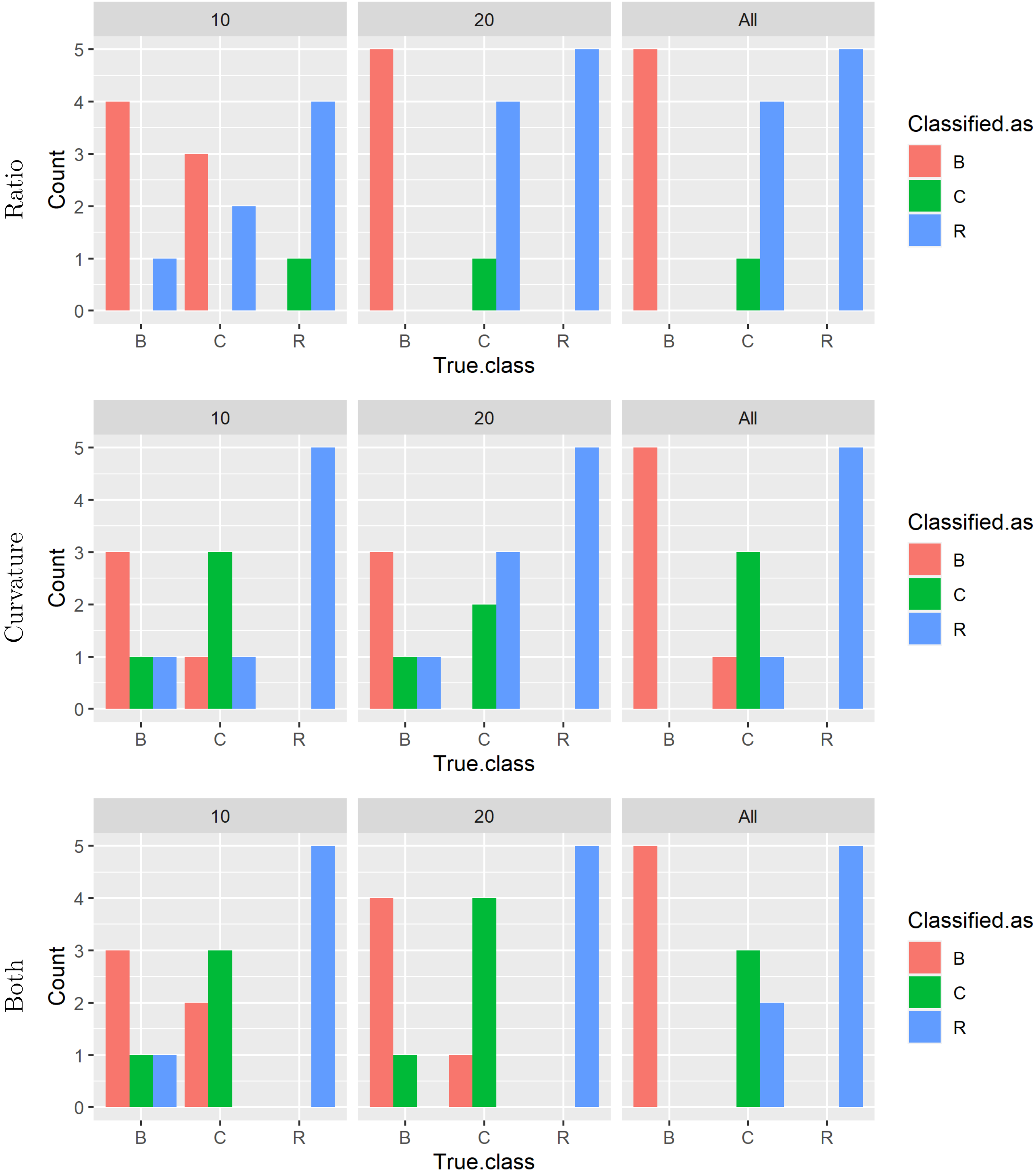


Figure 4.10: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 46.7%, 26.7% and 26.7% for 10, 20 and 'all' components, respectively, when using only the ratio, 26.7%, 33.3% and 13.3% when using only the curvature, and 26.7%, 13.3% and 13.3% when using both characteristics for a sample of 20 realisations that were osculated by a disc of radius $r = 3$.

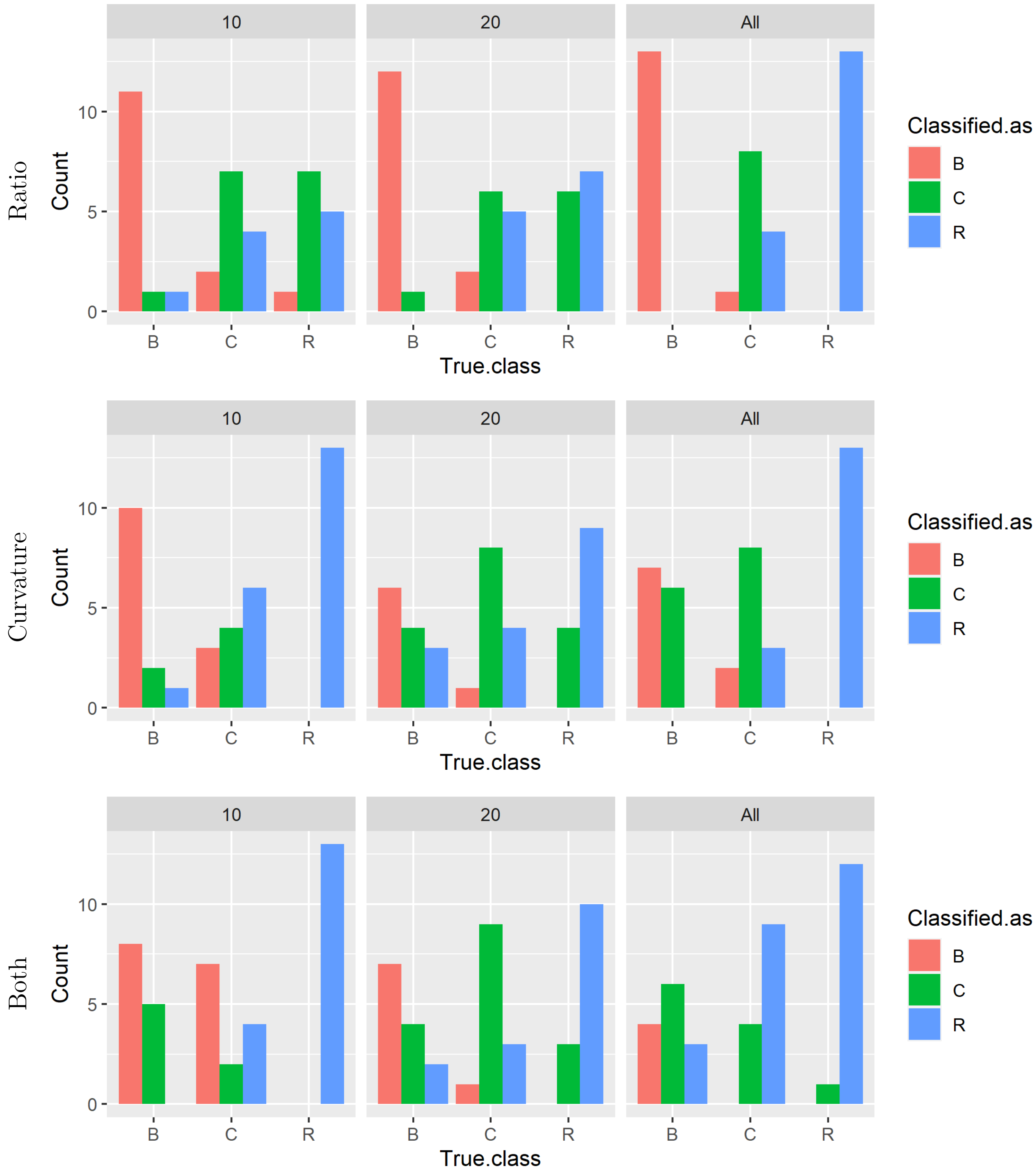


Figure 4.11: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 41%, 35.9% and 12.8% for 10, 20 and 'all' components, respectively, when using only the ratio, 30.8%, 41% and 28.2% when using only the curvature, and 41%, 33.3% and 48.7% when using both characteristics for a sample of 50 realisations that were osculated by a disc of radius $r = 3$.

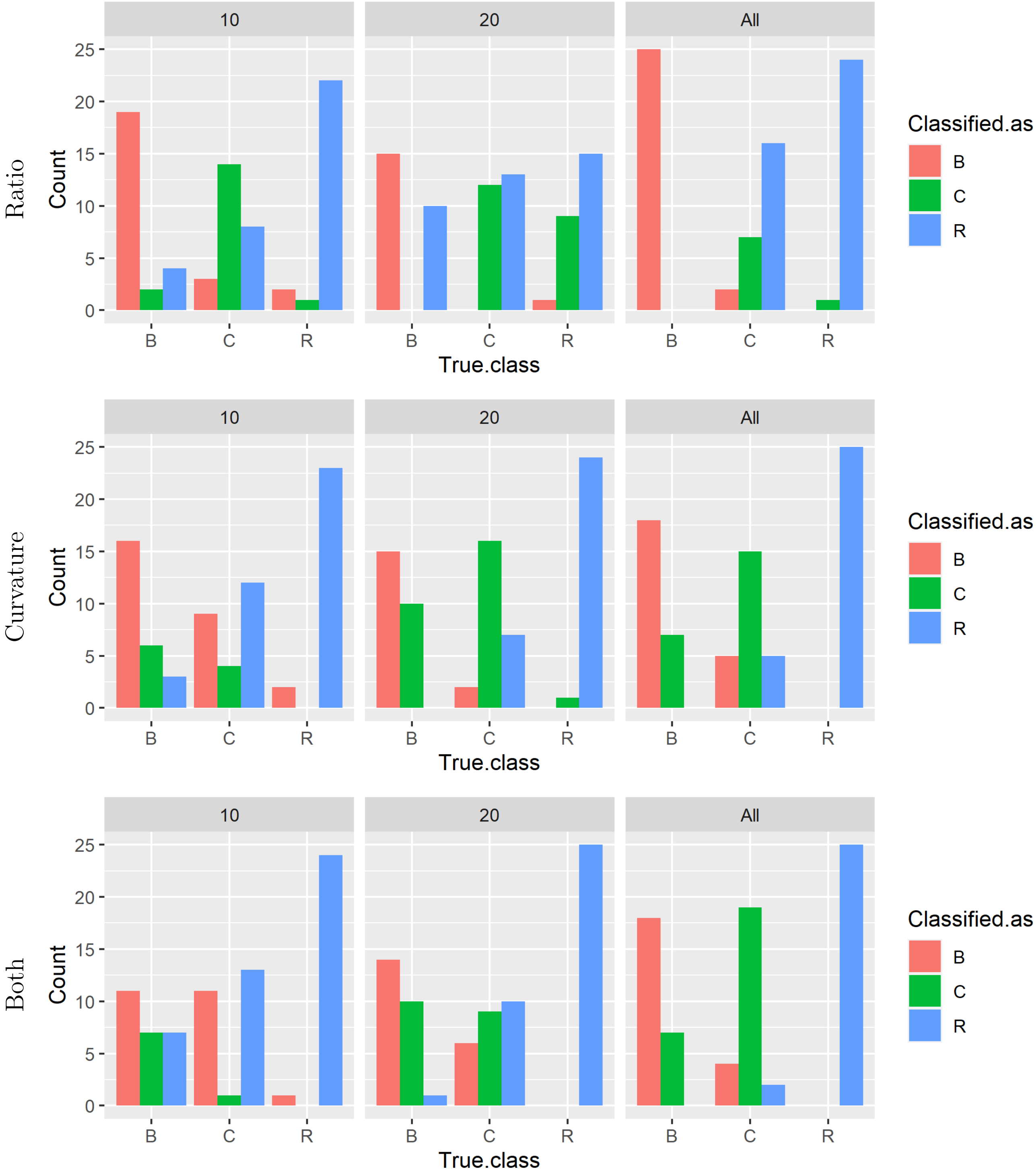


Figure 4.12: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 26.7%, 44% and 25.3% for 10, 20 and 'all' components, respectively, when using only the ratio, 42.7%, 26.7% and 22.7% when using only the curvature, and 52%, 36% and 17.3% when using both characteristics for a sample of 100 realisations that were osculated by a disc of radius $r = 3$.

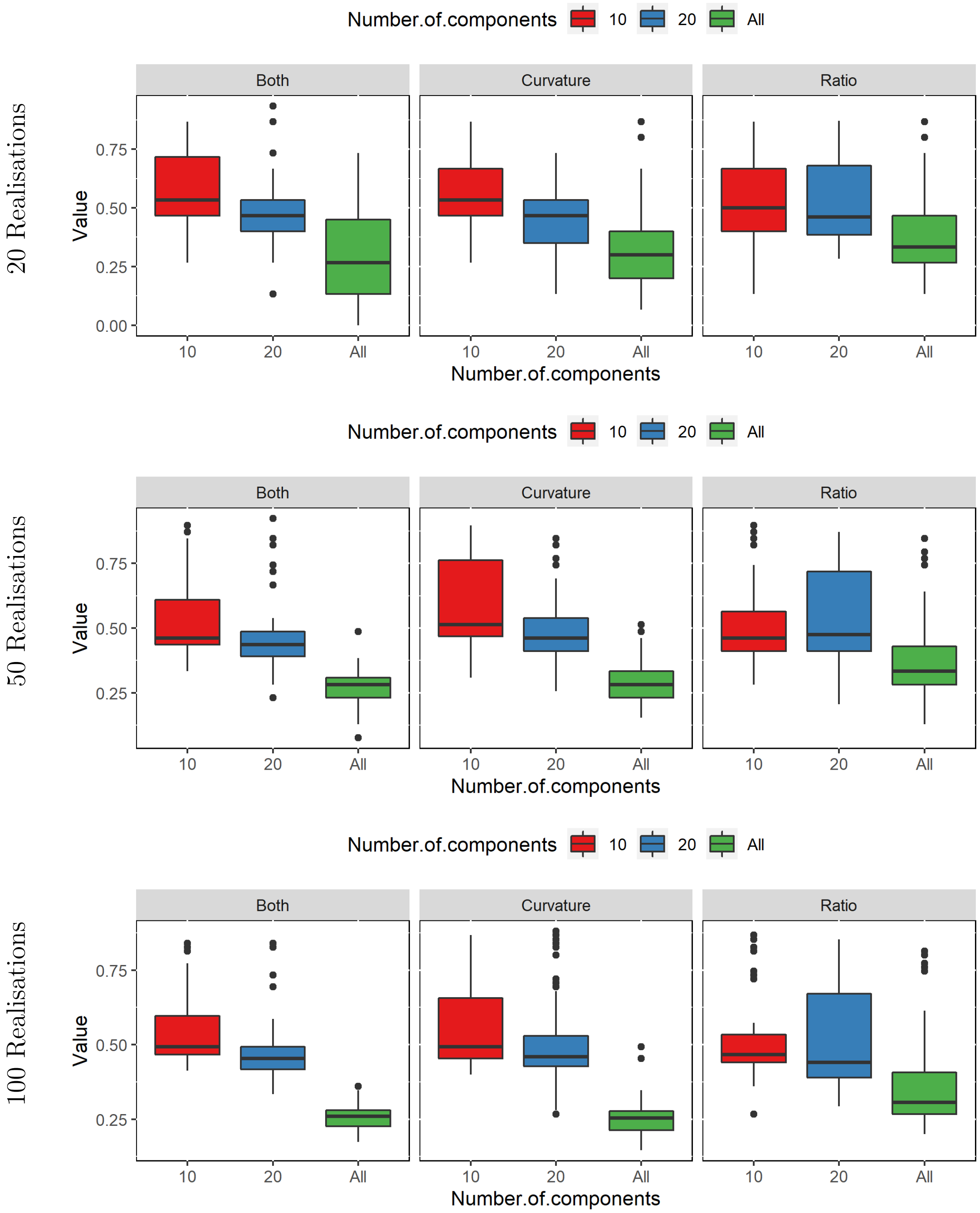


Figure 4.13: Boxplots of misclassification rate for 50 runs of k-means algorithm when considerring samples of 20 (top), 50 (central) and 100 (bottom) realisations using both ratio and curvature, only the curvature and only the ratio for discrimination, respectively. For each setting, misclassification rates for different number of components considered (namely 10, 20 and 'all') are shown. Note that the characteristics were obtained using an osculating disc of radius $r = 3$.

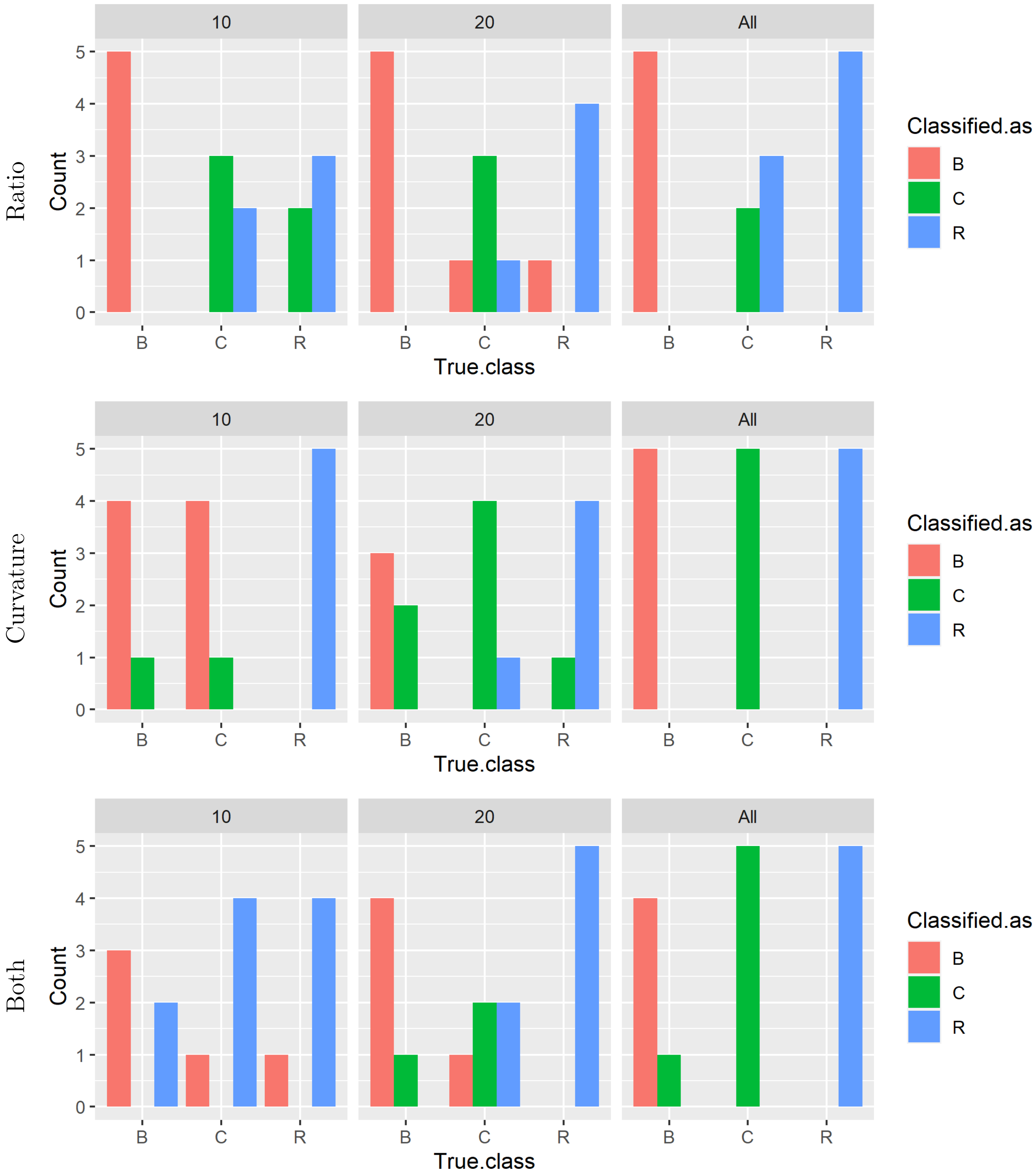


Figure 4.14: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 26.7%, 20% and 20% for 10, 20 and 'all' components, respectively, when using only the ratio, 33.3%, 26.7% and 0% when using only the curvature and 53.3%, 26.7% and 6.7% when using both characteristics for a sample of 20 realisations that were osculated by a disc of radius $r = 5$.

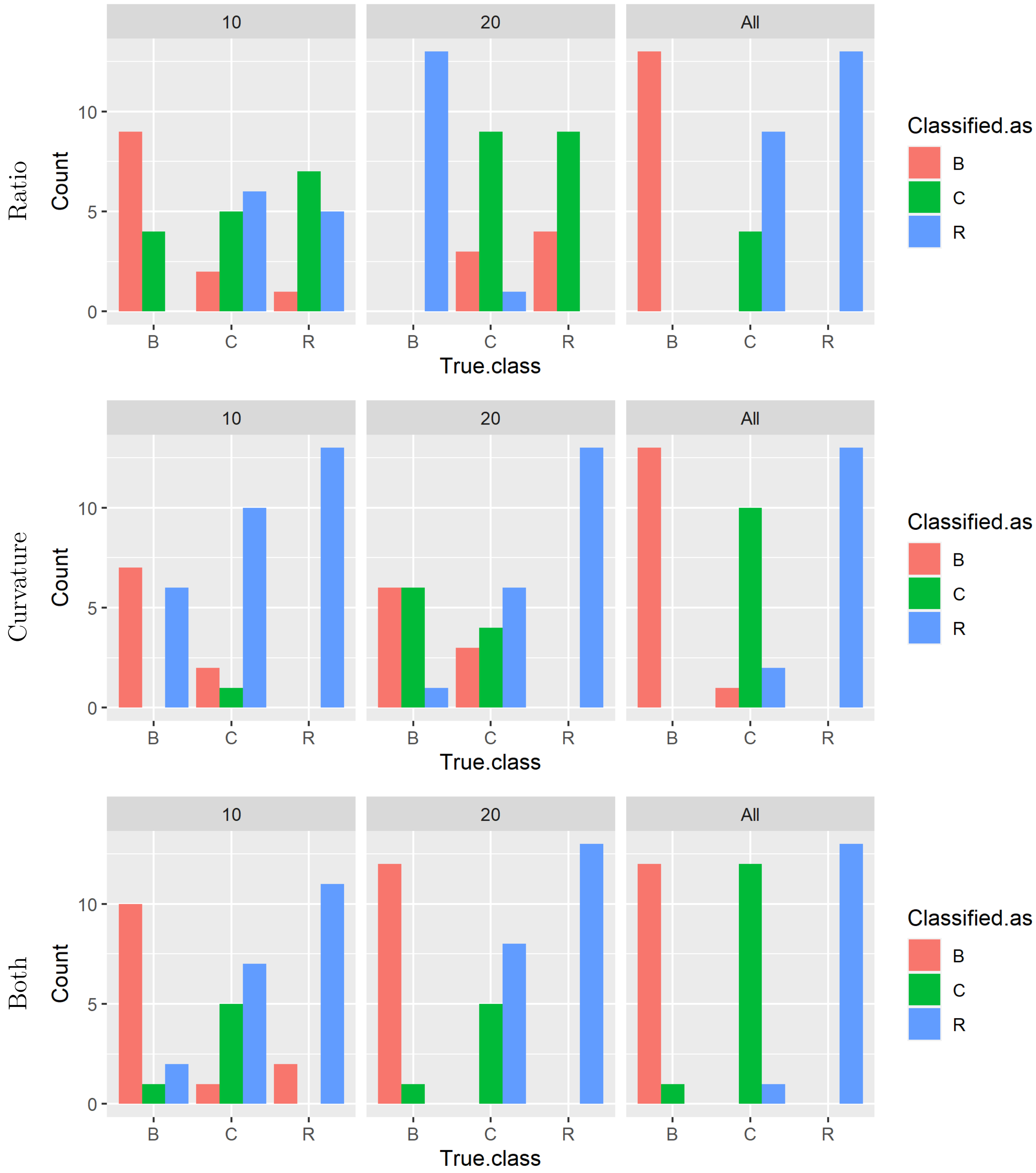


Figure 4.15: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 51.3%, 76.9% and 23.1% for 10, 20 and 'all' components, respectively, when using only the ratio, 46.2%, 41% and 7.7% when using only the curvature, and 33.3%, 23.1% and 5.1% when using both characteristics for a sample of 50 realisations that were osculated by a disc of radius $r = 5$.

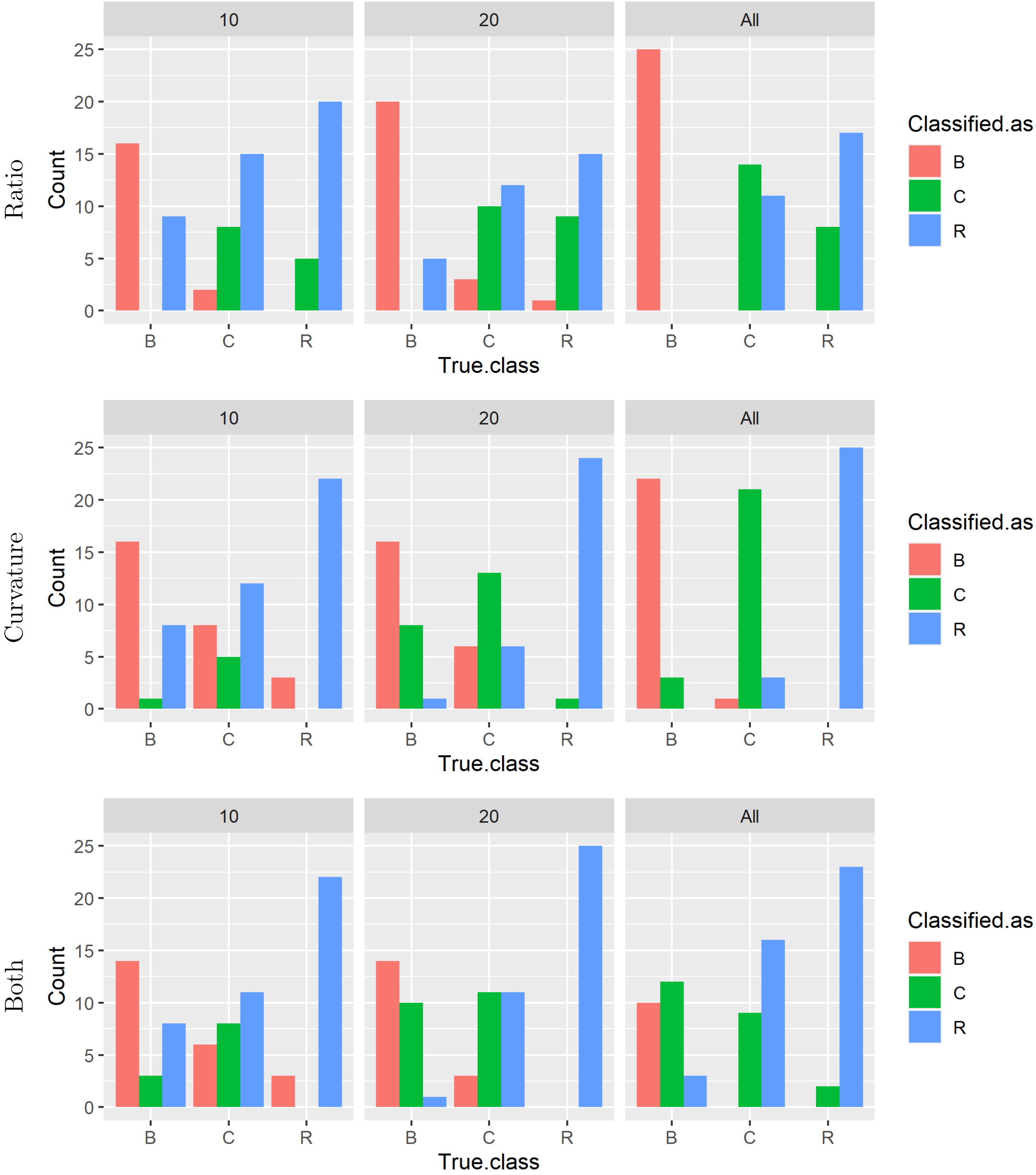


Figure 4.16: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 41.3%, 40% and 25.3% for 10, 20, and 'all' components, respectively, when using only the ratio, 42.7%, 29.3% and 9.3% when using only the curvature and 41.3%, 33.3% and 44% when using both characteristics for a sample of 100 realisations that were osculated by a disc of radius $r = 5$.

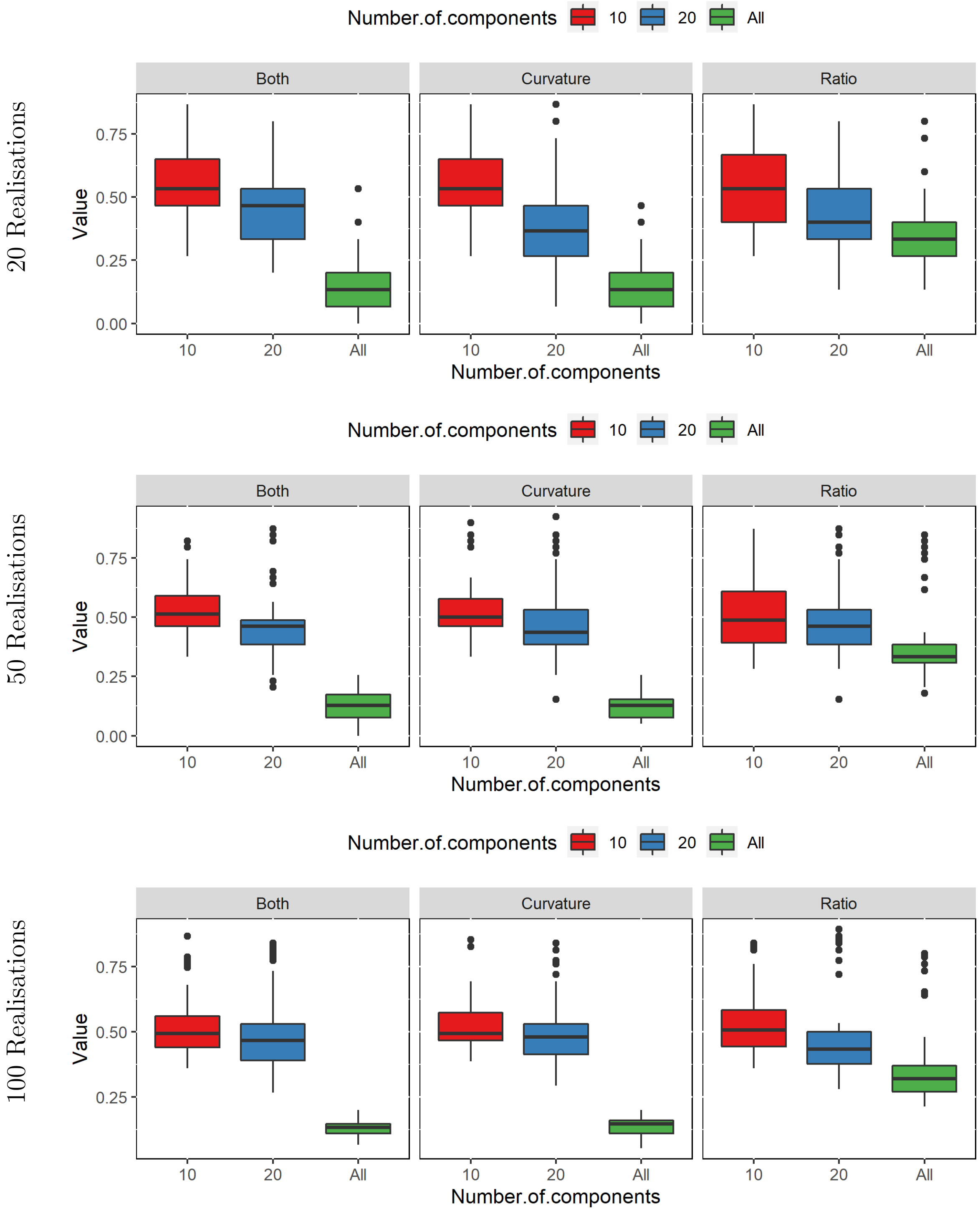


Figure 4.17: Boxplots of misclassification rate for 50 runs of k-means algorithm when considerring samples of 20 (top), 50 (central) and 100 (bottom) realisations using both ratio and curvature, only the curvature and only the ratio for discrimination, respectively. For each setting, misclassification rates for different number of components considered (namely 10, 20, and 'all') are shown. Note that the characteristics were obtained using an osculating disc of radius $r = 5$.

Chapter 5

Application to Real Data

Once we have shown that the classifier is able to distinguish between simulated random processes, we will apply it to the real data. Different types of benign or malignant changes can be indicated by the morphology of the tissue located between the lactiferous duct system and mammary glands. In our study, we will consider two types of mammary tissue - mastopathic (referred to as Masto or 'MP' only from now on) and mammary cancer tissue (referred to as Mamca or 'MC' only). Note that this data has already been studied in [4], [25] and [31]. Samples (in the form of binary images containing 10 sub-samples of size 512×512 representing cross-sections of the duct system), which are used in our study, are shown in Figure 5.1 and Figure 5.2, with black areas representing the aforementioned tissue. The data of mammary cancer and mastopathic tissue were kindly provided by the authors of [4] and [25].

Initially, we identified the components conventionally and then calculated the corresponding curvatures and ratios for both values of r. Since we were provided with only 8 images of size 512×5120 pixels of each tissue, for better learning, we had to augment our data. For mastopathic tissue, we merged the first four images (that is, 'MP1' – 'MP4') together, while for mammary cancer tissues, we merged the first six images (that is, 'MC1' – 'MC6') together and randomly sampled a number of components that roughly corresponds to the number of components in the original images (60 for mastopathy and 300 for mammary cancer). The procedure was repeated 200 times, and in this way we obtained 200 realisations that would be used for the training stage. Similarly, we merged the last two images (that is, 'MP5' and 'MP6' for 'MP', and 'MC7' and 'MC8' for 'MC' tissue) together and sampled the components in the same way as for the training data. This was repeated 50 times, so in the end we had 50 realisations that would be used for the testing phase. The images 'MP7' and 'MP8' were excluded from selection because the results obtained for them in [31] were not satisfactory in the sense that they were assessed as dissimilar to the remaining 'MP' observations.

To assess the classification problem, we follow the same procedure as that used for the simulated data. This means that we will either directly pass the data to the classifier in the supervised classification case, or first calculate means using samples of 10, 20 and 'all' components and pass the calculated means to the classifier in the unsupervised case. The classification is again based on only the ratio, only the curvature or both ratio and curvature together.

5.1 Supervised Classification

As already mentioned, we follow the same procedure as for the simulated data. This means that the data are divided into train set and test set with a 3:1 ratio (which means that 75% of the realisations is used for training, while 25% is used for testing the performance of the classifier). Since we wanted to test how fast the classifier learns and how much the amount of data at our disposal affects its performance, we again use three settings:

- in the first setting we used a sample of 20 randomly chosen realisations from each type of mammary tissue (further 'class'), mastopathic (class 'MP') and cancerous (class 'MC'), meaning that in the training set we have 30 realisations (15 of each class, 'MP' and 'MC') in the training set and 10 realisations for testing purpose (5 of each class)
- in the second setting we used a sample of 50 randomly chosen realisations from each class, meaning that in the training set we have 74 realisations (37 of each class) in the training set and 26 realisations for testing purpose (13 of each class)
- in the third setting we used a sample of 100 randomly chosen realisations from each class, meaning that we have 150 realisations (75 of each class) in the training set and 50 realisations for testing purpose (25 of each class).

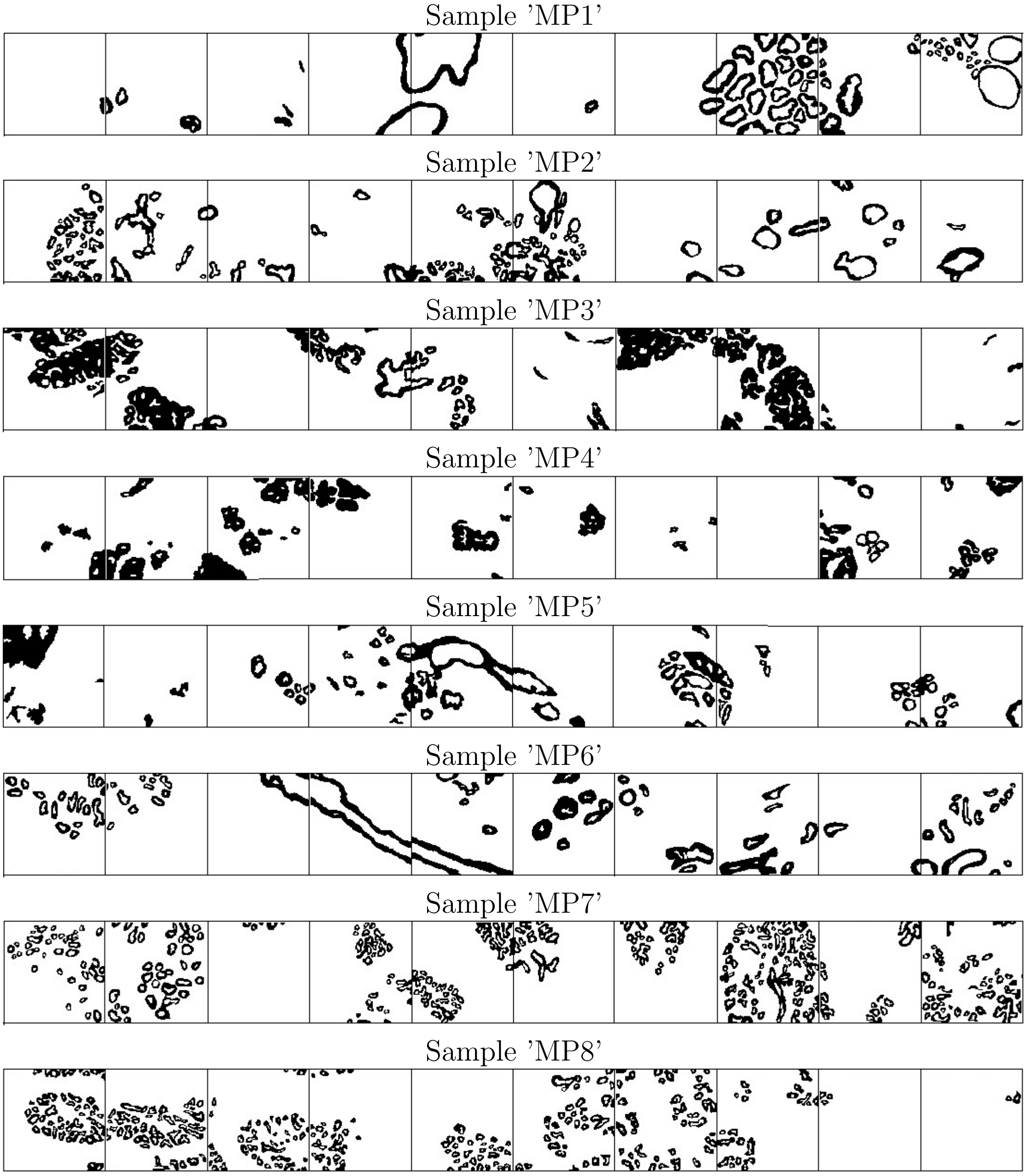
Each of the above-mentioned settings is then split into three subsettings according to the characteristic which is used for discrimination, namely 'ratio', 'curvature', and 'both'. After that, the classifier is learnt three times for different numbers of components which we use for calculating the $\mathcal{N}$-distance (i.e. 10, 20 and 'all'). After the learning stage, we use the test set and predict the labels using the posterior probabilities calculated for each class. The classification results for each setting are shown in Figures 5.3 (20 realisations), 5.4 (50 realisations) and 5.5 (100 realisations) for the data obtained using the osculating circle with radius $r = 3$, and Figures 5.7 (20 realisations), 5.8 (50 realisations) and 5.9 (100 realisations) for the data obtained using the osculating circle with radius $r = 5$, respectively. We can see that after the initial run, the classification precision follows the pattern observed for the simulated data – it increases with the growing sample size (i.e., it is the lowest when only 10 components are used and the highest when 'all' components are used) for all settings (i.e., for different number of realisations considered) in all cases (i.e., for data obtained with an osculating circle of radius $r = 3$ and $r = 5$, respectively). After the initial run, we repeat the procedure 50 times to obtain box plots of misclassification rates. The results are shown in Figure 5.6 for the data obtained using $r = 3$ and Figure 5.10 for the data obtained using $r = 5$. The minimum and maximum misclassification rates are shown in Table 5.1 for data obtained using $r = 3$ and Table 5.2 for data obtained using $r = 5$, respectively. We can see that the values reflect the ones in the initial run, meaning that the classification is most precise when using 'all' components in all settings.

Number of realisations	20			50			100		
Number of components	10	20	'all'	10	20	'all'	10	20	'all'
Characteristics considered	Misclassification rate [%]								
Both	30	10	0	7.7	0	0	4	0	0
Curvature	20	10	0	7.7	0	0	6	0	0
Ratio	40	20	0	23.1	11.5	0	18	10	0
Both	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$
Curvature	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$
Ratio	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{4}$	$\underline{0}$	$\underline{0}$

Table 5.1: Maximum and minimum (underlined) misclassification rates obtained after 50 runs of k-nearest neighbours algorithm for different settings (20, 50 and 100 realisations) and respective subsettings (Both, Curvature and Ratio) when using samples of 10, 20 and 'all' components, respectively. Note that the data used are the data obtained using an osculating circle of radius $r = 3$.

Number of realisations	20			50			100		
Number of components	10	20	'all'	10	20	'all'	10	20	'all'
Characteristics considered	Misclassification rate [%]								
Both	20	0	0	7.7	3.8	0	6	0	0
Curvature	30	0	0	11.5	0	0	6	2	0
Ratio	40	20	0	19.3	7.7	0	18	6	0
Both	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$
Curvature	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$
Ratio	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{0}$

Table 5.2: Maximum and minimum (underlined) misclassification rates obtained after 50 runs of k-nearest neighbours algorithm for different settings (20, 50 and 100 realisations) and respective subsettings (Both, Curvature and Ratio) when using samples of 10, 20 and 'all' components, respectively. Note that the data used are the data obtained using an osculating circle of radius $r = 5$.



Sample 'MP1'

Sample 'MP2'

Sample 'MP3'

Sample 'MP4'

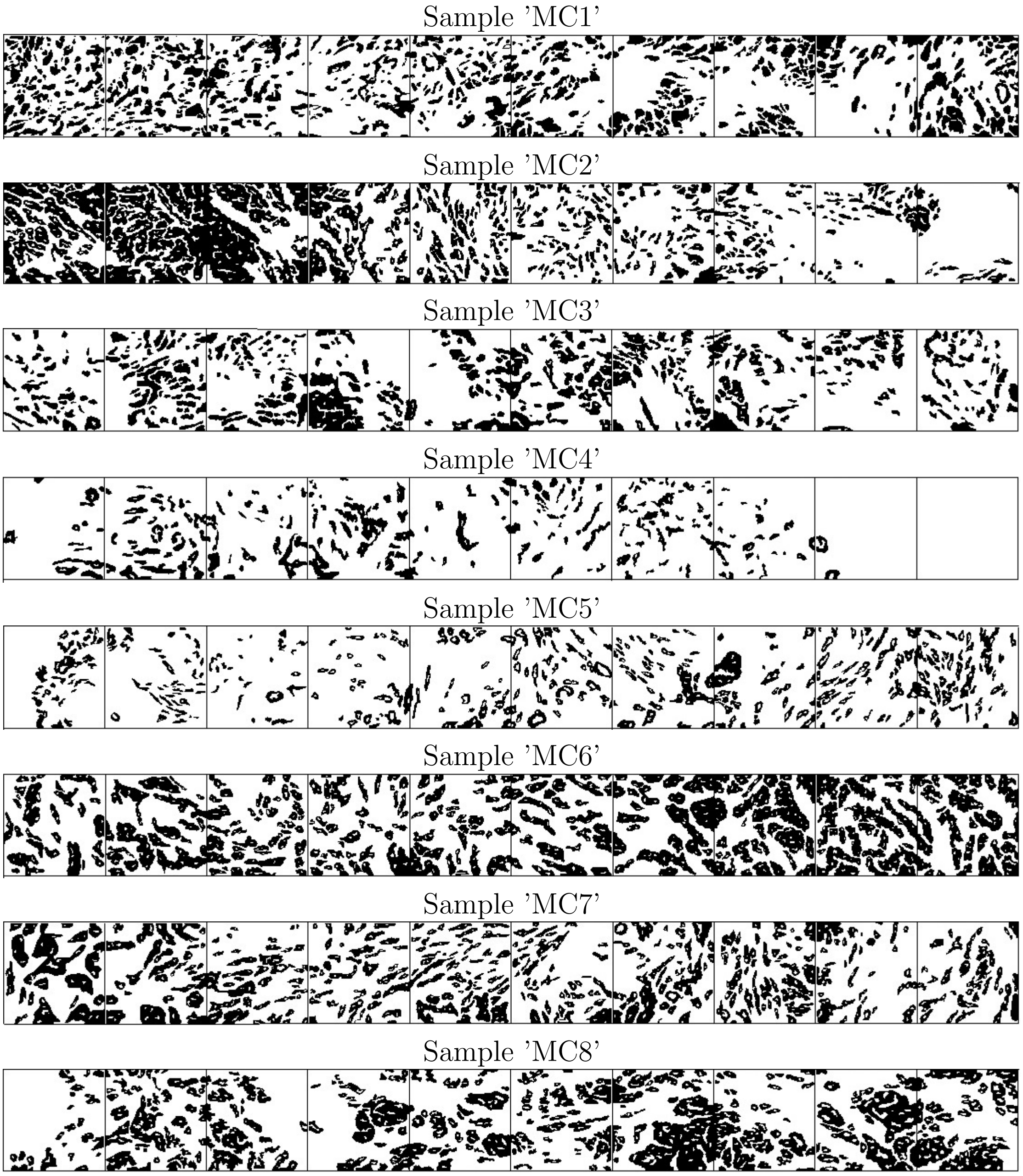
Sample 'MP5'

Sample 'MP6'

Sample 'MP7'

Sample 'MP8'

Figure 5.1: Samples of mastopathic breast tissue [4], [25]



Sample 'MC1'

Sample 'MC2'

Sample 'MC3'

Sample 'MC4'

Sample 'MC5'

Sample 'MC6'

Sample 'MC7'

Sample 'MC8'

Figure 5.2: Samples of mammary cancer [4], [25]

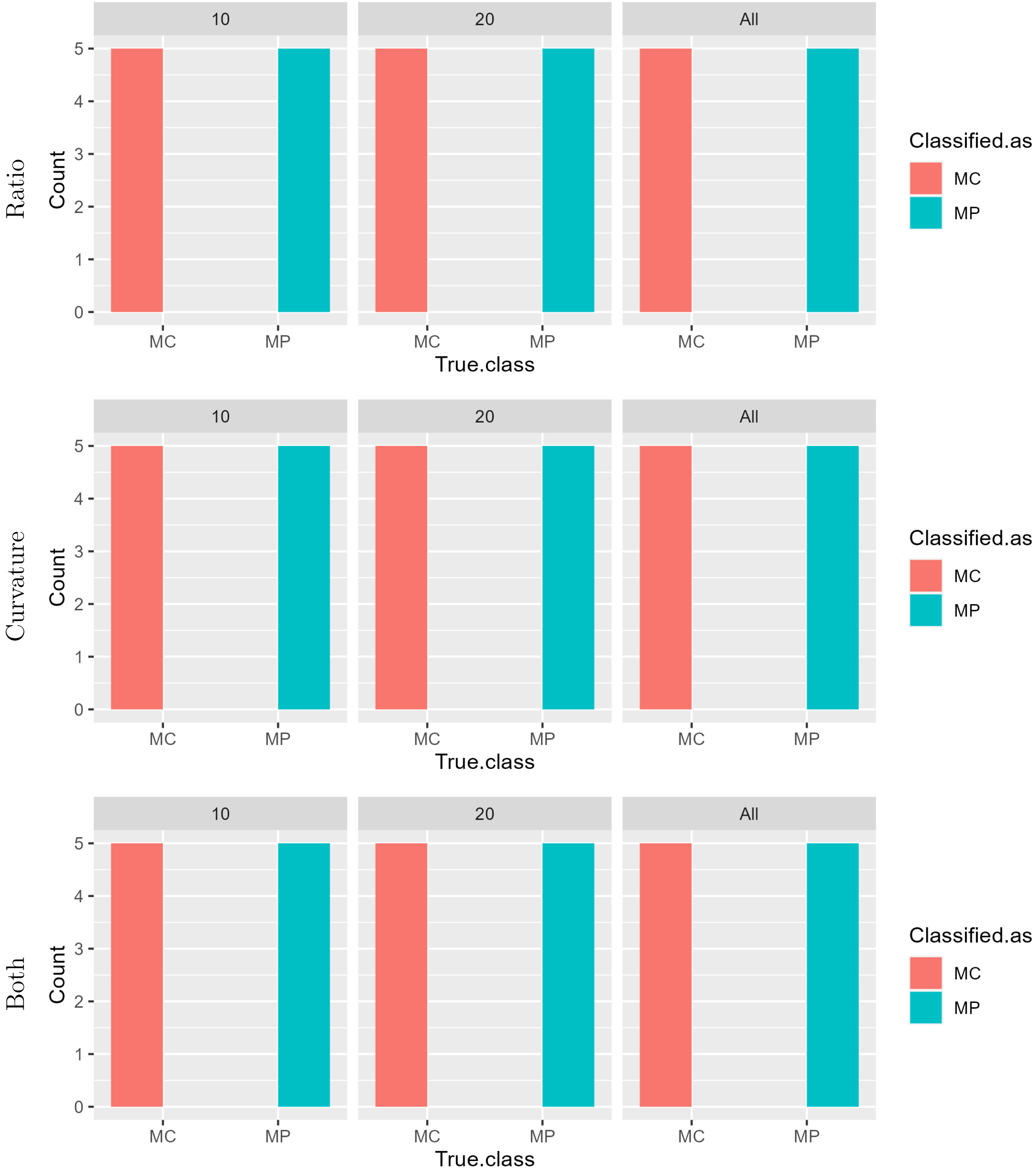


Figure 5.3: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 0%, 0% and 0% for 10, 20 and 'all' components, respectively, when using only the ratio, 0%, 0% and 0% when using only the curvature and 0%, 0% and 0% when using both characteristics for a sample of 20 realisations that were osculated by a disc of radius $r = 3$.

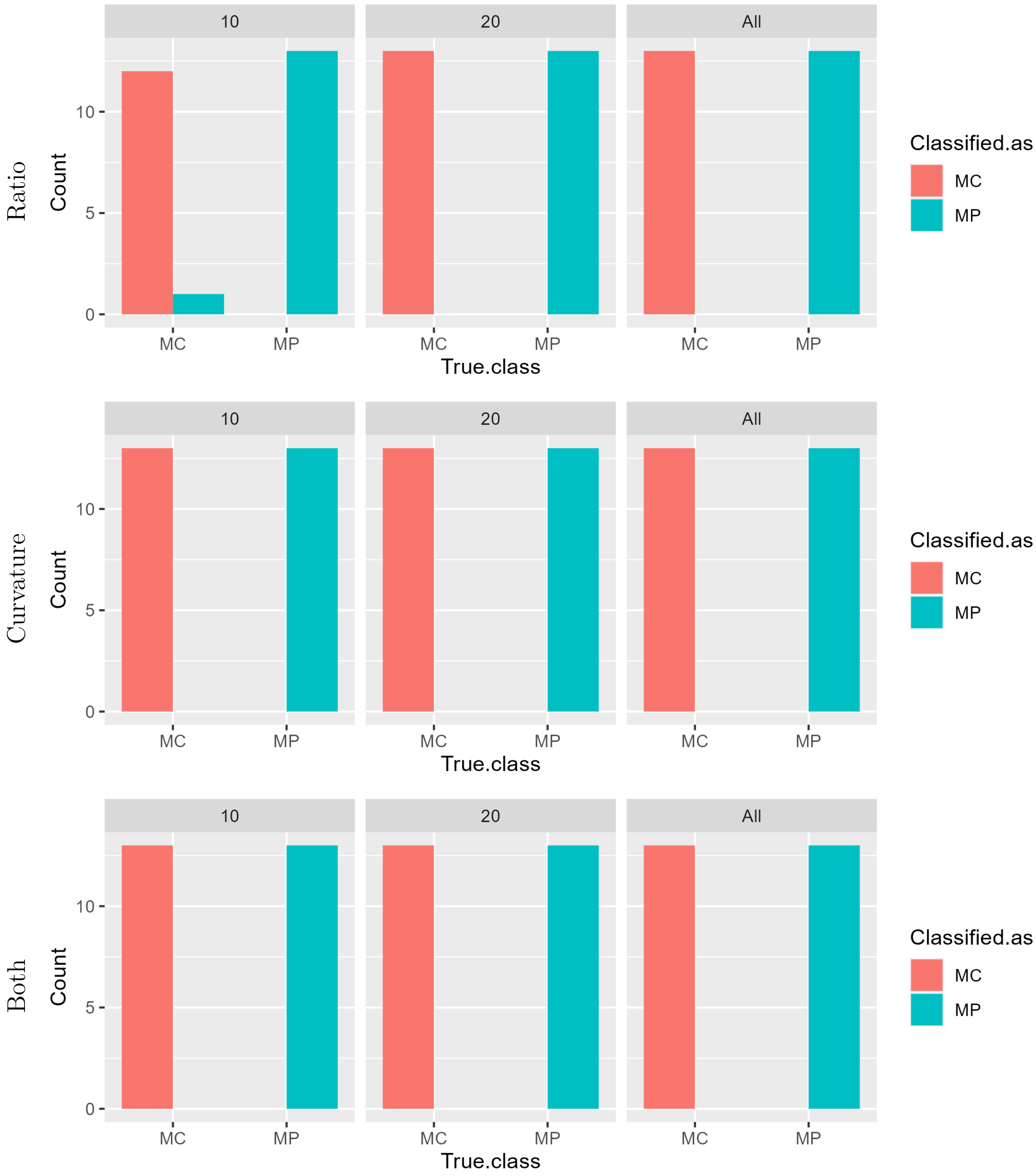


Figure 5.4: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 4%, 0% and 0% for 10, 20 and 'all' components, respectively, when using only the ratio, 0%, 0% and 0% when using only the curvature and 0%, 0% and 0% when using both characteristics for a sample of 50 realisations that were osculated by a disc of radius $r = 3$.

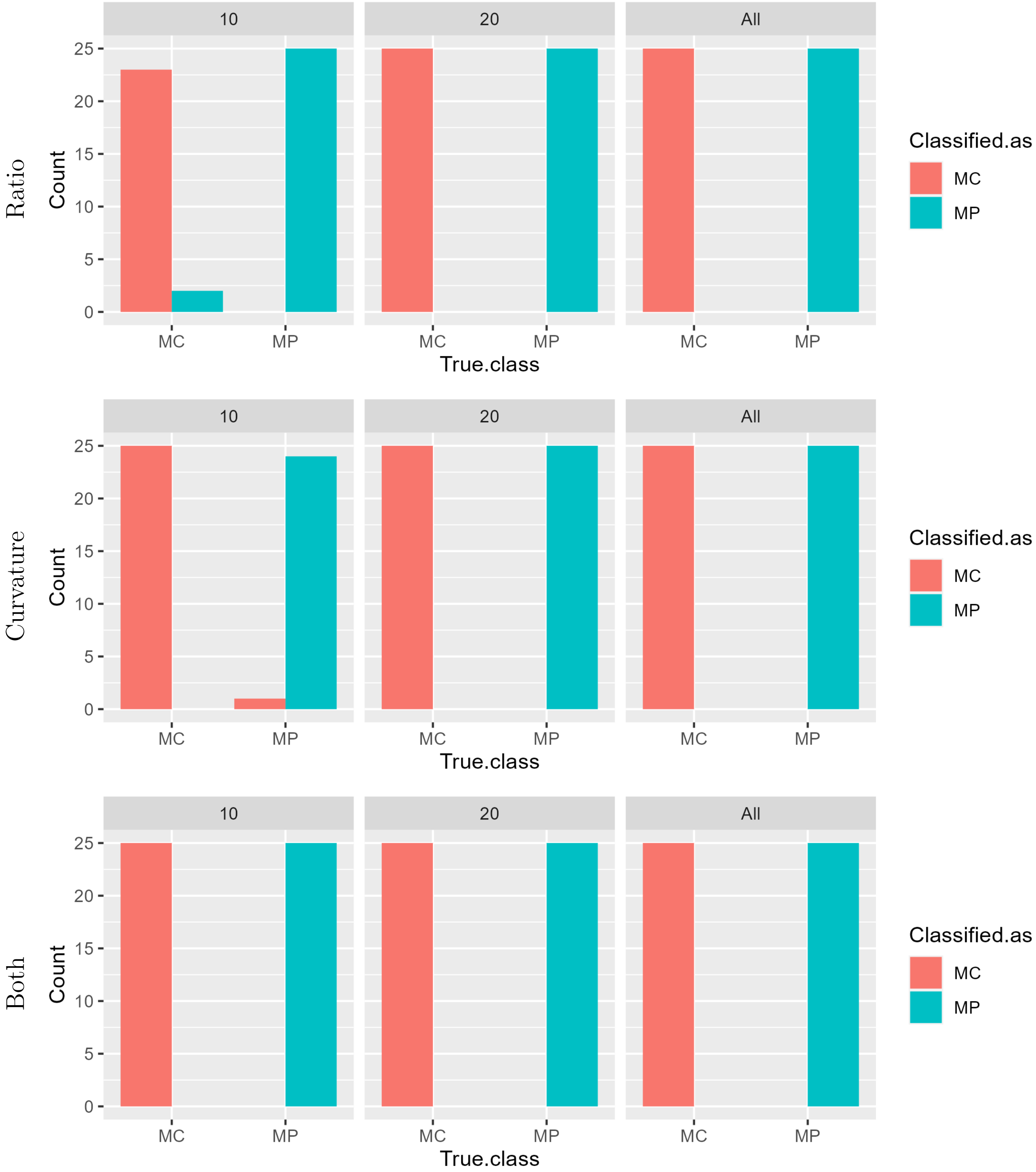


Figure 5.5: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 4%, 0% and 0% for 10, 20 and 'all' components, respectively, when using only the ratio, 2%, 0% and 0% when using only the curvature and 0%, 0% and 0% when using both characteristics for a sample of 100 realisations that were osculated by a disc of radius $r = 3$.

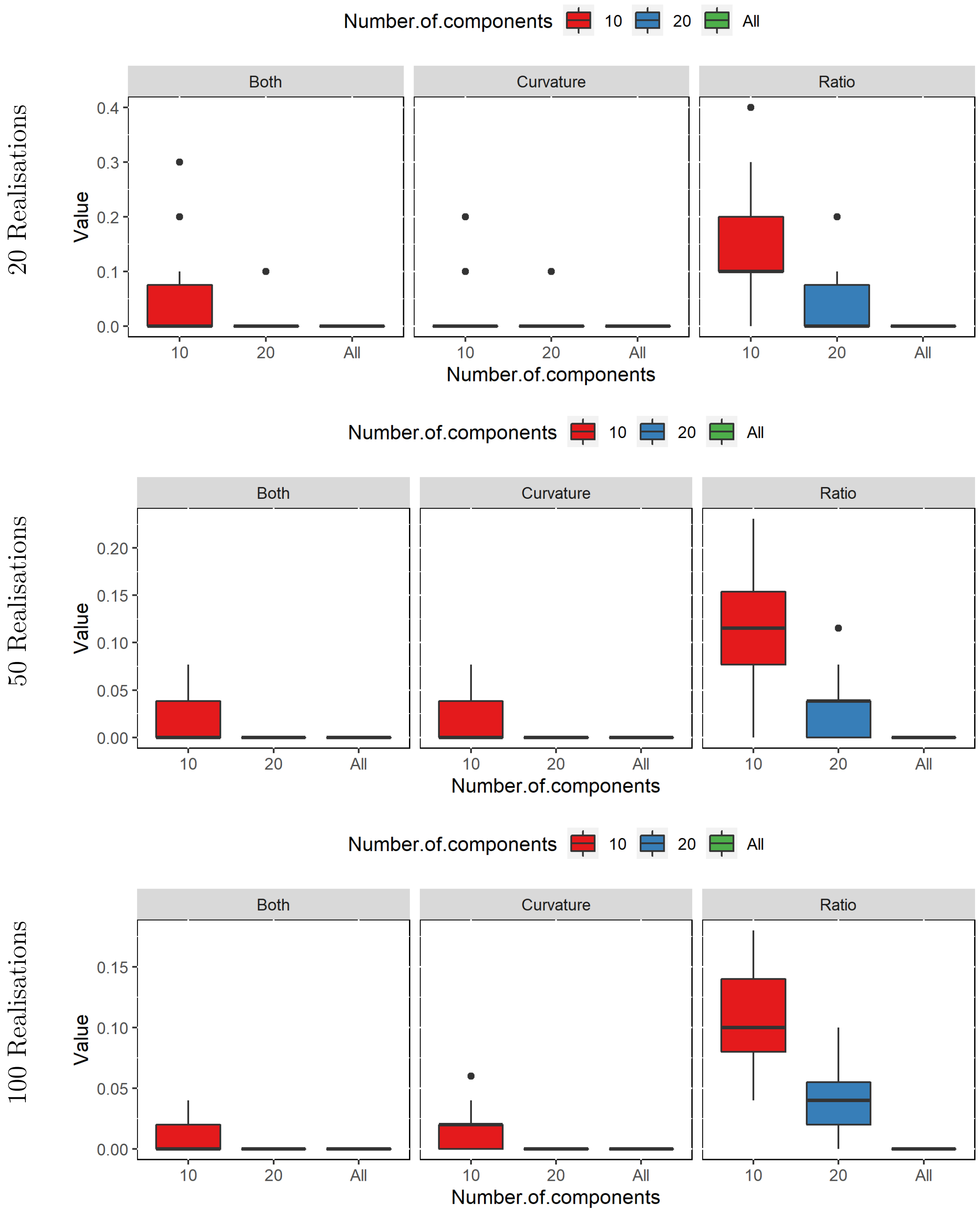


Figure 5.6: Boxplots of misclassification rate for 50 runs of k-nearest neighbours algorithm when considering samples of 20 (top), 50 (central) and 100 (bottom) realisations using both ratio and curvature, only the curvature and only the ratio for discrimination, respectively. For each setting, misclassification rates for different number of components considered (namely 10, 20, and 'all') are shown. Note that the characteristics were obtained using an osculating disc of radius $r = 3$.

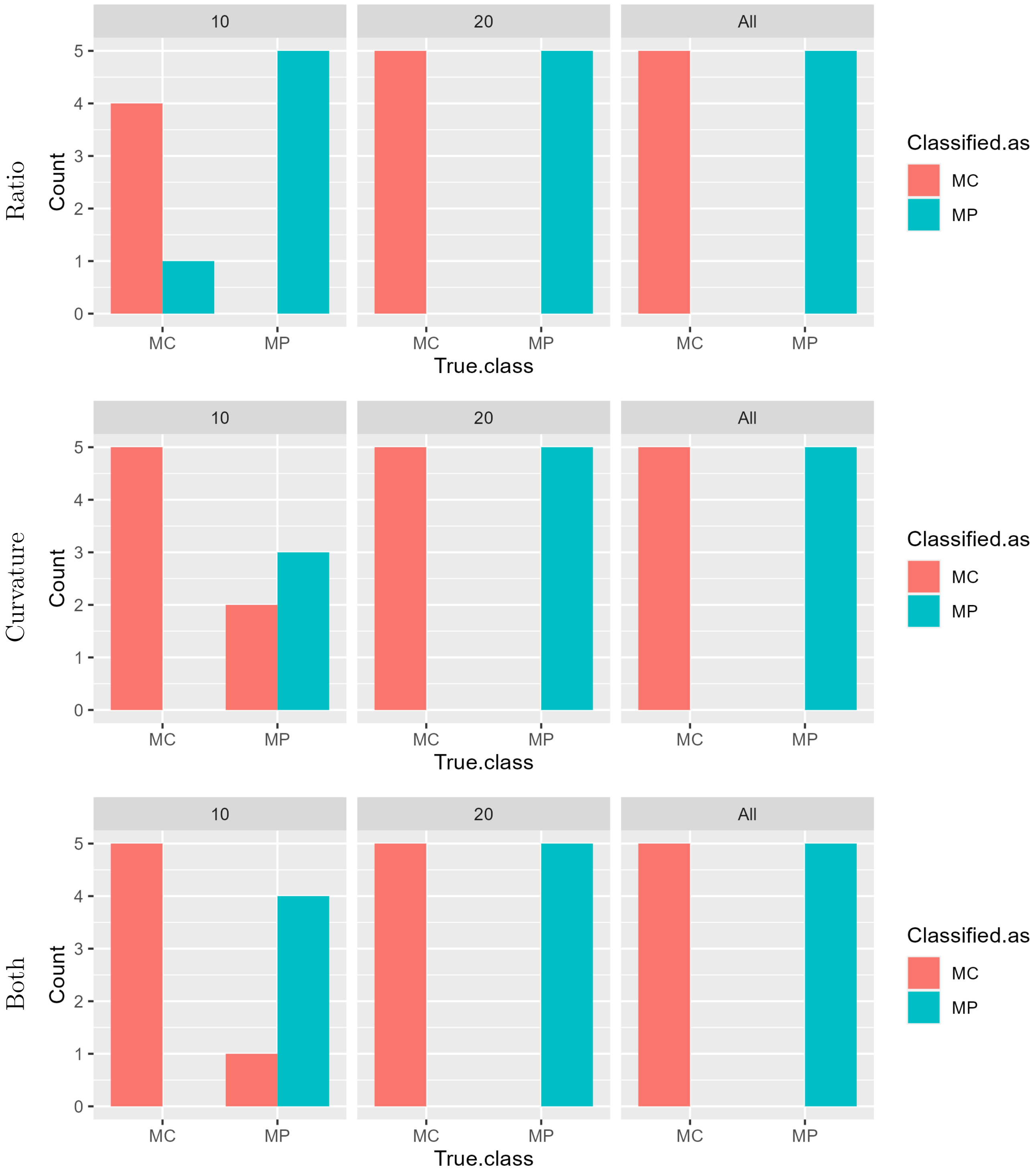


Figure 5.7: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 10%, 20% and 0% for 10, 20 and 'all' components, respectively, when using only the ratio, 0%, 0% and 0% when using only the curvature and 0%, 0% and 0% when using both characteristics for a sample of 20 realisations that were osculated by a disc of radius $r = 5$.

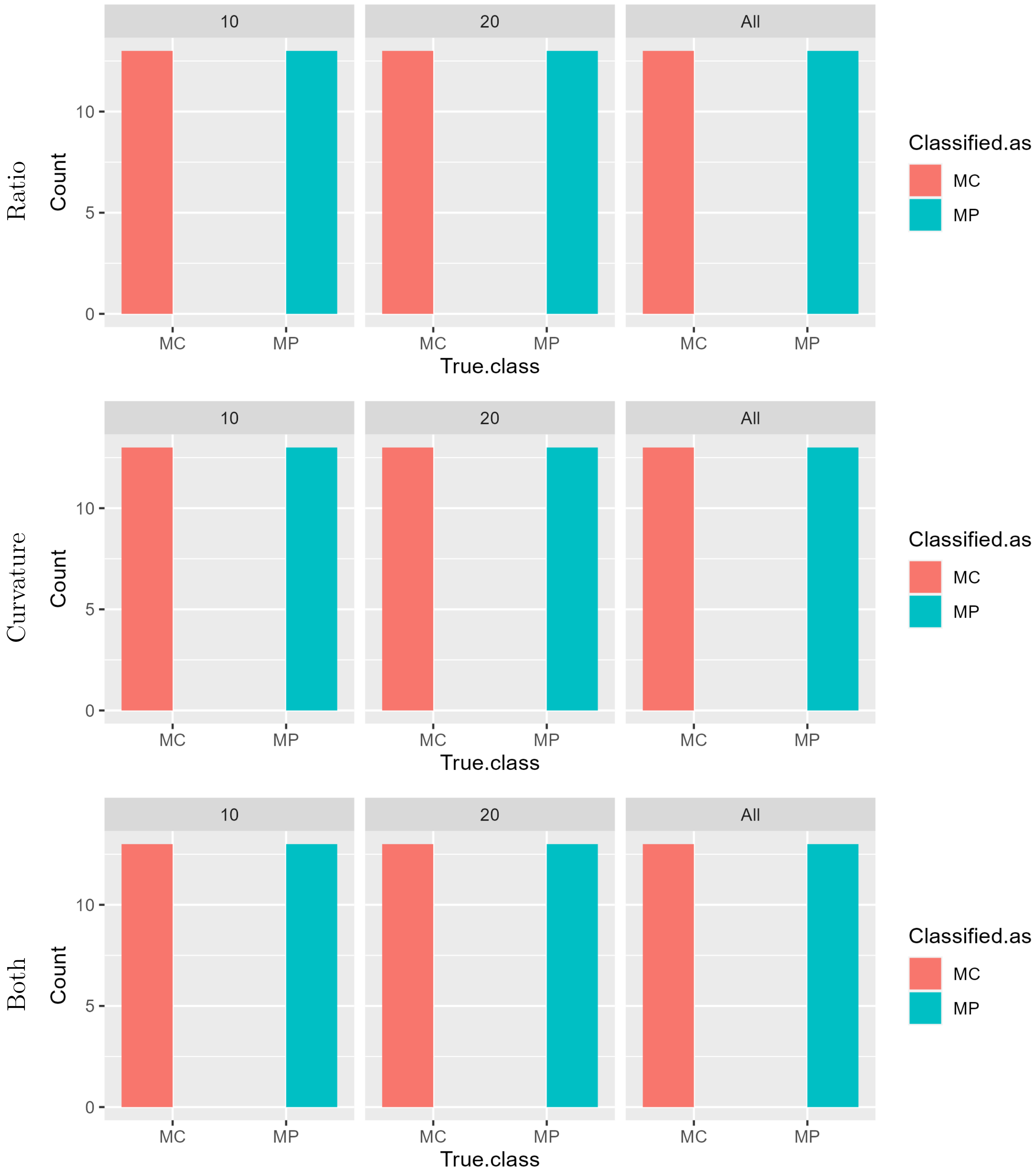


Figure 5.8: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 0%, 0% and 0% for 10, 20 and 'all' components, respectively, when using only the ratio, 0%, 0% and 0% when using only the curvature and 0%, 0% and 0% when using both characteristics for a sample of 50 realisations that were osculated by a disc of radius $r = 5$.

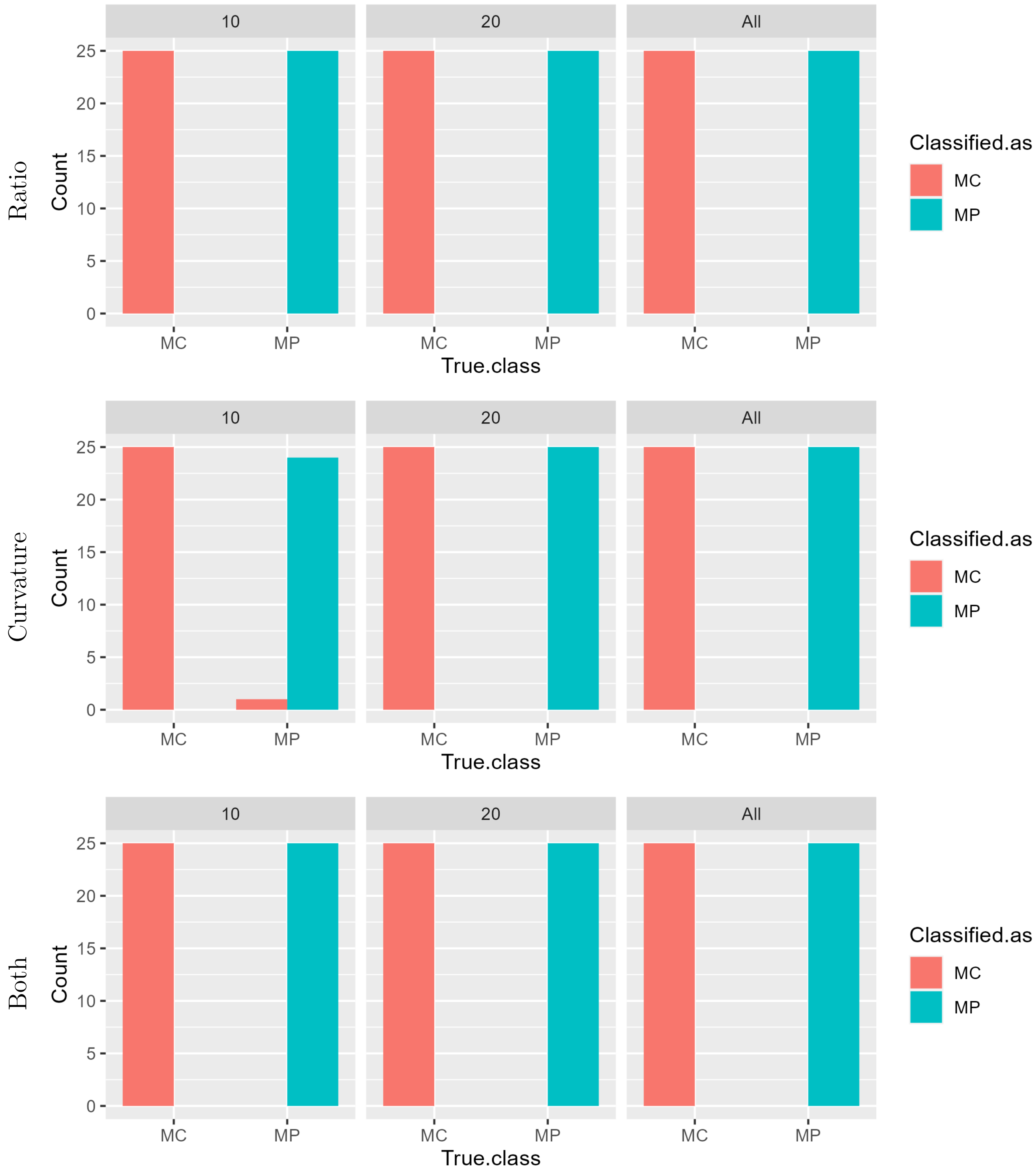


Figure 5.9: Histograms of k-nearest neighbours classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 8%, 2% and 0% for 10, 20 and 'all' components, respectively, when using only the ratio, 0%, 0% and 0% when using only the curvature and 0%, 0% and 0% when using both characteristics for a sample of 100 realisations that were osculated by a disc of radius $r = 5$.

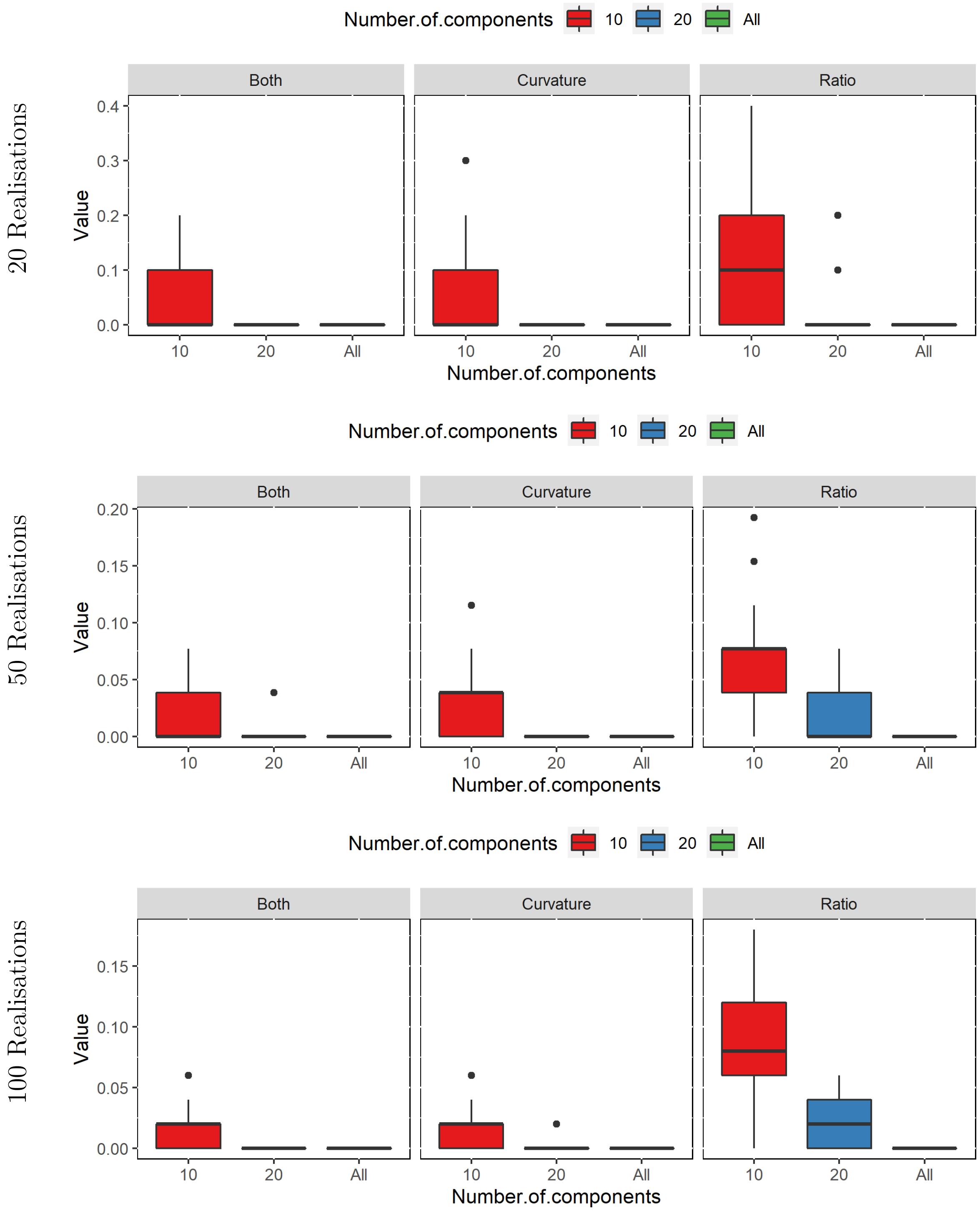


Figure 5.10: Boxplots of misclassification rate for 50 runs of k-nearest neighbours algorithm when considering samples of 20 (top), 50 (central) and 100 (bottom) realisations using both ratio and curvature, only the curvature and only the ratio for discrimination, respectively. For each setting, misclassification rates for different number of components considered (namely 10, 20, and 'all') are shown. Note that the characteristics were obtained using an osculating disc of radius $r = 5$.

5.2 Unsupervised Classification

For each newly obtained realisation (using the data augmentation procedure, as described above), we calculate the means by randomly choosing a sample of components of sizes 10, 20 and ’all’. After that, the data are passed to the classifier. The results are shown in Figures 5.11 (20 realisations), 5.12 (50 realisations), and 5.13 (100 realisation) for the data obtained using an osculating circle of size $r = 3$, and Figures 5.15 (20 realisations), 5.16 (50 realisations), and 5.17 (100 realisations) for the data obtained using a circle with radius $r = 5$, respectively. We can see that after the initial run, the classification precision reflects the results observed for the simulated data – it increases with the growing sample size (i.e., it is the lowest when only 10 components are used and the highest when ’all’ components are used) for all settings (i.e., for different number of realisations considered) for both the data obtained with an osculating circle of radius $r = 3$ and the data obtained using a circle of radius $r = 5$. After the initial run, the procedure is repeated 50 times for different settings. The resulting box plots are shown in Figure 5.14, for the data obtained using the osculating circle with $r = 3$ and Figure 5.18, for the data obtained using the circle with $r = 5$. The minimum and maximum misclassification rates are shown in Table 5.3 for data obtained using $r = 3$ and Table 5.4 for data obtained using $r = 5$, respectively. We can see that the values reflect those in the initial run, meaning that the classification is most precise when using ’all’ components in all settings, whereas the increasing number of realisations does not make the classifier significantly more precise.

Number of realisations	20			50			100		
Number of components	10	20	’all’	10	20	’all’	10	20	’all’
Characteristics considered	Misclassification rate [%]								
Both	60	60	10	65.4	42.3	7.7	40	32	6
Curvature	70	60	10	69.2	38.5	3.8	46	36	2
Ratio	80	100	100	84.6	80.8	23.1	64	76	16
Both	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{11.5}$	$\underline{3.8}$	$\underline{0}$	$\underline{14}$	$\underline{4}$	$\underline{0}$
Curvature	$\underline{0}$	$\underline{0}$	$\underline{0}$	$\underline{3.8}$	$\underline{3.8}$	$\underline{0}$	$\underline{18}$	$\underline{8}$	$\underline{0}$
Ratio	$\underline{10}$	$\underline{10}$	$\underline{0}$	$\underline{30.8}$	$\underline{15.4}$	$\underline{0}$	$\underline{38}$	$\underline{22}$	$\underline{6}$

Table 5.3: Maximum and minimum (underlined) misclassification rates obtained after 50 runs of k-means algorithm for different settings (20, 50 and 100 realisations) and respective subsettings (Both, Curvature and Ratio) when using samples of 10, 20 and ’all’ components, respectively. Note that the data used are the data obtained using an osculating circle of radius $r = 3$.

Number of realisations	20			50			100		
Number of components	10	20	'all'	10	20	'all'	10	20	'all'
Characteristics considered	Misclassification rate [%]								
Both	80	50	20	65.4	38.5	7.7	64	26	6
Curvature	60	40	10	50	30.8	3.8	44	26	2
Ratio	90	100	40	73.1	80.8	15.4	68	76	20
Both	0	0	0	7.7	3.8	0	12	10	0
Curvature	10	0	0	7.7	3.8	0	14	6	0
Ratio	10	10	0	30.8	15.4	0	38	16	4

Table 5.4: Maximum and minimum (underlined) misclassification rates obtained after 50 runs of k-means algorithm for different settings (20, 50 and 100 realisations) and respective subsettings (Both, Curvature and Ratio) when using samples of 10, 20 and 'all' components, respectively. Note that the data used are the data obtained using an osculating circle of radius $r = 5$.

.

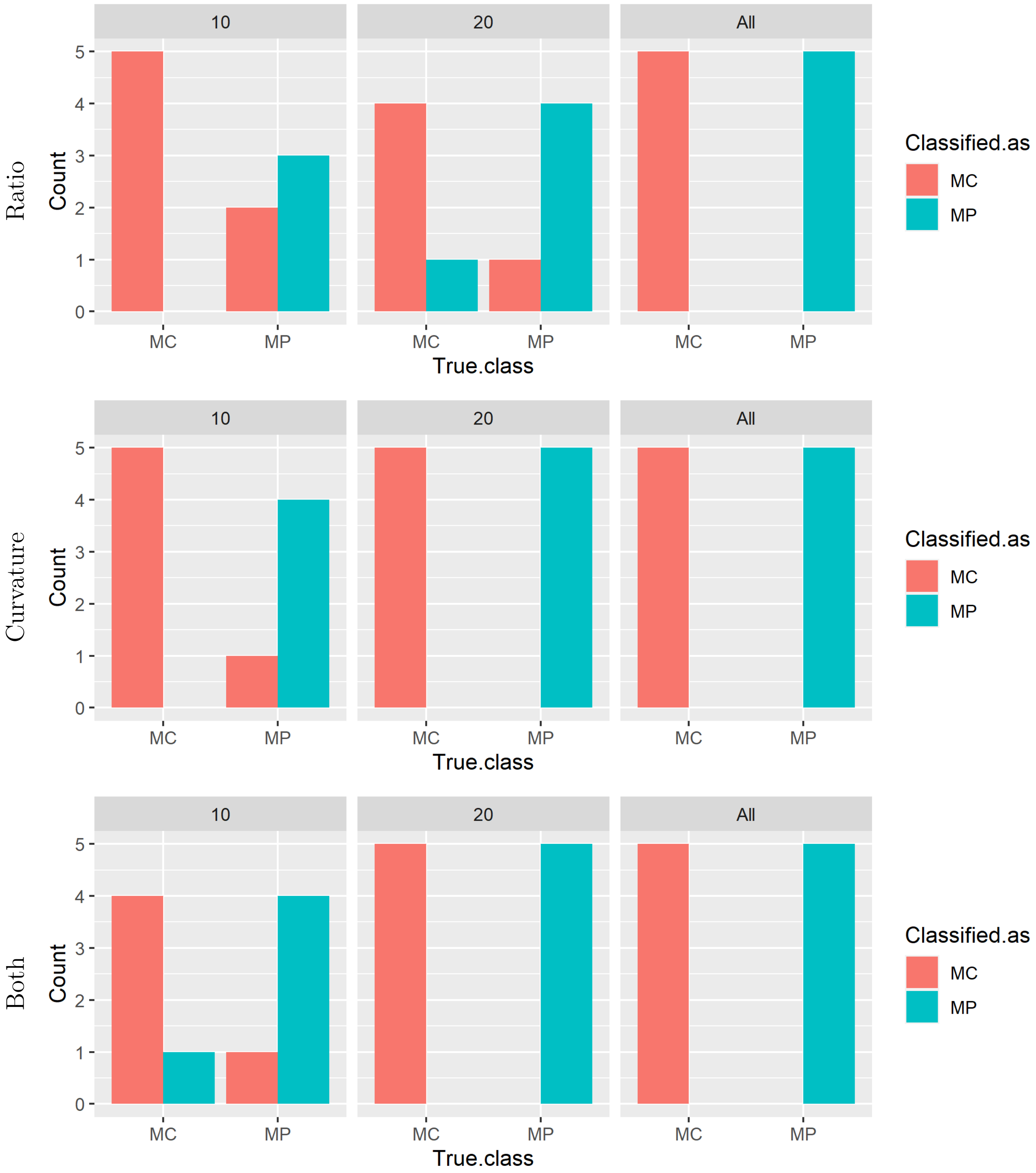


Figure 5.11: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 20%, 20% and 0% for 10, 20 and 'all' components, respectively, when using only the ratio, 10%, 0% and 0% when using only the curvature and 20%, 0% and 0% when using both characteristics for a sample of 20 realisations that were osculated by a disc of radius $r = 3$.

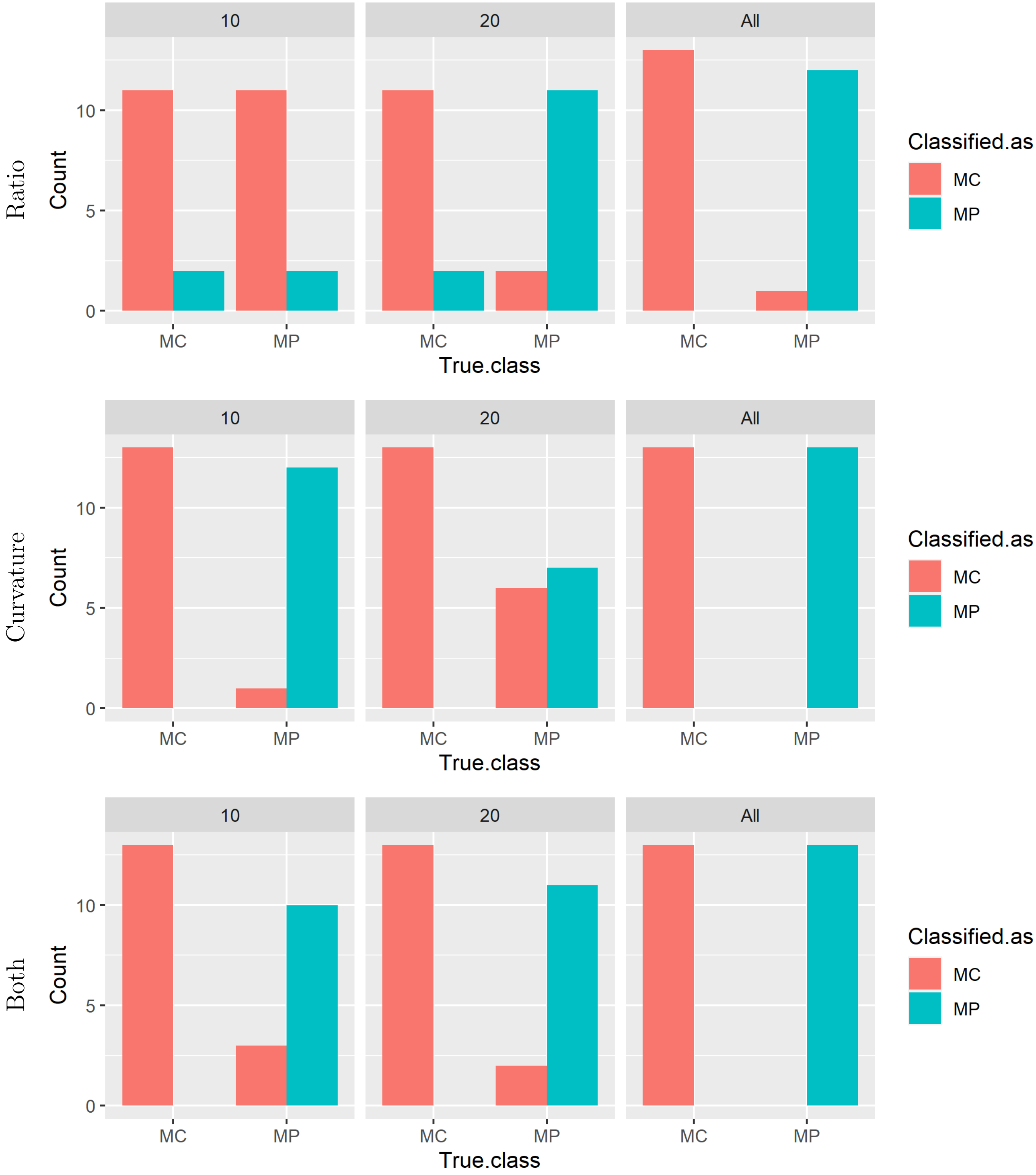


Figure 5.12: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 50%, 15.4% and 3.8% for 10, 20 and 'all' components respectively when using only the ratio, 3.8%, 23.1% and 0% when using only the curvature and 11.5%, 7.7% and 0% when using both characteristics for a sample of 50 realisations that were osculated by a disc of radius $r = 3$.

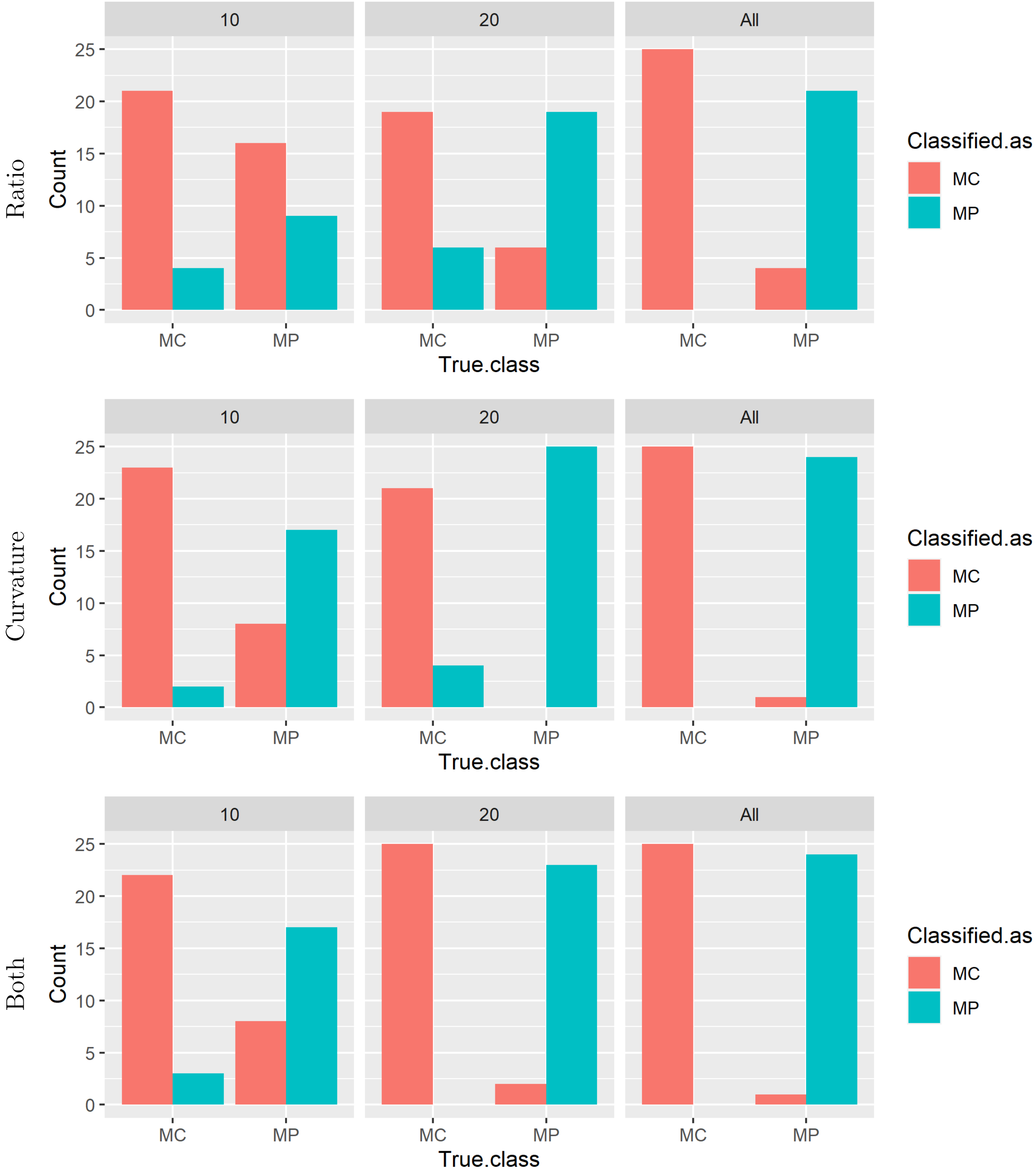


Figure 5.13: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 40%, 24% and 8% for 10, 20 and 'all' components, respectively, when using only the ratio, 20%, 8% and 2% when using only the curvature and 22%, 4% and 2% when using both characteristics for a sample of 100 realisations that were osculated by a disc of radius $r = 3$.

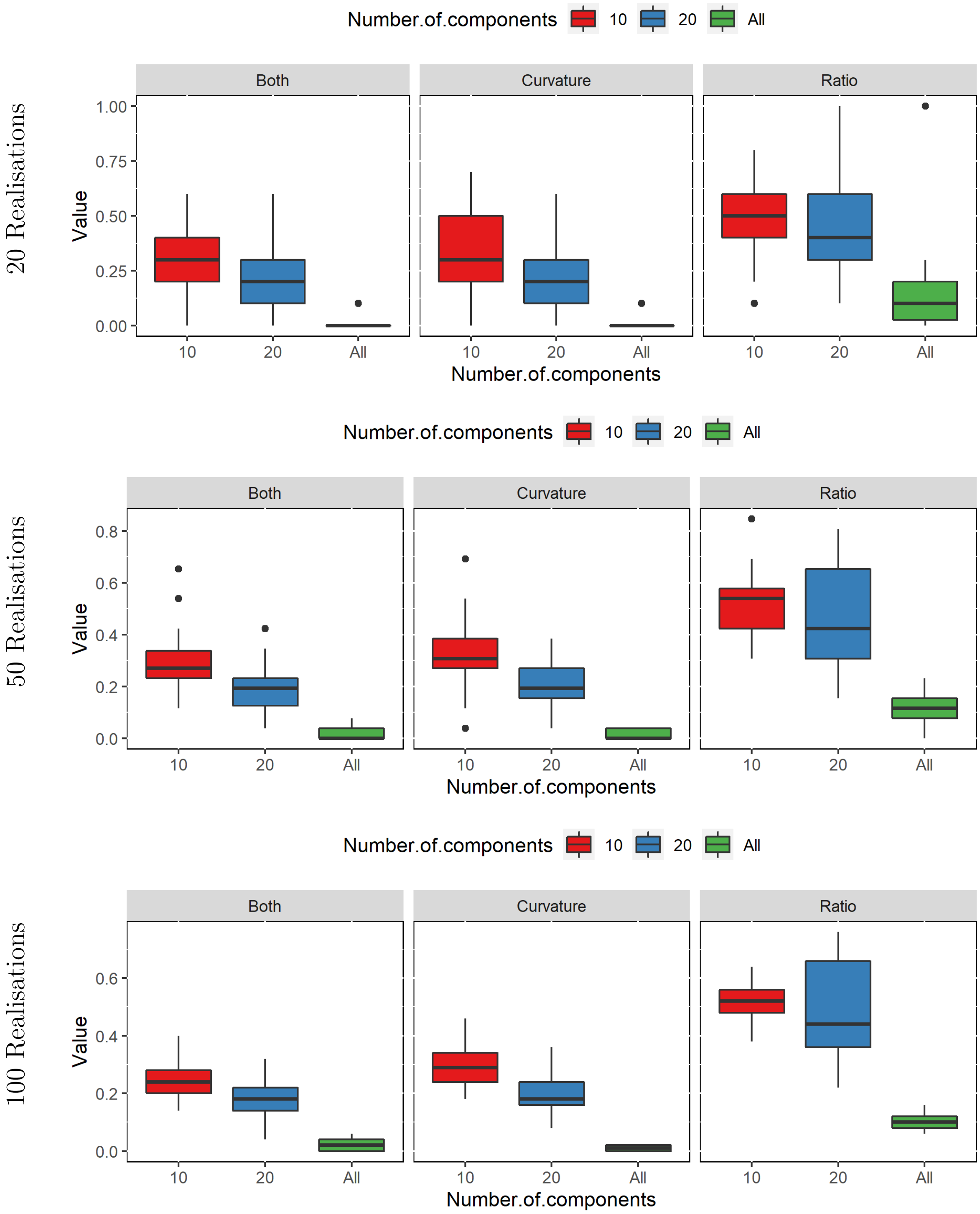


Figure 5.14: Boxplots of misclassification rate for 50 runs of k-means algorithm when considerring samples of 20 (top), 50 (central) and 100 (bottom) realisations using both ratio and curvature, only the curvature and only the ratio for discrimination, respectively. For each setting, misclassification rates for different number of components considered (namely 10, 20 and 'all') are shown. Note that the characteristics were obtained using an osculating disc of radius $r = 3$.

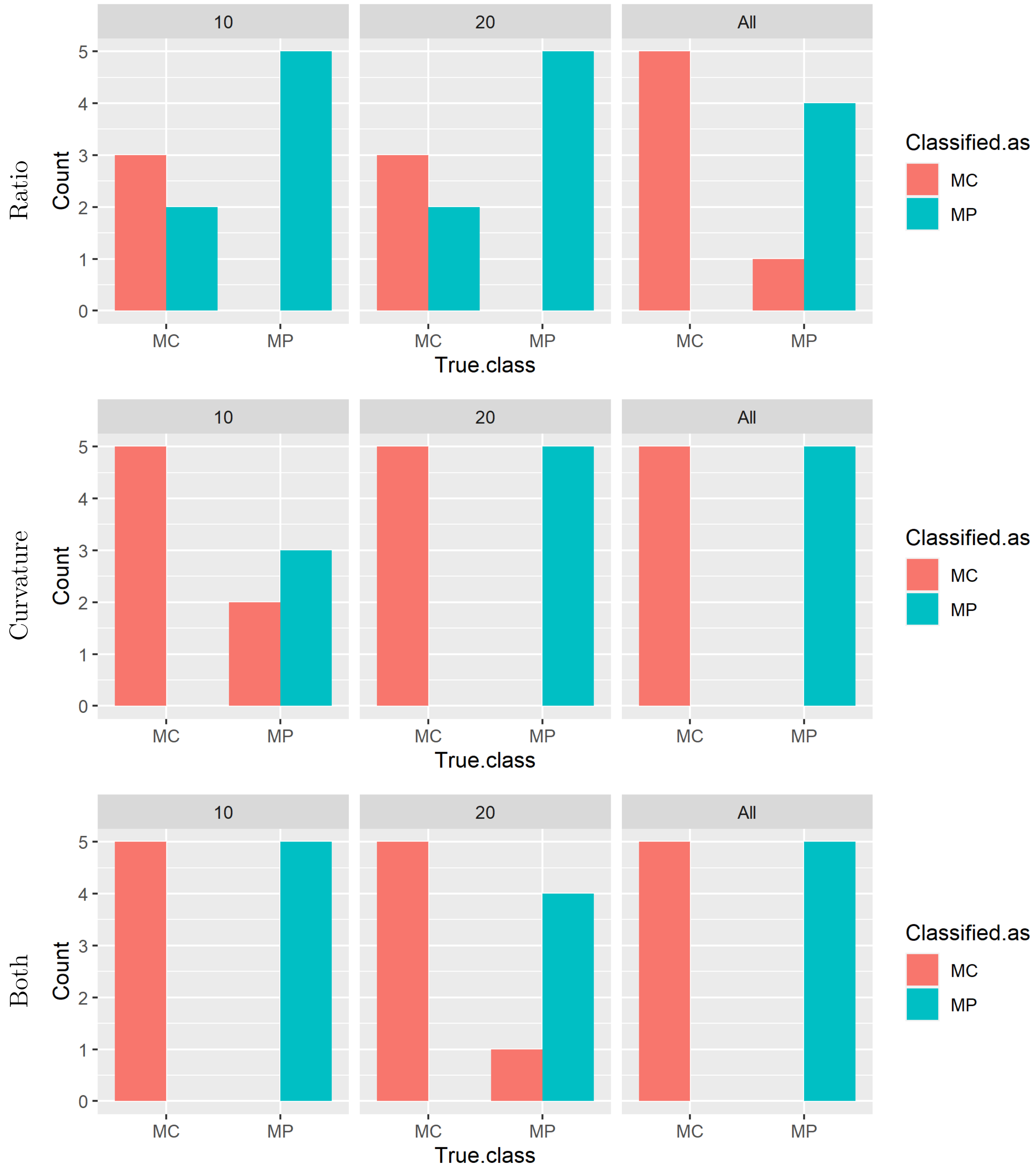


Figure 5.15: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 20%, 20% and 10% for 10, 20 and 'all' components, respectively, when using only the ratio, 20%, 0% and 0% when using only the curvature and 0%, 10% and 0% when using both characteristics for a sample of 20 realisations that were osculated by a disc of radius $r = 5$.

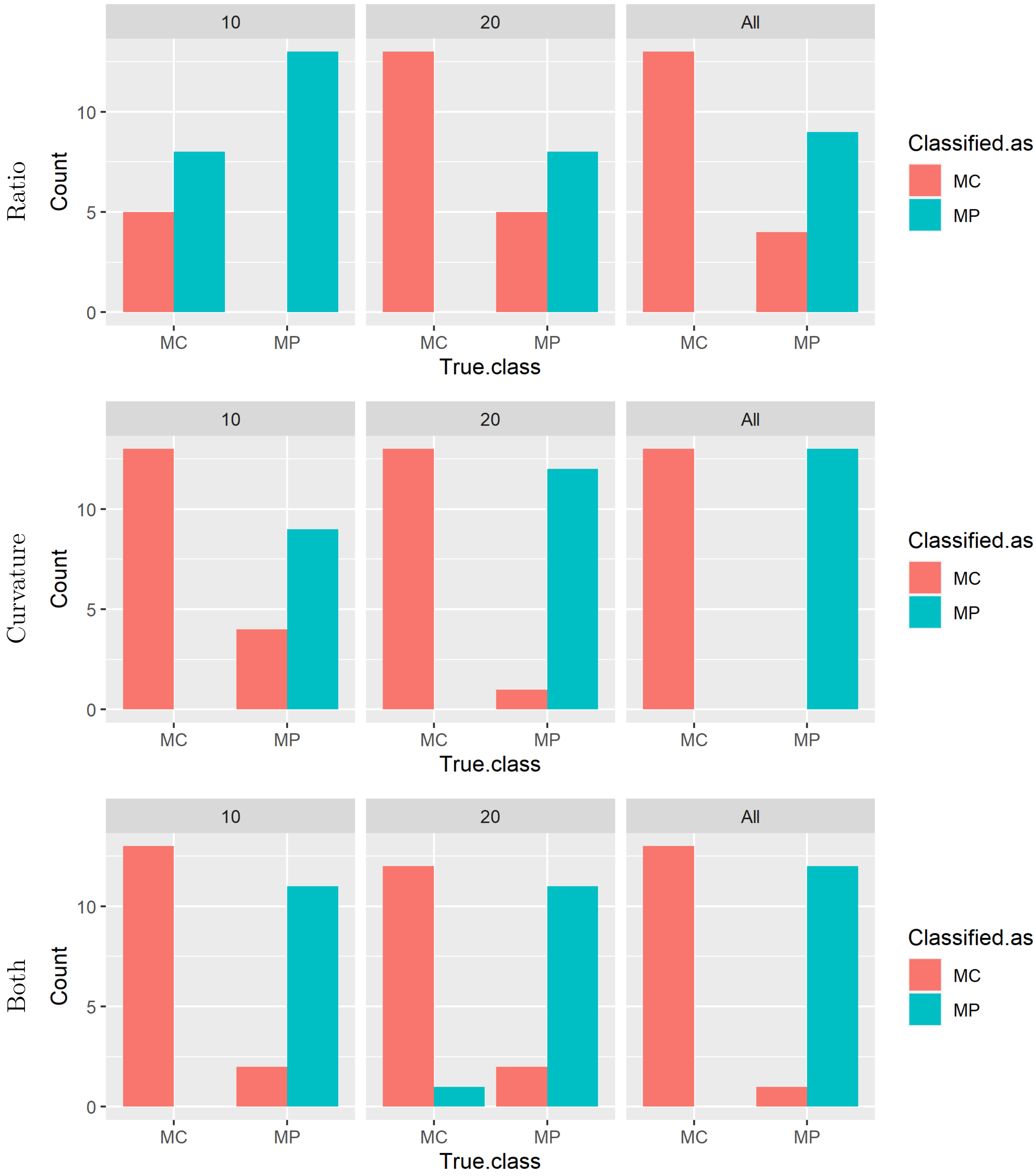


Figure 5.16: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 30.8%, 19.2% and 15.4% for 10, 20 and 'all' components, respectively, when using only the ratio, 15.4%, 3.8% and 0% when using only the curvature and 7.7%, 11.5% and 3.8% when using both characteristics for a sample of 50 realisations that were osculated by a disc of radius $r = 5$.

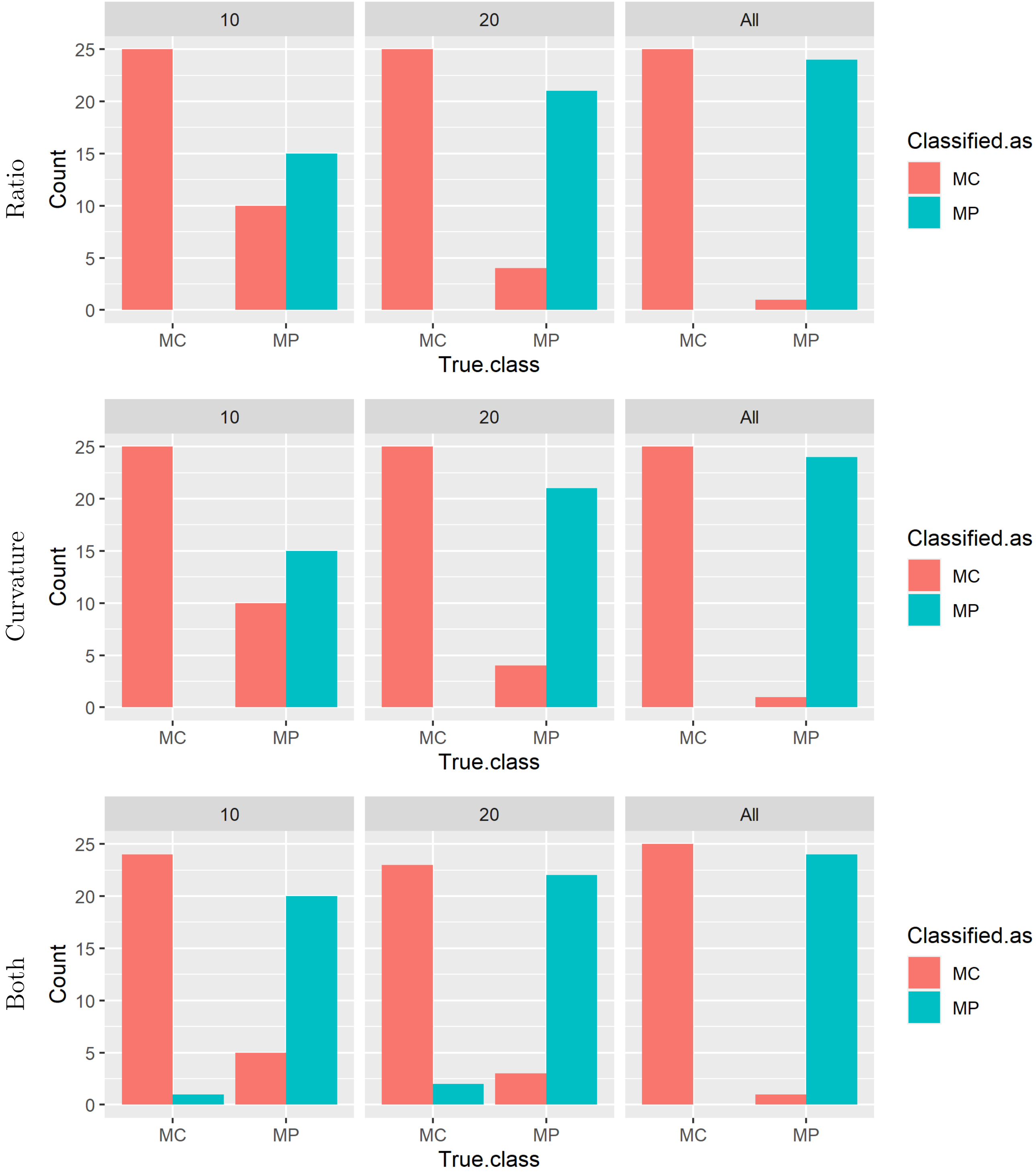


Figure 5.17: Histograms of k-means classification accuracy using only the ratio, only the curvature and both ratio and curvature for discrimination when using a sample of 10, 20, and 'all' components, respectively. Misclassification rates are 20%, 8% and 2% for 10, 20 and 'all' components, respectively, when using only the ratio, 20%, 8% and 2% when using only the curvature and 12%, 10% and 2% when using both characteristics for a sample of 100 realisations that were osculated by a disc of radius $r = 5$.

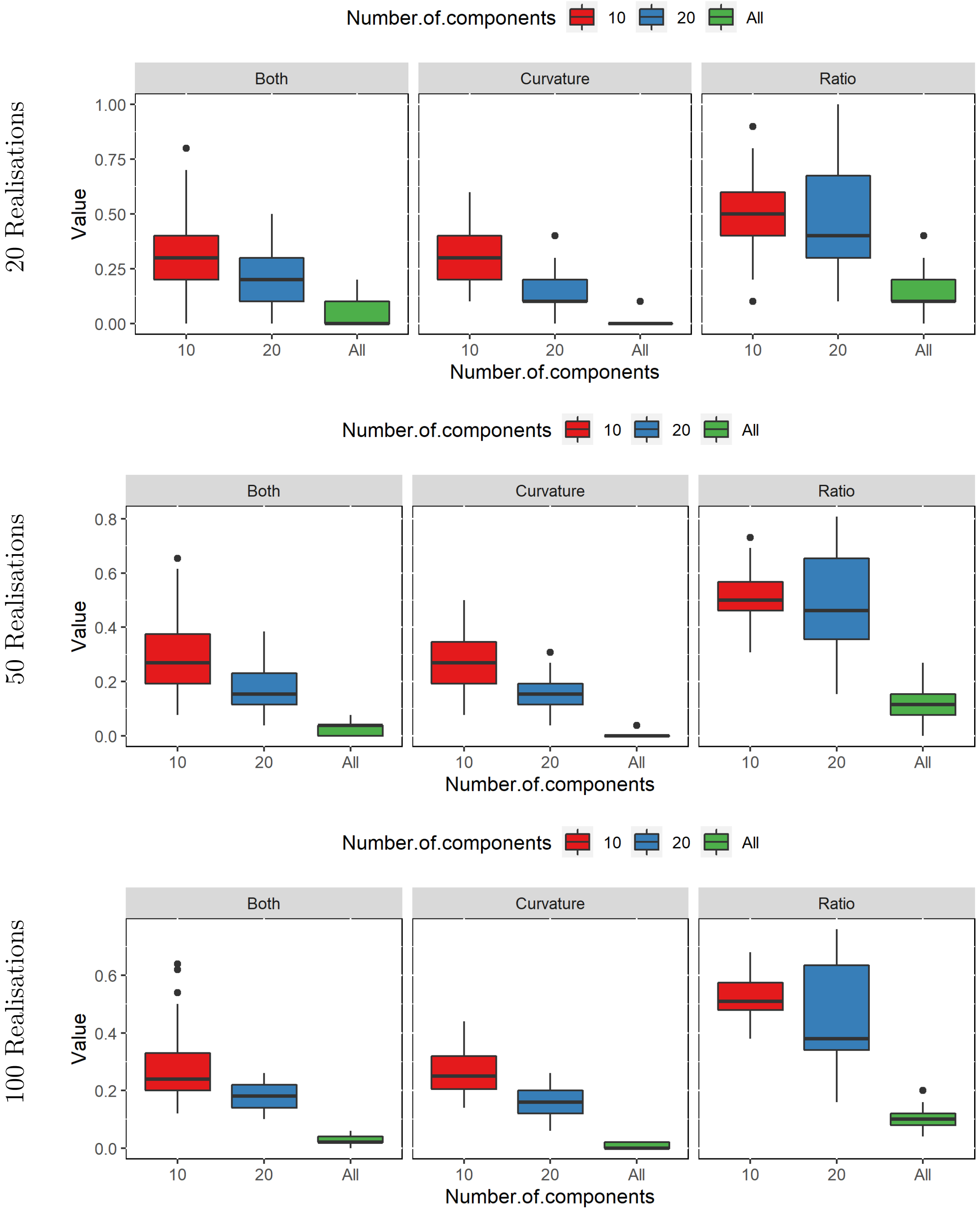


Figure 5.18: Boxplots of misclassification rate for 50 runs of k-means algorithm when considerring samples of 20 (top), 50 (central) and 100 (bottom) realisations using both ratio and curvature, only the curvature and only the ratio for discrimination, respectively. For each setting, misclassification rates for different number of components considered (namely 10, 20 and 'all') are shown. Note that the characteristics were obtained using an osculating disc of radius $r = 5$.

Chapter 6

Conclusion

The first goal of this thesis was to study classification methods for multidimensional and functional data. A brief summary of the basic theory of nonparametric functional data analysis, needed to successfully complete the main goal of this thesis, was introduced in Section 2.3. Since the main focus of this thesis is on the (planar) random sets, an introduction to stochastic geometry was also made in the same chapter.

The second and main goal of this thesis was to suggest and implement a suitable classifier of realisations of random sets. In Chapter 3 we constructed two classifiers, one for supervised learning based on the k-nearest neighbours algorithm and kernel-type estimator of posterior probabilities (defined in Section 2.3.1) and one for unsupervised learning based on the k-means clustering algorithm (defined in Section 2.3.2). In order to access the classification of realisations of random sets, a link between the methods used for random sets and the methods used for functional data analysis had to be established. This was done using the method proposed in [31], where functional data were derived from a realisation of a random set by evaluating the curvature measure at the points of the boundary and evaluating the ratio of the perimeter and the area of the components inside the realisation, as described in Section 2.6.1. In the supervised case, functional data are directly fed to the classifier, while in the unsupervised case, a mean from a sample of data is first evaluated. After that, we use an appropriate kernel proposed in [6] for estimating the $\mathcal{N}$-distance (see Section 2.4), which is used as a semi-metrics for measuring the distance between functional data (or the evaluated means in the unsupervised case).

In Chapter 4, we validated the procedures by applying them to simulated data studied by different authors. Due to the potential correlation between the individual components within a realisation (densely packed components can affect the shape of each other), we decided to follow the same validation procedure as proposed in [31]. We randomly chose a sample of components from a realisation, evaluated the functional characteristics and passed it on to the classifiers.

From the histograms of the classification accuracy, we conclude that the supervised classification classifier gives satisfactory results. Of course, the classifier exhibits some limitations. The first is the dependence of the accuracy on the amount of data at hand. The classifier gave better results when it had more data to learn from, which was expected. This dependence is observable on both the number of realisations (the more realisations are at hand, the higher the accuracy of classification) and on the number of components sampled from a realisation (the greater the sample, the higher the accuracy). The highest accuracy was obtained when both characteristics were used simultaneously for classification, reflecting the results obtained in [31]. It was also observed that the classification accuracy increases if we evaluate the functional characteristic reflecting the curvature of the boundary at more points, which was expected. However, the number of these points is limited by the size of the osculating disc because a disc too large cannot detect local changes in curvature.

From the histograms of the classification accuracy for the unsupervised classification, we observed higher misclassification rates than for the supervised case. In this case, dependency on the amount of data at disposal is not straightforward as in the unsupervised case – the classification accuracy increases with increasing number of components that are used, while it slightly decreases with increasing number of realisations. This could have been caused by the usage of the mean, which can be non-informative when data are rough, which is in our case caused by discretisation, instead some other centrality notion, e.g. mode, which is more robust since it is less sensitive to the outliers. The evaluation of the functional characteristic reflecting the curvature in more positions slightly improved the results.

As a final step, we applied the procedure to the samples representing two types of mammary tissue (mastopathic and mammary cancer tissue). The classification accuracy followed the pattern of the accuracy for simulated data: it increased with increasing number of realisations in the supervised classification while it slightly decreased in the unsupervised classification case. In both cases, the accuracy increased with increasing number of components. Taking into account the challenges posed by the variability in the shapes and sizes of components within the same type of tissue, as well as the difficulties in identifying distinctive features for different types of tissues, the results obtained can be considered satisfactory in both cases.

The research brought some possibilities of further study, e.g. the procedure maybe can be improved by deriving optimal values for the size of the osculating circle or by using the curve-smoothing methods. However, it is obvious that the presented procedure shows great potential for being used as a method for the classification of realisations of random sets in the form of binary images.

Bibliography

1. Diggle, P. J. & Milne, R. K. Bivariate Cox Processes: Some Models for Bivariate Spatial Point Patterns. *Journal of the Royal Statistical Society. Series B (Methodological)* **45,** 11–21. ISSN: 00359246. `http://www.jstor.org/stable/2345617` (1983).

2. Grabarnik, P., Pagès, L. & Bengough, A. Geometrical properties of simulated maize root systems: Consequences for length density and intersection density. *Plant and Soil* **200,** 157–167 (Mar. 1998).

3. Kadashevich, I., Schneider, H.-J. & Stoyan, D. Statistical modeling of the geometrical structure of the system of artificial air pores in autoclaved aerated concrete. *Cement and Concrete Research* **35,** 1495–1502 (Aug. 2005).

4. Mrkvička, T. & Mattfeldt, T. Testing histological images of mammary tissues on compatibility with the Boolean model of random sets. *Image Analysis and Stereology* **30,** 101–108 (Mar. 2011).

5. Hermann, P. *et al.* Fractal and stochastic geometry inference for breast cancer: a case study with random fractal models and Quermass-interaction process. *Statistics in medicine* **34,** 2636–2661 (Apr. 2015).

6. Gotovac, V. & Helisová, K. Testing Equality of Distributions of Random Convex Compact Sets via Theory of N-Distances. *Methodology and Computing in Applied Probability.* `https://doi.org/10.1007/s11009-019-09747-z` (2021+).

7. Illian, J., Penttinen, A., Stoyan, H. & Stoyan, D. *Statistical Analysis and Modelling of Spatial Point Patterns* English. ISBN: 978-0-470-01491-2 (John Wiley and Sons, Chichester, United Kingdom, 2008).

8. Baddeley, A. & Jensen, E. *Stereology for Statisticians* ISBN: 9780203496817. `https://books.google.cz/books?id=il0fXb%5C_GSowC` (CRC Press, 2004).

9. Chiu, S., Stoyan, D., Kendall, W. & Mecke, J. *Stochastic Geometry and Its Applications* ISBN: 9780470664810 (John Wiley and Sons, Chichester, United Kingdom, Sept. 2013).

10. Michie, D., Spiegelhalter, D. & Taylor, C. Machine Learning, Neural and Statistical Classification. *Technometrics* **37** (Jan. 1999).

11. Ferraty, F. & Vieu, P. *Nonparametric Functional Data Analysis: Theory and Practice* ISBN: 9780387366203. `https://books.google.cz/books?id=lMy6WPFZYFcC` (Springer New York, 2006).

12. Pawlasová, K. & Dvořák, J. Supervised Nonparametric Classification in the Context of Replicated Point Patterns. *Image Analysis and Stereology* **41,** 57–109. `https://www.ias-iss.org/ojs/IAS/article/view/2652` (July 2022).

13. Helemskiĭ, A. Y. *Lectures And Exercises on Functional Analysis* in (2006). `https://api.semanticscholar.org/CorpusID:125195489`.

14. Muscat, J. & Buhagiar, D. Connective spaces. *Series B: Mathematical Science* **39,** 1–13 (Jan. 2006).

15. Cohn, D. L. Measure theory (2013).

16. Lavie, M. Characteristic function for Random Sets and Convergence of Sums of Independent Random Sets. *Acta Mathematica Vietnamica* **25,** 87–99 (Jan. 2000).

17. Hunter, J. K. & Nachtergaele, B. *Applied Analysis* eprint: `https://www.worldscientific.com/doi/pdf/10.1142/4319`. `https://www.worldscientific.com/doi/abs/10.1142/4319` (WORLD SCIENTIFIC, 2001).

18. Helemskiĭ, A. Y. *Lectures And Exercises on Functional Analysis* in (2006). `https://api.semanticscholar.org/CorpusID:125195489`.

19. Hitzler, P. & Seda, A. Dislocated Topologies. *J. Electr. Eng.* **51** (Sept. 2000).

20. Feller, W. *An introduction to probability theory and its applications: Volume I* 3rd (Wiley, 2009).

21. Arthur, D. & Vassilvitskii, S. *K-Means++: The Advantages of Careful Seeding* in. **8** (Jan. 2007), 1027–1035.

22. Klebanov, L. *N-distances and Their Applications* ISBN: 80-246-1152-X (Karolinum Press, Charles University, Prague, Jan. 2005).

23. Baccelli, F. & Blaszczyszyn, B. Stochastic Geometry and Wireless Networks: Volume I Theory. *Foundations and Trends in Networking* **3,** 249–449 (Jan. 2009).

24. Kendall, W., Lieshout, M. & Baddeley, A. Quermass-Interaction Processes: Conditions for Stability. *Adv. Appl. Prob.* **31** (Nov. 1998).

25. Gotovac, V. Similarity Between Random Sets Consisting of Many Components. *Image Analysis & Stereology* **38,** 185–199 (July 2019).

26. Bullard, J., Garboczi, E., Carter, W. & Fuller, E. Numerical methods for computing interfacial mean curvature. *Computational Materials Science* **4,** 103–116. ISSN: 0927-0256. `https://www.sciencedirect.com/science/article/pii/092702569500014H` (1995).

27. Radović, B. *Similarity of Random Sets* PhD thesis (Czech Technical University in Prague. Computing and Information Centre., 2021), 27–29. `http://hdl.handle.net/10467/94679`.

28. Gotovac, V., Helisová, K. & Ugrina, I. Assessing Dissimilarity of Random Sets Through Convex Compact Approximations, Support Functions and Envlope Tests. *Image Analysis & Stereology* **35,** 181–193 (Dec. 2016).

29. Debayle, J., Gotovac, V., Helisová, K., Staněk, J. & Zikmundová, M. Assessing Similarity of Random sets via Skeletons. *Methodology and Computing in Applied Probability,* 471–490. `https://doi.org/10.1007/s11009-020-09785-y` (2021).

30. Møller, J. & Helisová, K. Power diagrams and interaction process for unions of discs. *Advances in Applied Probability* **40,** 321–347 (June 2008).

31. Gotovac Đogaš, V. *et al.* Two-step method for assessing dissimilarity of random sets. *Image Analysis and Stereology* **40,** 127–140 (2021).

32. Fraser, W. & Gotlieb, C. C. A calculation of the number of lattice points in the circle and sphere. *Mathematics of Computation* **16,** 282–282. `https://doi.org/10.1090/s0025-5718-1962-0155788-9` (Sept. 1962).

33. Sloane, N. J. A. *The On-Line Encyclopedia of Integer Sequences® (OEIS®)* [Accessed on 15.12.2023]. Apr. 1991. `https://oeis.org/A000328/b000328.txt`.

34. *sklearn.cluster.KMeans — scikit-learn.org* [Accessed 05.12.2023]. `https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html#sklearn.cluster.KMeans`.

Contents of Enclosed CD

- readme.txt .... the file with CD contents description
- create_tissue.py .... the program for data augmentation
- Kmeans .... the directory with k-means source codes
  - average_sample_simulated.py ... the program for calculating the mean of a sample of components inside the realisation of the simulated data
  - average_sample_tissue.py ...... the program for calculating the mean of a sample of components inside the realisation adapted to work with tissues data
  - kmeans_classifier_simulated.pythe program implementing k-means procedure for simulated data
  - kmeans_classifier_tissue.py ... the program implementing k-means procedure for tissues data
- Knn .... the directory with k-nearest neighbours source codes
  - knn_classifier_simulated.r ..... the program implementing k-nearest neighbours procedure for simulated data
  - knn_classifier_tissue.r .... the program implementing k-nearest neighbours procedure for tissue data