MSc Mathematics

Track: Biomedical Mathematics

Master thesis

Spatial imaging statistics for prediction of survival of primary CNS lymphoma patients

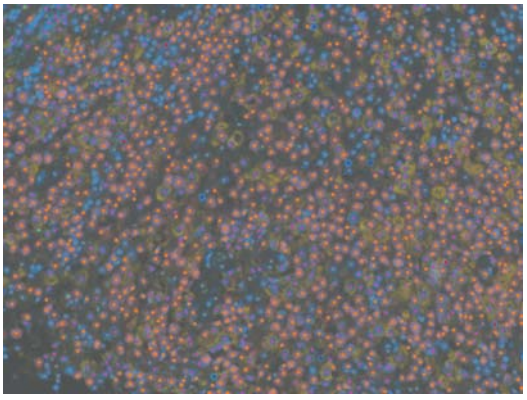
by

Erik Bosch

March 13, 2021

Supervisors: Tim van de Brug, Department of Epidemiology & Data Science of Amsterdam UMC; Yongsoo Kim, Department of Pathology of Cancer Centrum Amsterdam

Second examiner: Wessel van Wieringen, Department of Mathematics of the Vrije Universiteit Amsterdam



Department of Mathematics

Faculty of Sciences

VU UNIVERSITY AMSTERDAM

Abstract

Treatment of patients with primary Central Nervous System lymphoma is a process which brings high risk for the already staggering health of the patient. Consequently, knowing if treatment will improve survival of patients is essential. Recently, questions have risen whether a spatial analysis of the tumor micro-environment of patients can help in predicting their survival after receiving treatment. Here, we present several prediction models for the survival of primary Central Nervous System lymphoma patients. We extract an extensive list of radiomic features from the tumor micro-environment of patients, by representing the tumor micro-environment as a Marked Poisson Point Process. The radiomic features are then used in prediction models to predict the survival of patients, where the survival is quantified by the six and twelve months Event Free Survival. Results show that the prediction models perform better than random, with a minimal Area Under the Curve of 0.645 and maximal Area Under the Curve of 0.758. An analysis of the relative feature importance confirms that features related to Macrophages are of importance in the prediction of survival. In addition, results show that this can be specified more to spatial features based of phenotypes interacting with Macrophages.

Title: Spatial imaging statistics for prediction of survival of primary CNS lymphoma patients
Author: Erik Bosch, erikbosch94@gmail.com, 2539720
Supervisors: Tim van de Brug, Department of Epidemiology & Data Science of Amsterdam UMC; Yongsoo Kim, Department of Pathology of Cancer Centrum Amsterdam
Second examiner: Wessel van Wieringen, Department of Mathematics of the Vrije Universiteit Amsterdam
Date: March 13, 2021

Department of Mathematics
VU University Amsterdam
de Boelelaan 1081, 1081 HV Amsterdam
http://www.math.vu.nl/

Contents

List of Figures

1 Introduction

Central Nervous System lymphoma (CNS lymphoma) is a typed cancer mainly developing from lymphocytes (Taylor (2000)). CNS lymphoma is categorized by the World's Health Organisation into *primary CNS lymphoma* (PCNSL) and secondary CNS lymphoma.

PCNSL refers to cases mostly confined to the brain, eyes and spinal cord (Cai et al. (2019)). PCNSL is rare, but aggressive: Overall survival of PCNSL patients is 1.5 months when untreated, and the five year survival rate is 30% when treated (Yang and Liu (2017)). The standard treatment for PCNSL patients is high-dosed chemotherapy. Although not many patients tolerate this chemotherapy, such treatment has been leading to improved survival rates of treated patients (Cai et al. (2019)).

Although survival of treated patients has improved, treatment still has high risk and great impact on patients. Questions have risen whether properties of the tumor microenvironment of patients, analyzed before treatment, can predict the survival of the patients after treatment. Such properties could give an indication whether treatment for a patient would benefit them. This then contributes to the goal of developing personalized treatment for patients in health-care.

1.1 Literature review

The micro-environment of PCNSL tumors has been studied for several cohorts of patients. Biopts of the PCNSL tumors are treated with specific staining of biological markers such that cell phenotypes can be distinguished from each other. Recent studies mostly involved expression of markers of Macrophages, Tcells and Tumor cells. The stratification of cohorts of patients in these studies is commonly done by the high or low expression of biological markers. Nevertheless, studies turned out to have varied results.

High expression of the Macrophage marker CD68+ was associated with favorable prognosis of PCNSL in Marcelis et al. (2020) and in Furuse et al. (2020), while high expression of CD68 was associated with inferior overall survival in Cho et al. (2017) and in Li et al. (2019).

The presence of the marker PD1+ on tumor infiltrated Tcells was associated with poor prognosis in Four et al. (2017) but with favorable prognosis in Kim et al. (2019).

Feng et al. (2017) took a different approach of the analysis of the micro-environment. In addition to the standard analysis of high and low expression of markers in the micro-environment, Feng et al. (2017) performed spatial analysis of the micro-environment.

Patients were stratified by the amount of CD8 markers in both the Tumor regions as the Stroma regions of the invasive margin, regions seen in Figure 1.1. Nevertheless,

patients were not stratified based on the amount of FoxP3 and PD-L1 markers in both the Tumor regions as the Stroma regions. Surprisingly, it looked like having more FoxP3 in those regions improved survival rate, although not significant (p-value of 0.18 and 0.08, respectively). This is in contrary to their believe of FoxP3's biological mechanism, as FoxP3 is a biological marker for Tumor cells.

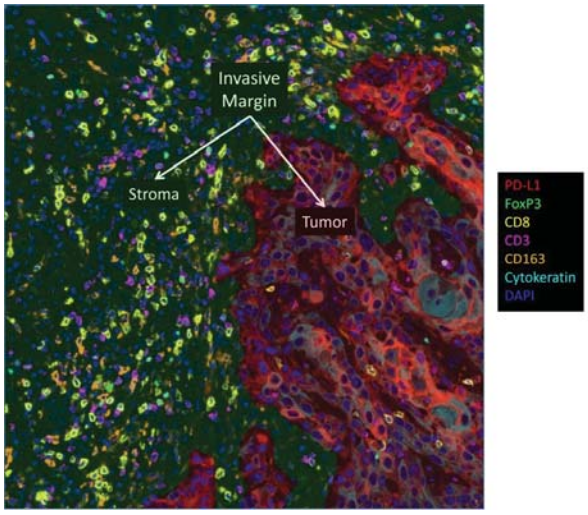


Figure 1.1: Distribution of phenotypes in the invasive margin of a tumor. Distribution of phenotypes clearly differs in the Stroma region compared with the Tumor region. Original figure adapted from Feng et al. (2017).

Feng et al. (2017) continued with a spatial analysis of the micro-environment to investigate their surprising results. This included finding the counts of FoxP3 in a radius of 30 μm of CD8 markers in the Tumor and Stroma regions.

The spatial analysis showed, after normalization by the amount of CD8, that more normalized counts of FoxP3 around CD8 significantly lowered survival rates. The corresponding p-values were 0.005 and 0.024 for the Tumor region and Stroma region, respectively.

To further investigate, the ratio of FoxP3 and CD8 was calculated in both the Tumor and Stroma regions. These ratios were non-significant with p-values of 0.26 and 0.10, respectively.

Both findings gave rise to a contradiction that high expression of FoxP3 improved survival rates.

These findings show the importance of spatial analysis of biological tissue.

The technical aspect behind the analysis of the micro-environment has improved over the years. Recently, the amount of different biological markers that can be detected simultaneously in patient tissue, has increased from four to eight. This gives research experts the tools to observe and analyse even more from the micro-environment of patients.

1.2 Research Objectives

To contribute to the recent development of the need of spatial analysis of (PCSNL-) patient tissue, the project consists of the following goals:

- Obtain an extensive list of spatial features that quantify the micro-environment of patients,
- Develop a prediction model for immune therapy response, using the obtained spatial features.

The main goal of developing a prediction model is to test the relevance of the spatial features for this cohort of patients. As described in Section 1.1, previous research has shown that biological markers for Macrophages are important for prediction of survival of PCSNL patients. Therefore, we expect the results of our models to reflect this statement.

To achieve the goals of this project, we build several prediction models for survival of PCNSL patients. Survival is quantified by the *Event Free Survival* (EFS), available as clinical data for this project. Two sets of response data for the prediction models are obtained by thresholding the EFS for each patient at six and twelve months. Several Multi Spectral Images (MSIs) are generated from biological tissue of each patient. Each MSI represents the micro-environment of the patient, from which a large amount of features are extracted. The final features at patient level to predict the response data are obtained by averaging the features over the different MSIs per patient.

Each patient MSI is represented as a *Marked Poisson Point Process*, a mathematical object, which is used to calculate the features. Features range from simple spatial statistics like the *counting statistic* and median distances between phenotypes, to more complex spatial statistics like the *Empty Space function* and *Ripley's K-function*. The prediction models constructed in this project are Ridge logistic regression, Empirical bayes Co-data learnt Prediction and Covariate Selection (ECPC) and Co-data guided Random Forest (CoRF). Methods such as *cross-validation*, *ROC-curves* and corresponding analysis of the Area Under the Curve (AUC) are used to evaluate the performance of the models. In addition, group weights obtained from ECPC are stated; the relative feature importance landscape obtained from CoRF is also shown. All results are stated, upon which we discuss the results and state our conclusion. At last, we discuss the project itself, and and give recommendations for future research.

3 Point Processes

As mentioned before, we will be using Marked Poisson Point Processes as representation of a patient MSI. Marked Poisson Point Processes are mathematical objects that enable us to calculate features for the prediction model. Before we discuss the features derived from the Marked Poisson Point Processes, we will give a small introduction to the theory of Point Processes. This chapter is derived from Kingman (2005), figures are created in the programming language *R* (R Core Team (2020)) with the package *spatstat* (Baddeley and Turner (2005)), a specialized package for analyzing spatial point patterns.

3.1 The Poisson distribution

The Poisson distribution is the building block for Poisson Point Processes. The Poisson distribution is a probability distribution that calculates the probability of a random variable N to be equal to a non-negative integer n. It is defined as

$$\mathbb{P}(N = n) = \frac{\lambda^n}{n!} e^{-\lambda}, \tag{3.1}$$

for $n \in \mathbb{N}$. The positive real parameter λ characterizes the Poisson distribution and is called the *density*. The random variable N is Poisson distributed with parameter λ and denoted by $N \sim Pois(\lambda)$. The density λ turns out to be the expected value of the Poisson distribution, as

$$\begin{aligned}
\mathbb{E}(N) &= \sum_{k=0}^{\infty} k\mathbb{P}(N = k) = \sum_{k=0}^{\infty} k\frac{\lambda^k}{k!}e^{-\lambda} = e^{-\lambda}\sum_{k=1}^{\infty} k\frac{\lambda^k}{k!}, \\
&= e^{-\lambda}\sum_{k=1}^{\infty}\frac{\lambda^k}{(k-1)!} = \lambda e^{-\lambda}\sum_{k=1}^{\infty}\frac{\lambda^{k-1}}{(k-1)!}, \\
&= \lambda e^{-\lambda}\sum_{k=0}^{\infty}\frac{\lambda^k}{k!} = \lambda e^{-\lambda}e^{\lambda} = \lambda,
\end{aligned}$$

where the second to last equation is due to the Taylor-series of the exponential function.

The Poisson distribution has the useful property that it is additive, when the distributions are independent. Two probability distributions M and N are independent if for all natural numbers m, n the following holds

$$\mathbb{P}(M = m, N = n) = \mathbb{P}(M = m)\mathbb{P}(N = n).$$

Now suppose that two Poisson random variables $M \sim Pois(\lambda)$ and $N \sim Pois(\mu)$ are independent. Then the distribution of the random variable K defined by $K = M + N$ is Poisson distributed with parameter $\lambda + \mu$. This follows for $k \geq 0$, such that

$$\begin{aligned}
\mathbb{P}(M+N=k) &= \sum_{j=0}^{k} \mathbb{P}(M+N=k, M=j) = \sum_{j=0}^{k} \mathbb{P}(N=k-j, M=j), \qquad (3.2)\\
&= \sum_{j=0}^{k} \mathbb{P}(N=k-j)\mathbb{P}(M=j) = \sum_{j=0}^{k} \frac{\mu^{k-j}}{(k-j)!} e^{-\mu} \frac{\lambda^j}{j!} e^{-\lambda},\\
&= e^{-(\mu+\lambda)} \frac{1}{k!} \sum_{j=0}^{k} \frac{k!}{(k-j)!j!} \mu^{k-j} \lambda^j,\\
&= e^{-(\mu+\lambda)} \frac{1}{k!} \sum_{j=0}^{k} \binom{k}{j} \mu^{k-j} \lambda^j = \frac{(\mu+\lambda)^k}{k!} e^{-(\mu+\lambda)} \qquad (3.3)
\end{aligned}$$

where we used the independence statement in Equation (3.2) and the Binomium of Newton[1] in Equation (3.3). This shows that $K \sim Pois(\lambda + \mu)$. The latter result can be extended for the sum of a countable amount of independent random variables. This is called the superstition theorem, which we will not prove here. We refer for more details and the proof to Kingman (2005).

The superstition theorem is as follows: For $m \in \mathbb{N}$, let $M_1, \ldots, M_m$ be a sequence of independent, Poisson distributed random variables with respectively finite parameters $\lambda_1, \ldots, \lambda_m$. The random variable M is defined as $M = \sum_{i=1}^{m} M_i$ and it follows that $M \sim Pois\left(\sum_{i=1}^{m} \lambda_i\right)$.

The additivity of the Poisson distribution has an important use in modeling of Marked Poisson Point Processes, which we will see in Section 3.3. In the next section we will see that the Poisson distribution is the building block of the modeling of spatial data.

3.2 Poisson Point Process

For the definition of a Poisson Point Process we will define a general form of the Poisson distribution discussed in Section 3.1.

Let B be a measurable set in the Euclidean space $\mathbb{R}^d$ for any $d \in \mathbb{N}$ and denote $N(B)$ for the number of points in the set B. Then $N(B)$ is a random variable and the probability that $N(B)$ equals n is given by

$$\mathbb{P}(N(B)=n) = \frac{\Lambda(B)^n}{n!} e^{-\Lambda(B)}, \qquad (3.4)$$

with non-negative integer n and Λ is a parameter that characterizes the shape of the distribution. How the parameter Λ is defined characterizes what type of Poisson Point Process it is, which we will see in the following subsections.

[1]The Binomium of Newton is defined as $(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^{n-k} y^k$ for non-negative integer n.

3.2.1 Homogeneous Poisson Point Processes

A *homogeneous* Poisson Point Process is characterized by the parameter Λ, which has the form $\Lambda(B) = \lambda \cdot \mathcal{L}(B)$, where λ is finite and $\mathcal{L}(B)$ is the Lebesgue measure of the set B. Analogue to the calculations done in Section 3.1, it follows that the parameter Λ is the expectation of $N(B)$, so $\mathbb{E}(N) = \lambda \cdot \mathcal{L}(B)$.

A homogeneous Poisson Point Process has the property of being *stationary* (invariant to translation). For any $x \in \mathbb{R}^d$, we define the set

$$x + B := \{x + y \mid y \in B\}.$$

Then, if for any $B \subset \mathbb{R}^d$ the following equality $\Lambda(B) = \Lambda(x + B)$ holds, we say that the Poisson Point Process is *stationary*. The Poisson Point Processes that are homogeneous are the only Poisson Point Processes that are stationary.

3.2.2 Inhomogeneus Poisson Point Processes

A Poisson Point Process that is not clasified as homogeneous is called a *inhomogeneous* Poisson Point Process. For such Poisson Point Processes, the parameter λ is assumed to be variable. For an inhomogeneous Poisson Point Process, the Lebesgue measure of the set B does not change, but the parameter λ becomes a variable depending on the location in B. This is written as $\Lambda(B) = \lambda(B) \cdot \mathcal{L}(B)$, where $\lambda(B) := \int_B \lambda(x)dx$. In this case, λ is a function from $\mathbb{R}^d$ to the non-negative reals which must satisfy the inequality $\int_B \lambda(x)dx < \infty$, for bounded sets B.

A numerical example for a homogeneous and an inhomogeneous Poisson Point Process is shown in Figure 3.2. Figure 3.2a shows a homogeneous Poisson Point Process with $\lambda_{\text{h}} := 50$ and Figure 3.2b shows an inhomogeneous Poisson Point Process with $\lambda_{\text{ih}}(x, y) := 50(1+10y^2)$, both in the unit square $[0, 1] \times [0, 1]$. We see that the definition of this specific λ_{ih} only depends on the $y-$coordinate. As this is an increasing function of y, we expect more points to be placed in the higher regions of the $y-$plane. The level curves of λ_{ih} represent this notion and are shown in Figure 3.1. Both Poisson Point Processes are shown in Figure 3.2.

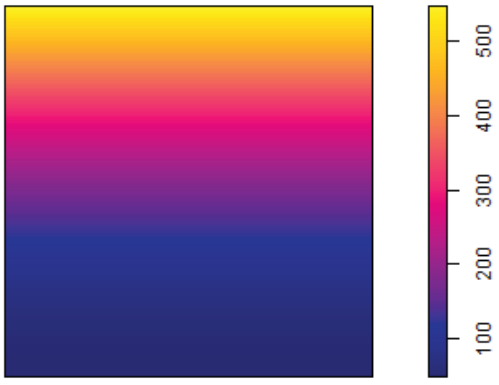
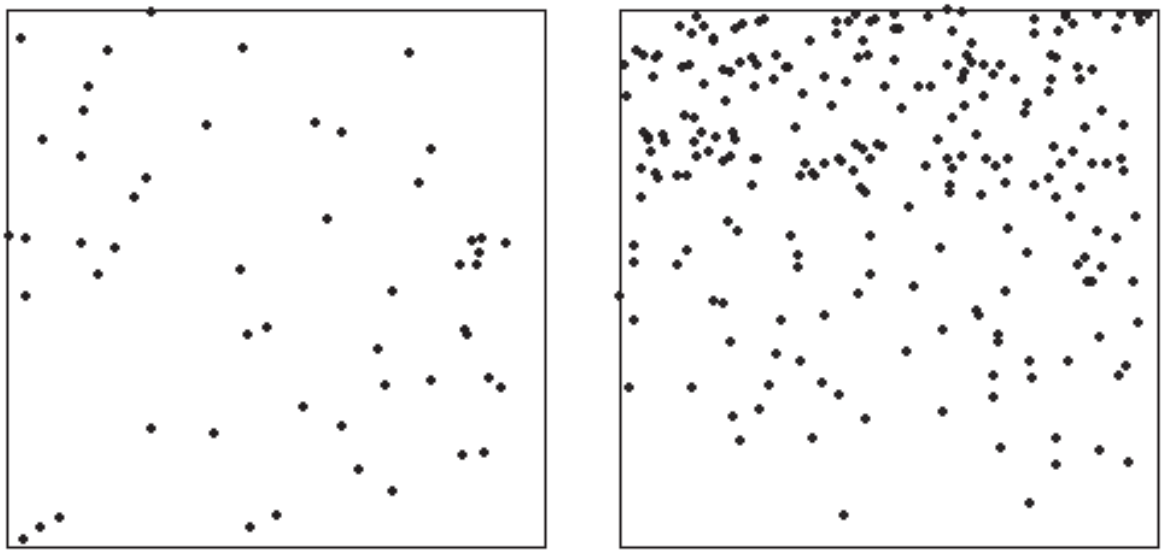


Figure 3.1: The level curves of the density function $\lambda(x, y) = 50(1 + 10y^2)$.



(a) A homogeneous Poisson Point Process generated with density function $\lambda(x, y) = 50$.

(b) An inhomogeneous Poisson Point Process generated with density function $\lambda(x, y) = 50(1 + 10y^2)$.

Figure 3.2: Poisson Point Processes generated with different density functions.

In the following section we will discuss the extension of the notion of a Poisson Point Process to the notion of a Marked Poisson Point Process.

3.3 Marked Poisson Point Process

Marked Poisson Point Processes are extensions of Poisson Point Processes in the sense that every point can have extra information attached to it. This extra information is called a *Mark*. The definition of a Marked Poisson Point Process is almost identical to the definition of the Poisson Point Process and differs from what space the data is generated. Therefore, let us repeat the definition of the Poisson distribution defined in Section 3.1.

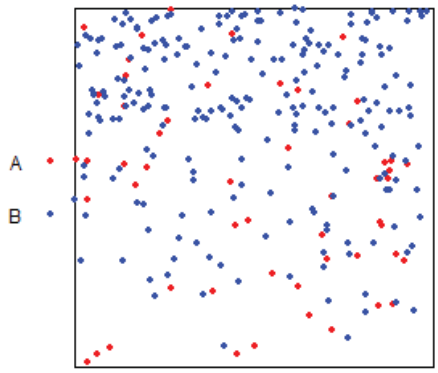
Let B be a set in the Euclidean space $\mathbb{R}^d$ for any $d \in \mathbb{N}$ and denote $N(B)$ for the number of points in the set B. Then $N(B)$ is a random variable and the probability that $N(B)$ equals n is given by

$$\mathbb{P}(N(B) = n) = \frac{\Lambda(B)^n}{n!} e^{-\Lambda(B)},$$

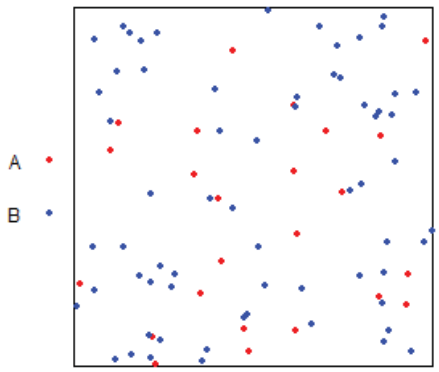
where $n \in \mathbb{N}$ and Λ is a parameter that characterizes the shape of the distribution.

For the Marked Poisson Point Process, we define a new space, which we refer to as the *Mark*-space. This Mark-space can be an arbitrary space, as long as a probability measure is attached to it. Each Mark-space defines what mark each generated data

point can be attached to, and the probability measure defines the probability of each attachment with a data point.



(a) The two Poisson Point Processes in Figure 3.2 combined into one Marked Poisson Point Process with marks A and B, respectively.



(b) Creating a Marked Poisson Point Process by defining the probability measure. By construction, the probability of a point to be marked 'A' was 0.3 and to be marked 'B' was 0.7.

Figure 3.3: Two methods of creating a Marked Poisson Point Process.

In Figure 3.3a, we combined the Poisson Point Processes from Figure 3.2, labeling the homogeneous Poisson Point Process with mark 'A' and the inhomogeneous Poisson Point Process with mark 'B'. The *Mark*-space for this example is then $\mathcal{M} = \{A, B\}$. Now of course, in this case, we labeled the points before combining the two Poisson Point Processes into one Marked Poisson Point Process.

If we want to generate a Marked Poisson Point Process from scratch with this mark-space in the same Euclidean Space, we should first define a probability measure $\mathbb{P}$ on the mark-space. A point is placed in the Euclidean Space depending on the density parameter. Then, for example, a probability measure marks the point as mark 'A' with a probability of 0.3, and as mark 'B' with a probability of 0.7. This example is shown in Figure 3.3b. More complex probability measures can be attached, for example probability measures that mark depending on where in the Euclidean Space the point is.

4 Radiomic features

Radiomics is defined as the conversion of images to high-dimensional data and the subsequent mining of these data for improved decision support (Gillies et al. (2016)). The radiomic features obtained in the mining of such images and the analysis of such can support the diagnosis and prognosis of patients. In this project we do not directly extract features from the patient MSIs, but from the patient MSIs converted to Marked Poisson Point Processes. The conversion of patient MSI to Marked Poisson Point Process enables us to use statistics already defined in the field of spatial statistics.

We will assume in the following sections that X is a Marked Poisson Point Process defined in the Euclidean Space $\mathbb{R}^2$ and the *Mark-Space* $\mathcal{M} = \{\text{Tumor, Tcell, Macrophage, Other}\}$. A subset W of the Euclidean Space $\mathbb{R}^2$ is called the *observation window* and represents the patient MSI. We use the abbreviations {Tu, Tc, Ma, Ot} respectively when indexing over the phenotypes, so for example X_{Tu} is the Poisson Point Process only containing Tumor cells. From now on, we will identify X_i with the set of all cells of phenotype i. We assume that there is some ordering of the cells in X_i, such that the k-th cell in X_i is denoted by x_i^k.

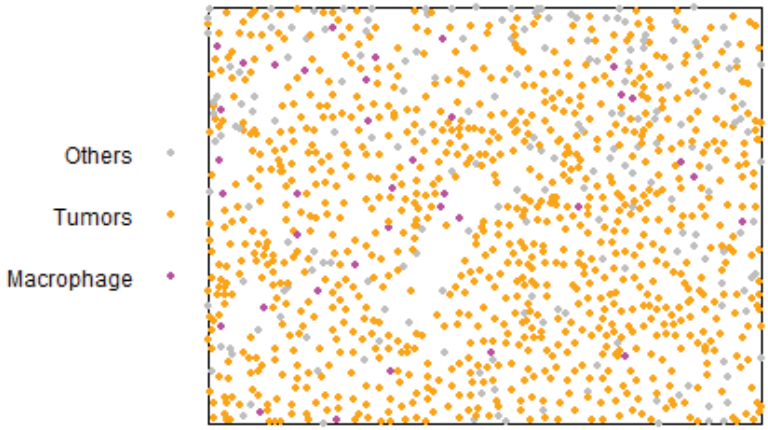
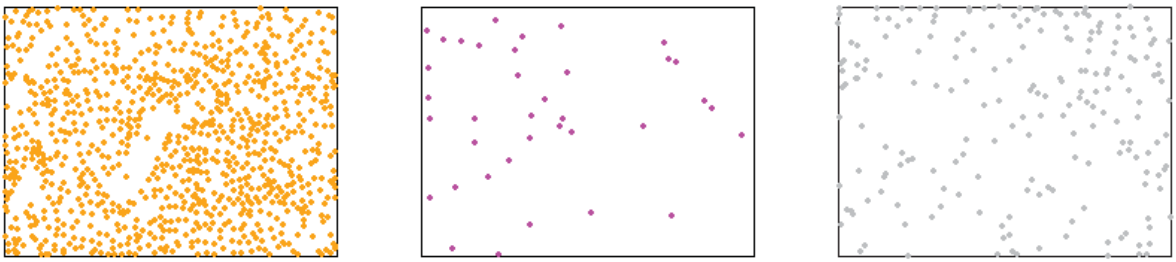


Figure 4.1: Marked Poisson Point Process X reflecting an MSI of a patient with 1153 cells of the three phenotypes 'Tumor'(orange), 'Macrophage'(pink) and 'Other'(grey).



(a) X_{Tu} (b) X_{Ma} (c) X_{Ot}

Figure 4.2: Splitting of the Marked Poisson Point Process from Figure 4.1 according to the phenotypes 'Tumor' (Figure 4.2a), 'Macrophage' (Figure 4.2b) and 'Other' (Figure 4.2c).

In the following sections we define the statistics and corresponding features that are used in the prediction models.

4.1 The counting statistics

Although our main focus is on using spatial statistics, we will first define the statistics related to counting. These are essentially not spatial statistics but are fundamental for the definition of the more complex spatial statistics which we will define in later sections.

4.1.1 The counting statistic

Let X be a Marked Poisson Point Process as usual and let X_i be the set of cells with phenotype i. The *Counting statistic for phenotype* i, denoted as $N_i(W)$, is defined as follows

$$N_i(W) := \#\text{counts of cells with phenotype } i \text{ in the observation window W.} \tag{4.1}$$

From this definition we denote $N_T(W)$ as the total amount of counts in the observation window, and consequently

$$N_{\text{Tu}}(W) + N_{\text{Tc}}(W) + N_{\text{Ma}}(W) + N_{\text{Ot}}(W) = N_{\text{T}}(W). \tag{4.2}$$

From now on, we will assume that the counting statistic is always taken over the observation window W, or else specified. Therefore we write N_T and N_i for $i \in \{\text{Tu, Tc, Ma, Ot}\}$.

As an example, we calculate the counts of the Tumor cells, Tcells, Macrophages and Other cells from the Marked Poisson Point Process in Figure 4.1. In this Marked Poisson Point Process, there are no Tcells, therefore $N_{\text{Tc}} = 0$. The rest of the statistics are given

by $N_{\text{Tu}} = 918$, $N_{\text{Ma}} = 37$ and $N_{\text{Ot}} = 198$. The total amount of cells is computed to be $N_{\text{T}} = 1153$.

The total amount of cells in the Marked Poisson Point Process will be used to calculate the normalized counting statistic.

4.1.2 The normalized counting statistic

Let X be a Marked Poisson Point Process, N_i be the counting statistic of phenotype i and N_{T} the total amount of cells in the Marked Poisson Point Process. Then the *Normalized Counting statistic for phenotype* i, denoted by $\widetilde{N}_i$, is defined as

$$\widetilde{N}_i := \frac{N_i}{N_T}, \tag{4.3}$$

for $i \in \{\text{Tu, Tc, Ma, Ot}\}$.

Continuing the example of the Marked Poisson Point Process in Figure 4.1, we calculate the normalized counts of the Tumor cells, Tcells, Macrophages and Other cells. As mentioned before, there are no Tcells in this Marked Poisson Point Process, therefore also $\widetilde{N}_{\text{Tc}} = 0$. The rest of the statistics is given by $\widetilde{N}_{\text{Tu}} = \frac{918}{1153} \approx 0.80$ and similarly $\widetilde{N}_{\text{Ma}} \approx 0.03$ and $\widetilde{N}_{\text{Ot}} \approx 0.17$.

In the definition of the normalized counting statistic the counts of one phenotype is divided by the total counts of all phenotypes. The latter means that the normalized counting statistic is not solely derived from information of the target phenotype but also from the other phenotypes. The following statistic also has this property, but then uses information from only two phenotypes.

4.1.3 The pairwise relative counting statistic

Following the normalized counting statistic, we will define the pairwise relative counting statistic.

Let X be a Marked Poisson Point Process and as usual, N_i the count statistic of phenotype i, for $i \in \{\text{Tu, Tc, Ma, Ot}\}$. Then we define the *pairwise relative counting statistic* as

$$N_{i,j} := \begin{cases} \frac{N_i}{N_j} & \text{when } N_j > 0, \\ \text{NA}, & \text{otherwise}, \end{cases} \tag{4.4}$$

for $i, j \in \{\text{Tu, Tc, Ma, Ot}\}$ and $i \neq j$. As dividing by zero is undefined, we will set the value of the feature to *Not Applicable*, denoted *NA*.

The pairwise relative counting statistic N_{ij} calculates how many cell of phenotype i there are per cell of phenotype j.

For the Marked Poisson Point Process in Figure 4.1, we calculate the amount of Tumor cells per Macrophage, which is $\widetilde{N}_{\text{Tu,Ma}} = \frac{918}{37} \approx 24.81$, close to 25 tumor cells per macrophage. Because there are no Tcells in this Marked Poisson Point Process, there are zero Tcells per Tumor cell.

4.2 The density statistic

The density statistic is closely related to the normalized counting statistic as it takes a different value to normalize the counts, namely the area.

Let X be a Marked Poisson Point Process and denote Area(B) for the area of subset B of $\mathbb{R}^2$ and the count statistic N_i. The *density statistic* $\lambda_i(W)$ is defined as the ratio of the counts of cells of phenotype i and the area of the observation window. Thus,

$$\lambda_i(W) := \frac{N_i}{\text{Area}(W)}, \tag{4.5}$$

for $i \in \{\text{Tu, Tc, Ma, Ot}\}$. We will write the statistic λ_i when the context of the observation window is clear.

The density statistic λ_i is thus interpreted as the averaged amount of cells of phenotype i per unit area. The area of the observation window for the Marked Poisson Point Process in Figure 4.1 is equal to 159762.9 square micronmeters. This results in, for example, that the density statistic for the Tumor cells is approximately 0.006, interpreted as an average of 0.006 Tumor cells per squared micronmeter.

Each distribution of cells of a phenotype is seen as one Poisson Point Process that together form the Marked Poisson Point Process. Consequently, the density statistic of each phenotype in the Marked Poisson Point Process can be summed to obtain the averaged amount of cells per unit area. This is based on the superstition theorem as mentioned in Section 3.1.

4.3 The Chi-squared statistic

The first spatial statistic we will define is the Chi-squared statistic. The Chi-squared statistic is based on the counts of cells in smaller rectangles of the observation window. The distribution of the counts of one phenotype over smaller rectangles, so called *quadratcounts*, gives insight how the cells are distributed over the whole observation window.

For each Poisson Point Process, the density is calculated. Recalling the definition of the density, which is the average amount of counts per square unit. Suppose that the quadrats are one square unit each. Then we expect in each quadrat as much counts as the density of the Poisson Point Process. Consequently, the observed quadratcounts are noted for each quadrat. The Chi-squared statistic is based on these expected and observed counts.

Let X be a Marked Poisson Point Process and observation window W with area Area(W) in square units. Let Q be a partition of the observation window with indices $k, l = 1, 2, \ldots, 5$, thus $\bigcup_{k=1,l=1}^{5} Q_{kl} = W$, where Q_{kl} is a rectangle with area Area(Q_{kl}). Thus, we divide the area of the observation window in 5×5 rectangles of equal size.

We compute $\lambda_i = N_i(W)/\text{Area}(W)$ as the density statistic of phenotype i and define $q_{kl} := N_i(Q_{kl})/\text{Area}(Q_{kl})$ as the observed counts per unit area in Q_{kl}.

Then the *Chi-square statistic* of phenotype i is defined as

$$\chi_i^2 := \begin{cases} \sum_{k,l=1}^{5} \frac{(q_{kl}-\lambda_i)^2}{\lambda_i}, & \text{when } N_i > 0, \\ \text{NA}, & \text{otherwise,} \end{cases} \tag{4.6}$$

for $i \in \{\text{Tu, Tc, Ma, Ot}\}$.

The Chi-Squared statistic is commonly used to test Poisson Point Processes on homogeneity. The Chi-Squared statistic is strictly positive, a value close to zero suggests the Poisson Point Process is homogeneous and a large value suggests the Poisson Point Process is inhomogeneous.

4.3.1 The Normalized Chi-Squared statistic

From the definition of the Chi-Squared statistic, one can deduce that its magnitude is dependent on the total amount of counts in the observation window. Therefore, we will introduce a normalized version of the Chi-Squared statistic.

Let X be a Marked Poisson Point Process in the observation window W, the Chi-Squared statistic χ_i^2 and the density statistic λ_i of X. Then the *Normalized Chi-Squared statistic* $\widetilde{\chi}_i^2$ of phenotype i is defined as

$$\widetilde{\chi}_i^2 := \begin{cases} \frac{\chi_i^2}{N_i}, & \text{when } N_i > 0, \\ \text{NA}, & \text{otherwise} \end{cases} \tag{4.7}$$

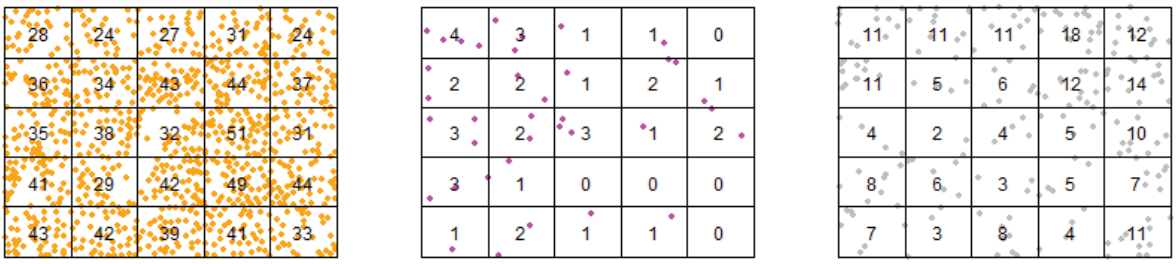
for $i \in \{\text{Tu, Tc, Ma, Ot}\}$.

The claim for this statistic is that it is independent of the total amount of counts in the observation window W, which is shown below

$$\begin{aligned} \widetilde{\chi}_i^2 &= \frac{\chi_i^2}{N_i}, \\ &= \sum_{k,l=1}^{5} \frac{(q_{kl}-\lambda_i)^2}{\lambda_i N_i}, \\ &= \frac{1}{\text{Area}(W)} \sum_{k,l=1}^{5} \frac{(q_{kl}-\lambda_i)^2}{\lambda_i^2}, \qquad (4.8) \\ &= \frac{1}{\text{Area}(W)} \sum_{k,l=1}^{5} \left(\frac{q_{kl}-\lambda_i}{\lambda_i}\right)^2, \\ &= \frac{1}{\text{Area}(W)} \sum_{k,l=1}^{5} \left(\frac{q_{kl}}{\lambda_i} - 1\right)^2, \\ &= \frac{1}{\text{Area}(W)} \sum_{k,l=1}^{5} \left(\frac{N_i(Q_{kl})}{N_i(W)} \frac{\text{Area}(W)}{\text{Area}(Q_{kl})} - 1\right)^2, \qquad (4.9) \end{aligned}$$

where Equation (4.8) resulted from plugging in N_i from the definition of the density. The sum in Equation (4.9) is independent of the total amount of counts in the observation window. Consequently, the normalized Chi-squared statistic is independent of the total amount of counts in the observation window and the claim has been proven.

An example of the application of the two Chi-Squared statistics can be seen in Figure 4.3. For each phenotype in the Marked Poisson Point Process the cells of that phenotype are counted in the 25 quadrats: q_{kl} for $k, l = 1, \ldots, 5$. After that, the Chi-Squared and the normalized Chi-Squared statistic are calculated, as all parameters needed for the calculation are already known. This results in a Chi-Squared and a normalized Chi-Squared statistic for the Tumors $\chi^2_{\text{Tu}} = 35.5$ and $\widetilde{\chi}^2_{\text{Tu}} = 0.039$, respectively. The Chi-Squared and the normalized Chi-Squared statistic for the Macrophages are $\chi^2_{\text{Ma}} = 20.4$ and $\widetilde{\chi}^2_{\text{Ma}} = 0.552$, and for the Others these are $\chi^2_{\text{Ot}} = 49.0$ and $\widetilde{\chi}^2_{\text{Ot}} = 0.247$



(a) Quadratcounts of X_{Tu} (b) Quadratcounts of X_{Ma} (c) Quadratcounts of X_{Ot}

Figure 4.3: Quadratcounts for the Marked Poisson Point Process in Figure 4.1.

4.4 The Pairwise Distance statistics

The statistics in this section are defined by calculating pairwise distances between cells of specific phenotypes.

In this section we will be subsetting matrices. We will indicate matrices in **bold**. Subsetting of matrices is indicated with the square brackets $[\cdot, \cdot]$, with subsets replaced for the dot $(\cdot)$.

Next we will define specific subsets of matrices. Let X be a Marked Poisson Point Process, with the total amount of counts of cells N_T. We define $\mathbf{D}$ to be the $N_T \times N_T$ distance matrix with the pairwise Euclidean distances, so $d^{kl} := d(x^k, x^l)$ from cell k to cell l.

Let $\mathbf{D}[X_i, X_j]$ be the submatrix of distances from cells with phenotype i to cells with phenotype j, for $i, j \in \{\text{Tu, Tc, Ma, Ot}\}$. The Distance matrix $\mathbf{D}[X_i, X_j]$ has dimensions $N_i \times N_j$ with pairwise Euclidean distances $d(x_i^k, x_j^l)$ from cell k with phenotype i to cell l with phenotype j.

The diagonal of $\mathbf{D}$ (and consequently the diagonal of $\mathbf{D}[X_i, X_i]$ for any given i) consists of all zeros. In the calculation of the following statistics, these zeros should be omitted.

Therefore we will write operators that omit zero's with the subscript greater than zero, for example $\underset{>0}{mean}(v) := mean(\{v_k | v_k > 0, k = 1, \ldots, n\})$ for any vector $v \in \mathbb{R}^n$.

4.4.1 The Median (Minimal) Distance statistics

Let $\mathbf{D}[X_i, X_j]$ be the submatrix of distances from cells with phenotype i to cells with phenotype j, for $i, j \in \{\text{Tu, Tc, Ma, Ot}\}$. The *Median Distance statistic*, abbreviated as the MD-statistic, for the phenotype pair (i, j) is defined as

$$\text{M-D}^{ij} := \underset{>0}{med}\left(\mathbf{D}[X_i, X_j]\right). \tag{4.10}$$

For the following statistic, we define the operator $row\min(\mathbf{D})$ as the minimum of each row vector in matrix D, with the extension of the operator to $\underset{>0}{row\min}(\mathbf{D})$ defined earlier.

The *Median Minimal Distance statistic* for the phenotype pair (i, j) is computed as

$$\text{MM-D}^{ij} = med\left[\underset{>0}{row\min}\left(\mathbf{D}[X_i, X_j]\right)\right]. \tag{4.11}$$

Replacing the Median operator with the Median Absolute Deviation (or *MAD*) operator in Equation (4.10) and Equation (4.11) defines the *MAD Distance statistic* and the *MAD Minimal Distance statistic*. Thus,

$$\text{MAD-D}^{ij} = \underset{>0}{mad}\left(\mathbf{D}[X_i, X_j]\right) \tag{4.12}$$

and similarly

$$\text{MAD-MD}^{ij} = mad\left[\underset{>0}{row\min}\left(\mathbf{D}[X_i, X_j]\right)\right]. \tag{4.13}$$

4.4.2 The Spatial Score statistic

In this subsection we define a statistic, the *Spatial Score*. This statistic relates to a triplet of cells of specific phenotype and was first proposed in Schurch et al. (2019). As the latter is still unpublished, our formulation might deviate from the original definition.

Let X be a Marked Poisson Point Process as usual, and X_{Tu} be the set of Tumor cells and x^k_{Tu} be Tumor cell k. Then we are looking for a triplet of cells $(x^k_{\text{Tu}}, x^{k,min}_{\text{Tc}}, x^{k,min}_{\text{Ma}})$, such that $x^{k,min}_{\text{Tc}}$ is the closest Tcell to x^k_{Tu} and $x^{k,min}_{\text{Ma}}$ is the closest Macrophage cell to $x^{k,min}_{\text{Tc}}$.

We define the spatial score for the k-th Tumor cell x^k_{Tu} as follows

$$\text{SS}(x^k_{\text{Tu}}) := \begin{cases} d(x^k_{\text{Tu}}, x^{k,min}_{\text{Tc}})/d(x^{k,min}_{\text{Tc}}, x^{k,min}_{\text{Ma}}) & \text{when } N_{\text{Tc}} > 0 \text{ and } N_{\text{Ma}} > 0, \\ \text{NA}, & \text{elsewhere.} \end{cases} \tag{4.14}$$

for $k = 1, \ldots, N_{\text{Tu}}$ and write

$$\text{SS}(X_{\text{Tu}}) = \{\text{SS}(x^k_{\text{Tu}}) \mid k = 1, \ldots, N_{\text{Tu}}\} \tag{4.15}$$

for the vector of spatial scores of Tumor cells.

The *Median Spatial Score* is then defined as the median of the spatial scores of Tumor cells,

$$\text{MED-SS} = med\left(\text{SS}(X_{\text{Tu}})\right). \tag{4.16}$$

Consequently, the *Median Absolute Deviation Spatial Score* is defined by replacing the Median operator by the Median Absolute Deviation, thus,

$$\text{MAD-SS} = mad\left(\text{SS}(X_{\text{Tu}})\right). \tag{4.17}$$

Thus, the statistic is only defined when there are Tcells and Macrophages to count. Assumed is that there are always Tumor cells available.

4.5 Cumulative distribution functions as statistics

In this section we define some cumulative distributions as statistics for closeness between cells with different phenotypes. Each cumulative distribution is defined as functions of distance.

For calculation of the features we define the vector $\overrightarrow{r} := \{1,2,3,4,5,\ 10,20,30,40,50\}$ containing radii in μm. For each radius r in $\overrightarrow{r}$ we will calculate the according function value for the cumulative distribution function and derive the value of the feature for that radius.

The following section and statistics are derived from Baddeley et al. (2015).

4.5.1 The Empty Space function

The Empty Space function quantifies how much empty space is in a Marked Poisson Point Process.

Let X be a Marked Poisson Point Process, with observation window W. Let o be an arbitrary coordinate in the observation window, not necessary coinciding with a point in X. The Empty Space function is then defined as

$$F_i(r) := \mathbb{P}(d(o, X_i) \leq r), \tag{4.18}$$

for $i \in \{\text{Tu, Tc, Ma, Ot}\}$. The Euclidean Distance is taken in Equation (4.18), such that it can be rewritten as

$$F_i(r) = \mathbb{P}(N_i(b(o, r)) > 0), \tag{4.19}$$

where $b(o, r)$ is the ball with radius r and origin o and $N_i(B)$ the function that counts the amount of cells of phenotype i in subset B. Thus, the Empty Space function is the probability of finding at least one cell of phenotype i in the ball with radius r and origin o.

For the calculation of the estimator, a grid of locations o_k, $k = 1, \ldots, m$ is generated in the observation window W. Then the Empty Space function has an estimator denoted by $F_i^*(r)$, given by the following equation

$$F_i^*(r) := \frac{1}{m} \sum_{k=1}^{m} \mathbb{1}(d(o_k, X_i) \leq r). \tag{4.20}$$

It has been shown in Baddeley and Gill (1997) that the estimator in Equation (4.20) is biased towards the edges of the observation window. Therefore, weights $e(o, r)$ are introduced such that the weighted estimator $\widehat{F}_i$ is unbiased.

For the computations of the features, we will use the *Kaplan-Meier correction*, denoted by e_{km}. The formal formulation goes beyond the scope of this thesis and can be found in Baddeley and Gill (1997).

The estimator $\widehat{F}_i(r)$ is then as follows

$$\widehat{F}_i(r) = \frac{1}{m} \sum_{k=1}^{m} \mathbb{1}(d(o_k, X_i) \leq r) e_{\text{km}}. \tag{4.21}$$

Suppose X_j is an homogeneous Poisson Point Process, then the theoretical value of the Nearest Neighbour function F_i^{theo} is a function of r,

$$F_i^{theo}(r) = 1 - e^{-\lambda_j \pi r^2}, \tag{4.22}$$

for phenotype pair (i, j).

Finally, we explain the procedure for the definition of the features. This is done as follows:

1. Firstly, the Marked Poisson Point Process X is split for each phenotype i, resulting in the Poisson Point Process X_i.
2. Secondly, the theoretical values F_i^{theo} and estimator values $\widehat{F}_i$ of the Empty Space function are computed for a finely spaced grid of r. The calculation of the estimator values includes the Kaplan-Meier border correction.
3. For each radius $r \in \overrightarrow{r}$ the values of $\widehat{F}_i(r)$ and $F_i^{theo}(r)$ are linearly interpolated (or extrapolated if necessary), using the finely spaced grid of r mentioned in the previous step.

The value of the feature *centered Empty Space statistic for phenotype i at r* is then defined as the difference of $\widehat{F}_i(r)$ and $F_i^{theo}(r)$, thus

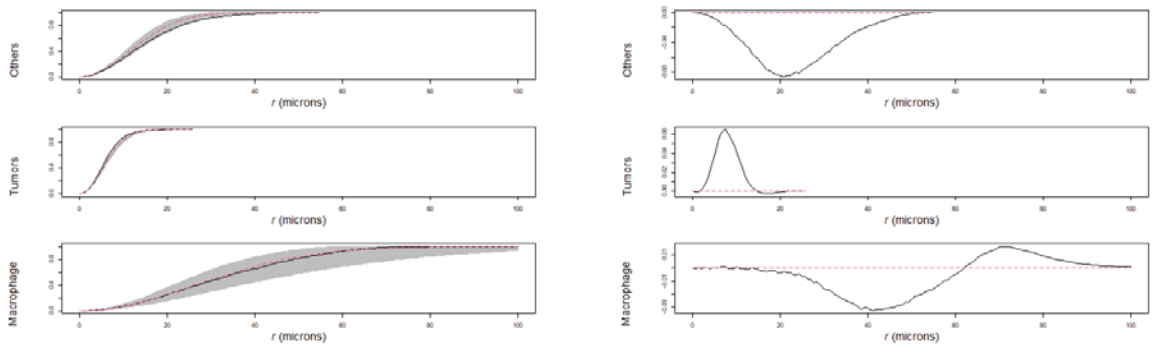
$$\widehat{F}_i(r) - F_i^{theo}(r).. \tag{4.23}$$

The sign and magnitude of the centered statistic described above can indicate whether the hypothesis of X_j being a homogeneous pattern can be accepted or not, for a significance level of 0.05. Therefore, an acceptance interval is generated for each r, centered around the theoretical value of the statistic. When the statistic value of the observed

pattern falls outside this acceptance interval, the hypothesis can be rejected. The following feature, and features constructed accordingly in the next sections, use the width of *acceptance interval* of the theoretical value of the statistic at the to be calculated r value, denoted with the σ-function. So for example, the width of the acceptance interval of the theoretical Empty Space function F_i^{theo} for phenotype i is denoted as $\sigma\left(F_i^{theo}(r)\right)$. The formal definition of the acceptance interval can be found in Baddeley et al. (2015).

The value of the feature *normalized Empty Space statistic for phenotype i at r* is defined as the centered Empty Space statistic in Equation (4.23) normalized by the width of the acceptance interval of F_i^{theo} at r, thus

$$\frac{\widehat{F}_i(r) - F_i^{theo}(r)}{\sigma\left(F_i^{theo}(r)\right)}. \tag{4.24}$$



(a) The Empty Space function $\widehat{F}_i(r)$ in black, $F_i^{theo}(r)$ in striped red and acceptance interval $\sigma\left(F_i^{theo}(r)\right)$ in grey.

(b) The Centered Empty Space function, for phenotype i, in black and the function identical to zero in striped red, for reference.

Figure 4.4: Application of the Empty Space function F_i on the Marked Poisson Point Process in Figure 4.1.

4.5.2 The Nearest Neighbour function

The Nearest Neighbour function is closely related to the Empty Space function.

Let X be a Marked Poisson Point Process, with observation window W. Let $x \in X_i$ be an arbitrary cell with phenotype i, for $i \in \{\text{Tu, Tc, Ma, Ot}\}$. Then the Nearest Neighbour function for the pair (i, j) is defined as

$$G_{ij}(r) := \mathbb{P}(d(x, X_j) \leq r \mid x \in X_i), \tag{4.25}$$

for $i, j \in \{\text{Tu, Tc, Ma, Ot}\}$. Written in terms of the function of the amount of cells of phenotype j, one obtains

$$G_{ij}(r) = \mathbb{P}(N_j(b(x, r)) > 0 \mid x \in X_i), \tag{4.26}$$

with $b(x, r)$ the ball with radius r and origin x.

Thus, the Nearest Neighbour function thus computes the probability of finding at least one cell with phenotype j less than a distance of r to cell with phenotype i.

Note that the Empty Space function is defined for a single phenotype i, while the Nearest Neighbour function is defined for the phenotype pair (i, j). Additionally, phenotype i can be paired up with more than one phenotype, as we will show next.

Let $i \in \{\text{Tu, Tc, Ma, Ot}\}$. One statistic of interest is the Nearest Neighbour function for i to the rest of the phenotypes. This is indicated with the dot $(\bullet)$ in the place of the j-index. Adapting the definition in Equation (4.25), we obtain the Nearest Neighbour function $G_{i\bullet}(r)$ for the phenotype pair $(i, \bullet)$

$$G_{i\bullet}(r) := \mathbb{P}(d(x, X_\bullet) \leq r \mid x \in X_i), \tag{4.27}$$

for $i \in \{\text{Tu, Tc, Ma, Ot}\}$.

The Nearest Neighbour function has an estimator denoted by $G^*_{ij}(r)$, given by the following equation

$$G^*_{ij}(r) := \frac{1}{N_i} \sum_{k=1}^{N_i} \mathbb{1}\{N_j(b(x_k, r)) > 0 \mid x_k \in X_i\}. \tag{4.28}$$

It has been shown in Baddeley and Gill (1997) that the estimator in Equation (4.28) is biased towards the edges of the observation window. Therefore, in the calculations, weights are introduced such that this corrected estimator $\widehat{G}_{ij}$ is unbiased.

For the computations of the features, we will use the *Kaplan-Meier correction*, denoted as e_{km}. The estimator $\widehat{G}_{ij}(r)$ is then as follows

$$\widehat{G}_{ij}(r) = \frac{1}{N_i} \sum_{k=1}^{N_i} \mathbb{1}\{N_j(b(x_k, r)) > 0 \mid x_k \in X_i\} e_{\text{km}} \tag{4.29}$$

and the estimator for $\widehat{G}_{i\bullet}(r)$ accordingly as

$$\widehat{G}_{i\bullet}(r) := \frac{1}{N_i} \sum_{k=1}^{N_i} \mathbb{1}\{N_\bullet(b(x_k, r)) > 0 \mid x_k \in X_i\} e_{\text{km}} \tag{4.30}$$

Suppose X_j is an homogeneous Poisson Point Process, then the theoretical value of the Nearest Neighbour function G^{theo}_{ij} is a function of r,

$$G^{theo}_{ij}(r) = 1 - e^{-\lambda_j \pi r^2}, \tag{4.31}$$

for phenotype pair (i, j). Note that Equation (4.31) only depends on phenotype j and not on phenotype i. For $X_\bullet$ the independence of phenotype i holds, such that

$$G^{theo}_{i\bullet}(r) = 1 - e^{-\lambda_\bullet \pi r^2}. \tag{4.32}$$

for phenotype pair $(i, \bullet)$.

Finally, we will explain the procedure for the definition of the features. This is done as follows:

1. Firstly, for every phenotype pair (i, j), including $(i, \bullet)$, the Marked Poisson Point Process is filtered of all other phenotypes. This Marked Poisson Point Process, say X_{ij}, only contains cells with phenotype i or j.
2. Secondly, the theoretical values G_{ij}^{theo} and estimator values $\widehat{G}_{ij}$ of the Nearest Neighbour function are computed for a finely spaced grid of r. The calculation of the estimator values includes the Kaplan-Meier border correction.
3. For each radius $r \in \overrightarrow{r}$ the values of $\widehat{G}_{ij}(r)$ and $G_{ij}^{theo}(r)$ are linearly interpolated (or extrapolated if necessary), using the finely spaced grid of r in the previous step.

The value of the feature *centered Nearest Neighbour statistic for the phenotype pair* (i, j) *at* r is then defined as the difference of $\widehat{G}_{ij}(r)$ and $G_{ij}^{theo}(r)$, thus

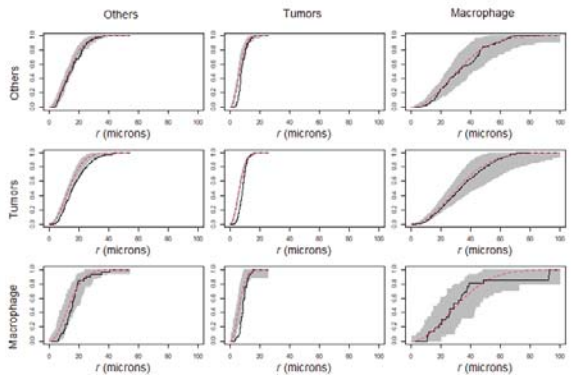
$$\widehat{G}_{ij}(r) - G_{ij}^{theo}(r). \tag{4.33}$$

Using the acceptance interval $\sigma\left(G_{ij}^{theo}(r)\right)$ the normalized Nearest Neighbour statistic is calculated.

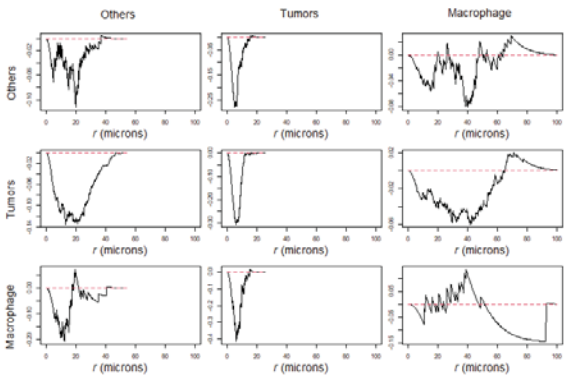
The value of the feature *normalized Nearest Neighbour statistic for phenotype pair* (i, j) *at* r is defined as the centered Nearest Neighbour statistic in Equation (4.33) normalized by the width of the acceptance interval of F_i^{theo} at r, thus

$$\frac{\widehat{G}_{ij}(r) - G_{ij}^{theo}(r)}{\sigma\left(G_{ij}^{theo}(r)\right)}. \tag{4.34}$$

For the features for the phenotype pair $(i, \bullet)$, the j is replaced for $\bullet$ in Equations (4.33) and (4.34).

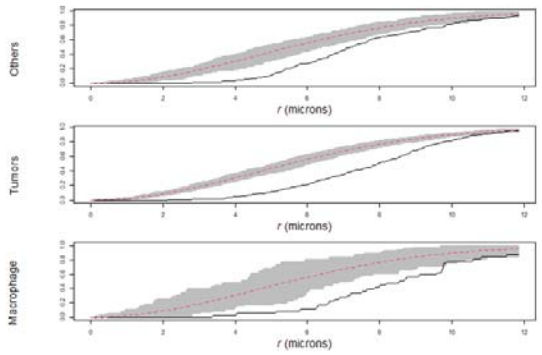


(a) The Nearest Neighbour function $\widehat{G}_{ij}(r)$ in black, $G_{ij}^{theo}(r)$ in striped red and acceptance interval $\sigma\left(G_{ij}^{theo}(r)\right)$ in grey.

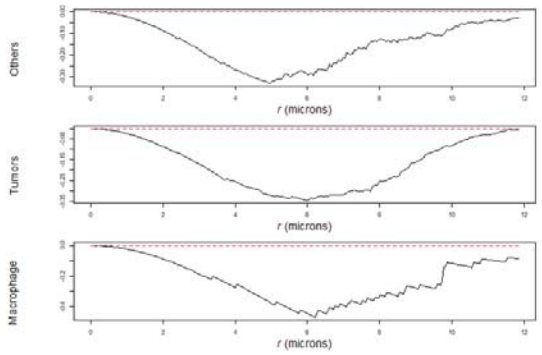


(b) The Centered Nearest Neighbour function, for phenotype pair (i, j), in black and the function identical to zero in striped red, for reference.

Figure 4.5: Application of the Nearest Neighbour function G_{ij} on the Marked Poisson Point Process in Figure 4.1.



(a) The Nearest Neighbour function $\widehat{G}_{i\bullet}(r)$ in black, $G_{i\bullet}^{theo}(r)$ in striped red and acceptance interval $\sigma\left(G_{i\bullet}^{theo}(r)\right)$ in grey.



(b) The Centered Nearest Neighbour function, for phenotype pair $(i, \bullet)$, in black and the function identical to zero in striped red, for reference.

Figure 4.6: Application of the Nearest Neighbour function $G_{i\bullet}$ on the Marked Poisson Point Process in Figure 4.1.

4.6 Ripley's functions

In this section, we discuss several Ripley's functions, named after the statistician Brian D. Ripley. Ripley's K-function is the fundamental version; Ripley's L-function and the Pair Correlation function are derived from the former.

4.6.1 Ripley's K-function

Let X be a Marked Poisson Point Process and denote the density statistic $\lambda_i(W)$ for $i \in \{\text{Tu, Tc, Ma, Ot}\}$. The Ripley's K-function for the phenotype pair (i, j) is defined as

$$K_{ij}(r) := \frac{1}{\lambda_j}\mathbb{E}[d(x, X_j) \leq r \mid x \in X_i]. \tag{4.35}$$

An alternative version of Equation (4.35) is when written in terms of the counting function of phenotype j, thus

$$K_{ij}(r) = \frac{1}{\lambda_j}\mathbb{E}[N_j(b(x, r)) > 0 \mid x \in X_i]. \tag{4.36}$$

In other words, Ripley's K-function is the expected amount of counts of phenotype j within a distance of r of an arbitrary cell of phenotype i, normalized by the density statistic of phenotype j.

Equation (4.35) can be adapted to produce the comparison of phenotype i to all other phenotypes. The definition for Ripley's K-function of the phenotype pair $(i, \bullet)$ is

$$K_{i\bullet}(r) := \frac{1}{\lambda_\bullet}\mathbb{E}[d(x, X_\bullet) \leq r \mid x \in X_i], \tag{4.37}$$

where $\lambda_\bullet$ is the density estimator for all but phenotype i.

Ripley's K-function has an estimator, denoted by $K^*_{ij}(r)$, given by the following equation

$$K^*_{ij}(r) := \frac{1}{\lambda_j} \sum_{k=1}^{N_i} \sum_{\substack{l=1 \\ l \neq k}}^{N_j} \mathbb{1}(d(x_k, x_l) \leq r), \tag{4.38}$$

where the double sum calculates the distant for each distinct pair of cells with phenotype i and j. For the case that $i = j$, the equation simplifies to Ripley's K-function for an unmarked Poisson Point Process.

For this estimator we will use the *isometric correction* to control the bias towards the edges. Calculation of this correction is done for every pair of cells with phenotype pair (i, j). Given the cell x_k of phenotype i and the cell x_l with phenotype j, the cell x_l lies on the boundary of a ball with origin x_k. The fraction of the length of the circumference of the ball inside observation window W and the circumference of the ball itself is calculated. The latter fraction is a probability and the isometric correction is the reciprocal of that probability, i.e.

$$e_{\text{iso}}(x_k, x_l, d_{kl}) := \frac{2\pi d_{kl}}{\ell\left(W \cap \partial b(x_k, d_{kl})\right)}, \tag{4.39}$$

where ℓ denotes the length and ∂ denotes the boundary. This results in the estimator $\widehat{K}_{ij}(r)$ for phenotype pair (i, j)

$$\widehat{K}_{ij}(r) := \frac{1}{\lambda_j} \sum_{k=1}^{N_i} \sum_{\substack{l=1 \\ l \neq k}}^{N_j} \mathbb{1}(d(x_k, x_l) \leq r) e_{\text{iso}}(x_k, x_l, d_{kl}), \tag{4.40}$$

and $\widehat{K}_{i\bullet}(r)$ for phenotype pair $(i, \bullet)$

$$\widehat{K}_{i\bullet}(r) = \frac{1}{\lambda_\bullet} \sum_{k=1}^{N_i} \sum_{\substack{l=1 \\ l \neq k}}^{N_\bullet} \mathbb{1}(d(x_k, x_l) \leq r) e_{\text{iso}}(x_k, x_l, d_{kl}). \tag{4.41}$$

Suppose X_j is an homogeneous Poisson Point Process, then the theoretical value of Ripley's K-function K^{theo}_{ij} is a function of r,

$$K^{theo}_{ij}(r) = \pi r^2, \tag{4.42}$$

for phenotype pair (i, j). For $X_\bullet$ the same holds, such that

$$K^{theo}_{i\bullet}(r) = \pi r^2, \tag{4.43}$$

for phenotype pair $(i, \bullet)$.

Finally, we will explain the procedure for the definition of the features. This is done as follows:

1. Firstly, for every phenotype pair (i, j), including $(i, \bullet)$, the Marked Poisson Point Process is filtered of all other phenotypes. This Marked Poisson Point Process, say X_{ij}, only contains cells with phenotype i or j.
2. Secondly, the theoretical values K_{ij}^{theo} and estimator values $\widehat{K}_{ij}$ of the Ripley's K-function are computed for a finely spaced grid of r. The calculation of the estimator values includes the isometric border correction.
3. For each radius $r \in \overrightarrow{r}$ the values of $\widehat{K}_{ij}(r)$ and $K_{ij}^{theo}(r)$ are linearly interpolated (or extrapolated if necessary), using the finely spaced grid of r in the previous step.

The value of the feature *centered Ripley's K-statistic for the phenotype pair* (i, j) *at* r is then defined as the difference of $\widehat{K}_{ij}(r)$ and $K_{ij}^{theo}(r)$, thus

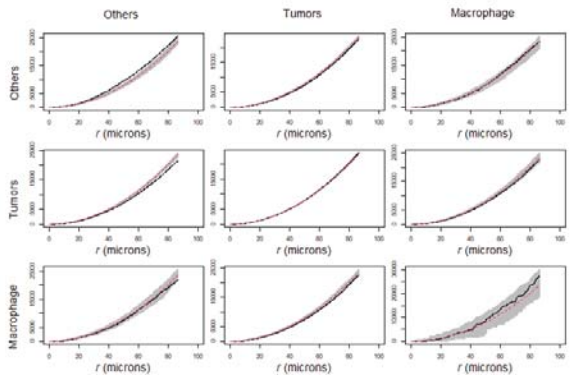
$$\widehat{K}_{ij}(r) - K_{ij}^{theo}(r). \tag{4.44}$$

Using the acceptance interval $\sigma\left(K_{ij}^{theo}(r)\right)$ the normalized Ripley's K-statistic is calculated.

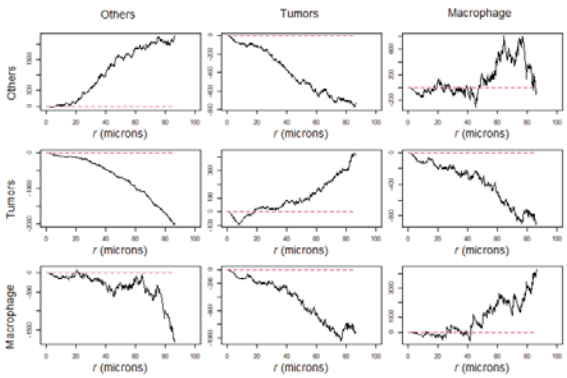
The value of the feature *normalized Ripley's K-statistic for phenotype pair* (i, j) *at* r is defined as the centered Ripley's K-statistic in Equation (4.44) normalized by the width of the acceptance interval of F_i^{theo} at r, thus

$$\frac{\widehat{K}_{ij}(r) - K_{ij}^{theo}(r)}{\sigma\left(K_{ij}^{theo}(r)\right)}. \tag{4.45}$$

For the features for the phenotype pair $(i, \bullet)$, the j is replaced for $\bullet$ in Equations (4.44) and (4.45).

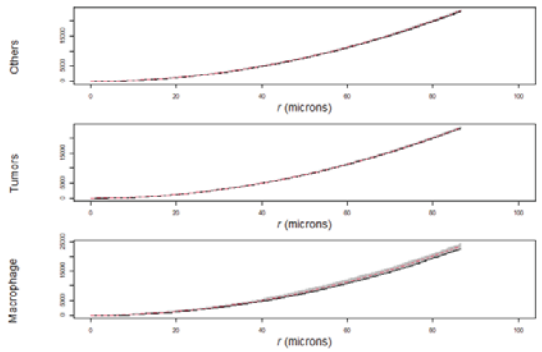


(a) Ripley's K-function $\widehat{K}_{ij}(r)$ in black, $K_{ij}^{theo}(r)$ in striped red and acceptance interval $\sigma\left(K_{ij}^{theo}(r)\right)$ in grey.

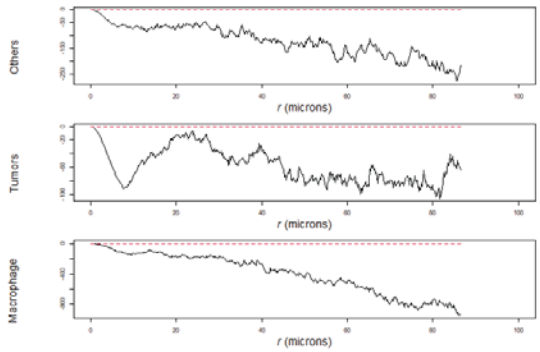


(b) Centered Ripley's K-function, for phenotype pair (i, j), in black and the function identical to zero in striped red, for reference.

Figure 4.7: Application of the Ripley's K-function K_{ij} on the Marked Poisson Point Process in Figure 4.1.



(a) Ripley's K-function $\widehat{K}_{i\bullet}(r)$ in black, $K_{i\bullet}^{theo}(r)$ in striped red and acceptance interval $\sigma\left(K_{i\bullet}^{theo}(r)\right)$ in grey.



(b) Centered Ripley's K-function, for phenotype pair $(i, \bullet)$, in black and the function identical to zero in striped red, for reference.

Figure 4.8: Application of the Ripley's K-function $K_{i\bullet}$ on the Marked Poisson Point Process in Figure 4.1.

4.6.2 Ripley's L-function

Ripley's L-function is a non-linear normalization of the Ripley's K-function.

Let X be a Marked Poisson Point Process and let $K_{ij}(r)$ denote Ripley's K-function for the phenotype pair (i, j). Then Ripley's L-function is the following

$$L_{ij}(r) := \sqrt{\frac{K_{ij}(r)}{\pi}}. \tag{4.46}$$

As earlier described, Ripley's K-function at r for a homogeneous Marked Poisson Point Process is πr^2. Consequently, Ripley's L-function of a homogeneous Marked Poisson Point Process result in a value equal to r. Thus,

$$L_{ij}^{theo}(r) = r, \tag{4.47}$$

for phenotype pair (i, j) and

$$L_{i\bullet}^{theo}(r) = r, \tag{4.48}$$

for phenotype pair $(i, \bullet)$.

As Ripley's L-function is derived from Ripley's K-function, the estimator with *isometric correction* is constructed accordingly. Thus, $\widehat{L}_{ij}(r)$ for the phenotype pair (i, j)

$$\widehat{L}_{ij}(r) := \sqrt{\frac{\widehat{K}_{ij}(r)}{\pi}}, \tag{4.49}$$

and $\widehat{L}_{i\bullet}(r)$ for the phenotype pair $(i, \bullet)$

$$\widehat{L}_{i\bullet}(r) := \sqrt{\frac{\widehat{K}_{i\bullet}(r)}{\pi}}. \tag{4.50}$$

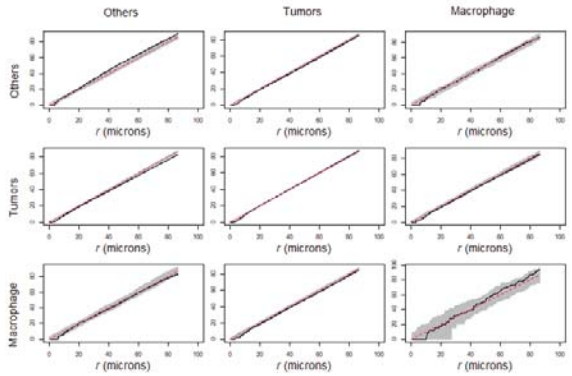
The procedure for the creation of features for Ripley's L-statistic is the same as the procedure for Ripley's K-statistic.

The value of the feature *centered Ripley's L-statistic for the phenotype pair* (i, j) *at* r is then defined as the difference of $\widehat{L}_{ij}(r)$ and $L_{ij}^{theo}(r)$, thus

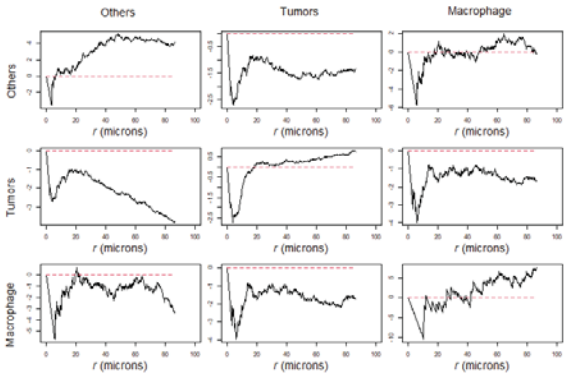
$$\widehat{L}_{ij}(r) - L_{ij}^{theo}(r). \tag{4.51}$$

The value of the feature *normalized Ripley's L-statistic for phenotype pair* (i, j) *at* r is defined as the centered Ripley's L-statistic in Equation (4.51) normalized by the width of the acceptance interval of L_i^{theo} at r, thus

$$\frac{\widehat{L}_{ij}(r) - L_{ij}^{theo}(r)}{\sigma\left(L_{ij}^{theo}(r)\right)}. \tag{4.52}$$

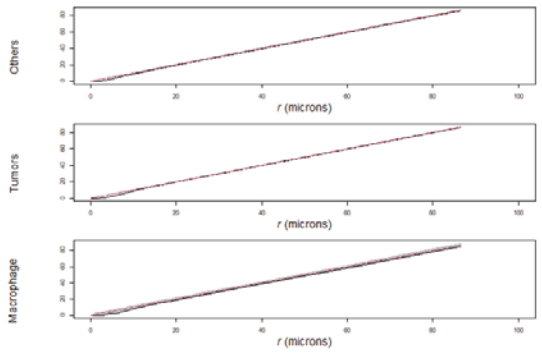


(a) Ripley's L-function $\widehat{L}_{ij}(r)$ in black, $L_{ij}^{theo}(r)$ in striped red and acceptance interval $\sigma\left(L_{ij}^{theo}(r)\right)$ in grey.

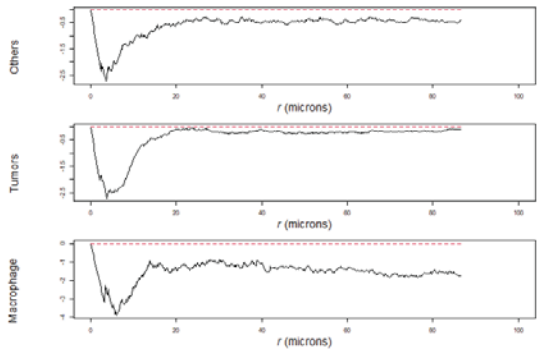


(b) Centered Ripley's L-function, for phenotype pair (i, j), in black and the function identical to zero in striped red, for reference.

Figure 4.9: Application of the Ripley's L-function L_{ij} on the Marked Poisson Point Process in Figure 4.1.



(a) Ripley's Ldot-function $\widehat{L}_{i\bullet}(r)$ in black, $L^{theo}_{i\bullet}(r)$ in striped red and acceptance interval $\sigma\left(L^{theo}_{i\bullet}(r)\right)$ in grey.



(b) Centered Ripley's L-function, for phenotype pair $(i, \bullet)$, in black and the function identical to zero in striped red, for reference.

Figure 4.10: Application of the Ripley's L-function $L_{i\bullet}$ on the Marked Poisson Point Process in Figure 4.1.

4.6.3 Pair Correlation function

Another function derived from Ripley's K-function is the Pair Correlation function.

Let X be a Marked Poisson Point Process and let $K_{ij}(r)$ denote Ripley's K-function for the phenotype pair (i, j). Then the Pair Correlation function for phenotype pair (i, j), denoted $g_{ij}(r)$, is the following

$$g_{ij}(r) := \frac{K'_{ij}(r)}{2\pi r}, \tag{4.53}$$

where $K'_{ij}(r)$ is the derivative with respect to r.

The derivative is calculated from the difference of the expected counts in the small circle between the circles with radius r and $r + h$. This difference is normalized by the expected count for a homogeneous Poisson Point Process. The derivation for an unmarked Poisson Point Process with density λ can be seen below,

$$\begin{aligned} g_h(r) &:= \frac{\lambda K(r+h) - \lambda K(r)}{\lambda\pi(r+h)^2 - \lambda\pi r^2}, \\ &= \frac{K(r+h) - K(r)}{2\pi rh + \pi h^2}, \\ &\approx \frac{K(r+h) - K(r)}{2\pi rh}, \\ &= \frac{K'(r)}{2\pi r}. \end{aligned} \tag{4.54}$$

Assuming the increment h is small in Equation (4.54), makes sure that $\pi h^2 \to 0$, such that $g_h(r)$ is approximately equal to the pair correlation function.

Suppose X_j is an homogeneous Marked Poisson Point process, then the theoretical value of the Pair Correlation function is identical one,

$$g_{ij}^{theo}(r) = 1. \tag{4.55}$$

The estimator of the Pair Correlation function replaces the indicator function in Equation (4.40) by the rescaled 'fixed-bandwidth' kernel $\kappa_h(r - d_{kl})$. This kernel is a smooth estimation of $K'_{ij}(r)$. Using the *isometric correction*, this leads to the pair correlation estimator $\widehat{g}_{ij}(r)$ as follows

$$\widehat{g}_{ij}(r) = \frac{A}{2\pi r N_i N_j} \sum_{k=1}^{N_i} \sum_{\substack{l=1 \\ l \neq k}}^{N_j} \kappa_h(r - d_{kl}) e_{\text{iso}}(x_k, x_l, d_{kl}). \tag{4.56}$$

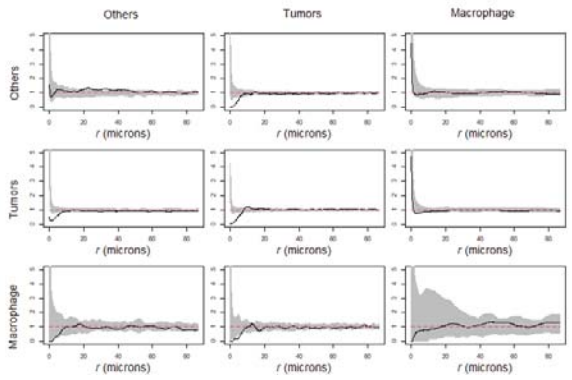
The procedure of the creation of features for the Pair Correlation function is the same as the procedure for Ripley's K-statistic.

The value of the feature *centered Pair Correlation function for the phenotype pair* (i, j) *at* r is then defined as the difference of $\widehat{g}_{ij}(r)$ and $g_{ij}^{theo}(r)$, thus

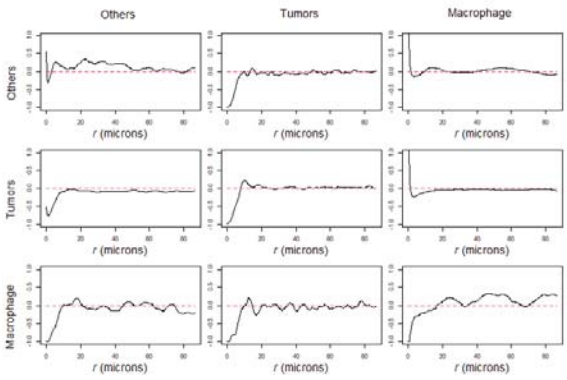
$$\widehat{g}_{ij}(r) - g_{ij}^{theo}(r). \tag{4.57}$$

The value of the feature *normalized Pair Correlation function for phenotype pair* (i, j) *at* r is defined as the centered Pair Correlation function in Equation (4.57) normalized by the width of the acceptance interval of g_{ij}^{theo} at r, thus

$$\frac{\widehat{g}_{ij}(r) - g_{ij}^{theo}(r)}{\sigma\left(g_{ij}^{theo}(r)\right)}. \tag{4.58}$$



(a) The Pair Correlation function $\widehat{g}_{ij}(r)$ in black, $g_{ij}^{theo}(r)$ in striped red and acceptance interval $\sigma\left(g_{ij}^{theo}(r)\right)$ in grey.



(b) The Centered Pair Correlation function, for phenotype pair (i, j), in black and the function identical to zero in striped red, for reference.

Figure 4.11: Application of the Pair Correlation function g_{ij} on the Marked Poisson Point Process in Figure 4.1.

5 Machine Learning

In this chapter we describe the Machine Learning tools that were used in this project.

5.1 Regression model

5.1.1 Logistic regression

A logistic regression model is a *Generalized Linear Model* for outcome data that is binary. Such a logistic regression model is defined as follows.

Let $n \in \mathbb{N}$ and $\mathbf{Y} \in \{0,1\}^n$ denote the response data random vector with realization $y := (y_1, \ldots, y_n)$. Let $\mathbf{X} \in \mathbb{R}^{n \times p}$ denote the high-dimensional ($p \gg n$) covariate matrix. Vector $x_i := (x_{i1}, \ldots, x_{ip})$ is a realization of covariate vector $\mathbf{X}_i$, for $i = 1, \ldots, n$.

The model,

$$\mathbf{Y}_i | \mathbf{X}_i, \beta \sim Bin(1, \mu_i), \tag{5.1}$$

$$\eta_i = x_i^{\mathrm{T}} \beta, \tag{5.2}$$

$$\eta_i := g(\mu_i) = \ln\left(\frac{\mu_i}{1-\mu_i}\right), \tag{5.3}$$

for $i = 1, \ldots, n$ is called the *logistic regression model* with *regression parameter* $\beta := (\beta_0, \ldots, \beta_p)$.

In a logistic regression model the response data $\mathbf{Y}_i$ is binary. Therefore, it is assumed that $\mathbf{Y}_i$, with given regression parameter β and the covariate random vector $\mathbf{X}_i$, is Binomial distributed with known size parameter $n_i = 1$ and unknown success parameter μ_i. This is formulated in Equation (5.1). Recall that consequently

$$\mu_i := \mathbb{P}(Y_i = 1 \mid X = x_i) = \sum_{k=0}^{1} k \cdot \mathbb{P}(Y_i = k \mid X = x_i) = \mathbb{E}(Y_i \mid X = x_i). \tag{5.4}$$

From the former Equation and the *Law of Full Probability*

$$\mathbb{P}(Y_i = 1 \mid X = x_i) = 1 - \mathbb{P}(Y_i = 0 \mid X = x_i), \tag{5.5}$$

it follows that in a logistic regression model the *log-odds* of μ_i, denoted as π_i, are in linear relationship with the covariates, i.e.

$$\ln(\pi_i) := \ln\left(\frac{\mu_i}{1-\mu_i}\right) = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}. \tag{5.6}$$

Note that in the covariate vector x_i, x_{i0} for all i has been set to one to model the *intercept* β_0 in the regression parameter β.

Simplifying Equation (5.6) for μ_i, the formula for $\mathbb{P}(Y_i = 1 \mid X = x_i)$ is then the following identity

$$\mathbb{P}(Y_i = 1 \mid X = x_i) = \frac{1}{1 + e^{-(\beta_0 + \sum_{j=1}^{p} \beta_j x_j)}}, \tag{5.7}$$

and the formulation of $\mathbb{P}(Y_i = 0 \mid X = x_i)$ is derived from Equation (5.5).

The function on the right hand side in Equation (5.7) is called the *logistic function*. The logistic function has the following properties:

1. The logistic function has a domain equal to $\mathbb{R}$.
2. The logistic function has a range equal to $(0, 1)$.

This is shown for the one dimensional logistic function in Figure 5.1.

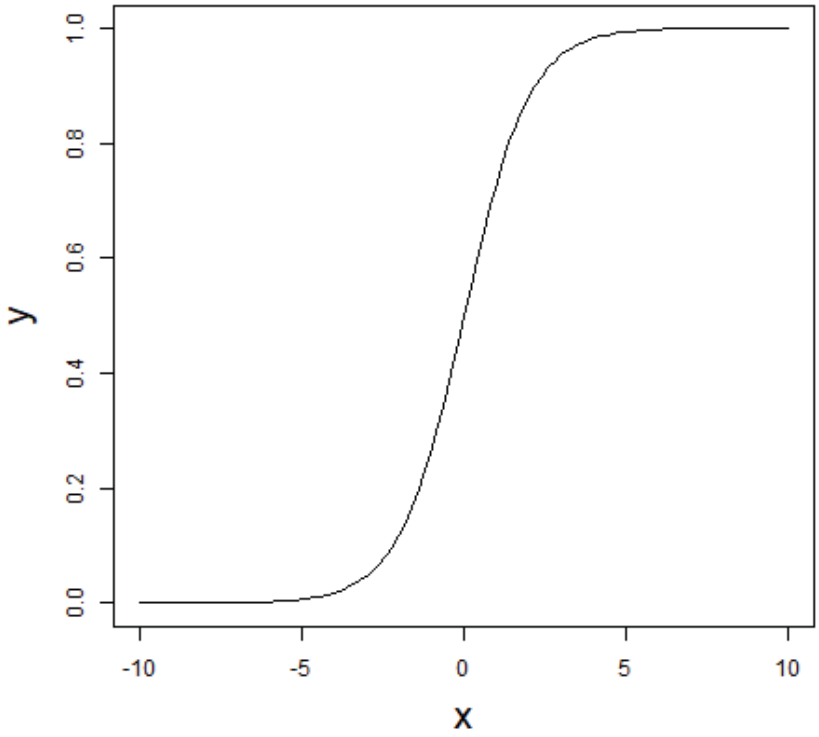


Figure 5.1: Form of the one-dimensional logistic function $y(x) = \frac{1}{1+e^{-x}}$. The domain of the logistic function is $\mathbb{R}$ and the range is $(0, 1)$.

These properties show that the logistic function actually qualifies for representing a probability function. Therefore, the assumption of the linear relationship of log-odds with the covariates as in Equation (5.6) is justified.

The unknown regression parameter β in the logistic regression model in Equation (5.6) should be estimated from the known data $\mathbf{X}$ and known response data $\mathbf{Y}$. Before we give the estimator of the regression parameter, we define the argument of minima. The *argument of a minima* of a function $f : A \to B$ is defined

$$\arg\min_{x \in A} f(x) := \{x \mid \forall y \in A : f(x) \leq f(y)\}. \tag{5.8}$$

An estimator of β, denoted by $\hat{\beta}$, is found through the minimization problem defined below

$$\hat{\beta} := \arg\min_{\beta \in \mathbb{R}^{p+1}} \ln(\pi(\mathbf{Y} \mid \mathbf{X}, \beta)). \tag{5.9}$$

The function to minimize is called the *objective function*.

Sometimes we want to obtain an estimator of β that possesses certain properties. One of such estimators is found in the Ridge logistic regression model, which we will discuss in the following subsection.

5.1.2 Ridge logistic regression

The *Ridge logistic regression model* is a logistic regression model in which the objective function in Equation (5.9) is extended to the following

$$\begin{aligned} \hat{\beta} &= \arg\min_{\beta} \left\{ \ln(\pi(\mathbf{Y} \mid \mathbf{X}, \beta) + \frac{\lambda}{2} \|\beta\|_2^2 \right\}, \\ &= \arg\min_{\beta} \left\{ \ln(\pi(\mathbf{Y} \mid \mathbf{X}, \beta) + \frac{\lambda}{2} \sum_{k=1}^{p} \beta_k^2 \right\}, \end{aligned} \tag{5.10}$$

for $\lambda > 0$.

The term $\frac{\lambda}{2}\|\beta\|_2^2$ is called the *regularization term* with *regularizer parameter* λ and *penalty function* $\|\beta\|_2^2$.

The goal of the regularization term is to put a penalty on the parameter β. The Euclidean norm forces the parameter β to attain small values while the regulizer parameter λ quantifies the penalty for attaining those values. For small λ, the objective function will allow β to attain bigger values across all its parameters, while a large value of λ forces all parameters in β to attain small, non-zero values.

5.1.3 ECPC

An extension of the Ridge logistic regression model is the method *Empirical bayes Co-data learnt Prediction and Covariate selection*, abbreviated as *ECPC*. For the means of this project, we will shortly describe the model extension. For a detailed description, see van Nee et al. (2020b).

In ECPC, complementary data (co-data) on covariates is taken into account in the estimation of the parameter β. This co-data is incorporated in the model by the means

of *groups* and *groupings*. These co-data is a form of prior information, and according to these co-data groups of covariates and groupings are defined. Then, instead of one regulizer parameter λ for β like in Equation 5.15, in ECPC, there is a regulizer parameter per group. This results in regularisation of parameters in β corresponding to the groups, potentially improving the prediction of the response data.

Let $\mathbf{Y} \in \mathbb{R}^n$ denote the response data vector, let $\mathbf{X} \in \mathbb{R}^{n \times p}$ denote the high-dimensional ($p \gg n$) covariate matrix. For d = 1,. . . , D, let $Z^{(d)} \in \mathbb{R}^{p \times G^{(d)}}$ denote D different *co-data matrices* representing groupings of covariates, defined as follows

Definition 5.1. Define each **grouping** $\mathcal{G}^{(\mathbf{d})}$, $d = 1, \ldots, D$, as a collection of sets $\mathcal{G}_g^{(d)}$, called **groups**, of covariate indices in $\{1, \ldots, p\}$, such that each covariate belongs to at least one group:

$$\{1, \ldots, p\} = \bigcup_{\mathcal{G}_g^{(d)} \in \mathcal{G}^{(d)}} \mathcal{G}_g^{(d)}, \quad \forall d = 1, \ldots, D. \tag{5.11}$$

Denote the grouping size, i.e number of groups in each grouping, by $G^{(d)} := |\mathcal{G}^{(d)}|$, and denote the group size of group g in grouping d, i.e the number of covariates in that group by $G_g^{(d)} := |\mathcal{G}_g^{(d)}|$.

Note that the groups can be overlapping, as long as all covariates are included. This concept is formalized in the definition of the **co-data matrix** $Z^{(d)}$, for which we refer to van Nee et al. (2020b). The co-data matrix $Z^{(d)}$ is defined for each grouping. Each row $Z_k^{(d)}$, with elements in the interval $[0, 1]$, is used to pool the information from the groups in grouping d that k belongs to.

The method assigns a group weight to each group in the grouping, denoted by $\gamma^{(d)}$ and $\gamma^{(d)} \in \mathbb{R}_+^{G^{(d)}}$, modeling the relative importance of the groups in grouping d. The grouping weight $w^{(d)}$ with $w^{(d)} \in \mathbb{R}_+$ models the relative importance of grouping d.

In Figure 5.2 the before mentioned definitions are illustrated.

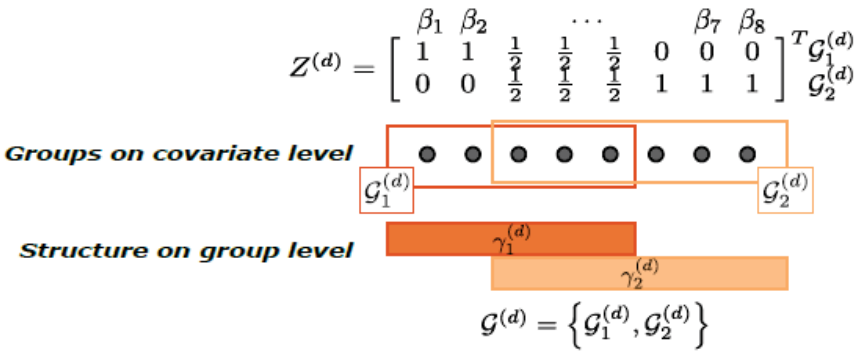


Figure 5.2: Illustration of the concepts of *groupings*, *groups* and the *co-data matrix* in ECPC. Original figure from van Nee et al. (2020b).

The method takes a Bayesian approach for the model, therefore a normal prior is imposed on β with global prior variance τ_{global}^2 and local $\tau_{k,local}^2$, as proposed in van de Wiel et al. (2016).

The model is then the following

$$\mathbf{Y}_i|\mathbf{X}_i, \beta \sim Bin(1, \mu_i), \tag{5.12}$$

$$\eta_i = x_i^{\mathrm{T}}\beta, \tag{5.13}$$

$$\eta_i := g(\mu_i) = \ln\left(\frac{\mu_i}{1-\mu_i}\right), \tag{5.14}$$

$$\beta_k \sim N(0, \tau^2_{global}\tau^2_{k,local}), \quad k = 1, \ldots, p, \tag{5.15}$$

$$\tau^2_{k,local} = \sum_{d=1}^{D} w^{(d)}\mathbf{Z}_k^{(d)}\gamma^{(d)}, \quad k = 1, \ldots, p. \tag{5.16}$$

The unknown model parameters are the regression parameter β and the prior parameters, called *hyperparameters*, $\left\{\tau^2_{global}, \gamma^{(1)}, \ldots, \gamma^{(D)}, w^{(1)}, \ldots, w^{(D)}\right\}$. The hyperparameters $\tau^2_{k,local}$ are known from the relation in Equation (5.16) and we introduce the notation $(\hat{\tau}_{local})^{-1} := (1/\hat{\tau}_{1,local}, \ldots, 1/\hat{\tau}_{p,local})$.

The minimization problem for β is then as follows

$$\hat{\beta} = \arg\min_{\beta}\left\{\ln(\pi(\mathbf{Y} \mid \mathbf{X}, \beta) + \frac{1}{\hat{\tau}^2_{global}}\|(\hat{\tau}_{local})^{-1}\beta\|_2^2\right\}, \tag{5.17}$$

$$= \arg\min_{\beta}\left\{\ln(\pi(\mathbf{Y} \mid \mathbf{X}, \beta) + \frac{1}{\hat{\tau}^2_{global}}\sum_{k=1}^{p}\frac{1}{\hat{\tau}^2_{k,local}}\beta_k^2\right\}. \tag{5.18}$$

As this is an ordinary ridge regression estimator with weighted penalty, existing solving methods can be used such as described in Friedman et al. (2010). In ECPC, hyperparameters are estimated to potentially improve the prediction. Nevertheless, estimation of these hyperparameters is beyond the scope of this project and therefore omitted. We refer to van Nee et al. (2020b) for the details.

5.2 Random Forests

5.2.1 Decision Trees and Random Forests

Random Forest is a Machine Learning method that is based on decision trees. *A decision tree* is a sequence of If/Else-questions concerning the covariates in a prediction model. Such a question is called *a test*.

At first, the whole data set is to our possession, defined as *the root* position in the decision tree. From all covariates, a single covariate is selected with a threshold, defining the test for the root, for example: *Is for the data point the count of the Macrophages greater than 140?*. For this example the covariate is 'the count of the Macrophages' and the threshold being 'greater than 140'.

What follows is a split of the root into two sets of data points, called *nodes*, separated according to the outcome of the test. Each node can be split with sequential tests in a recursive manner. This increases the so called *depth* of the decision tree. When the

maximal depth is reached for the decision tree, a class is assigned to each node. This way, an unclassified data point with corresponding known covariates can be classified by following the decision tree. The data point obtains the class of the last node of the decision tree. This is known as a 'classification' problem. Decision trees are also used for 'regression' problems, which will be the case for this project. In regression problems, the goal is to find thresholds for tests in the decision tree, such that the decision tree classifies the overall data set as 'correct' as possible. We will discuss how 'correct' can be defined in Subsection 5.3.2.

Decision trees are commonly used and have great intuition. In this project, the two sets of response data have been derived following a decision tree, as described in Section 2.1. More examples and strategies for creating decision trees can be found in Muller and Guido (2016).

Before touching the generation of a Random Forest model, we will need the introduction of a sampling method called *bootstrapping*. A *bootstrap sample* S^* of a data set S is a set of equal size as S, where elements of S^* are generated by sampling elements of S with replacement.

From a bootstrap sample of the data, a decision tree is generated, slightly different than described earlier. In this decision tree, for each splitting of the node, only a random subset of covariates are to be considered. Nevertheless, from this random subset of covariates, one covariate is selected and a threshold accordingly. This continues until some pre-defined depth has been reached, upon which the first decision tree is complete. Starting with a new bootstrap sample, the before mentioned process is repeated many times, generating independent, uncorrelated decision trees. A data point is then classified by either the rule of 'majority votes' or the rule of 'predicted probabilities'. The rule of majority votes means that a data point is classified as the class that was attached the most to the data point by all decision trees. The rule of predicted probabilities, does generally the same as the majority votes rule, but returns the number of majority votes normalized by the number of decision trees used.

Random Forest has an implementation where the relative importance of covariates can be determined during the running of the algorithm. Each covariate is assigned a *relative importance* value, as implemented by Breiman (2001). We will not discuss the actual computation here, for details see Breiman (2001), Liaw and Wiener (2002) or Muller and Guido (2016). The concept is intuitive though: High positive relative importance of a covariate signifies that the covariate contributed much to the prediction, while high negative relative importance signifies that the covariate contributed much when left out of the prediction. A relative importance close to zero signifies no contribution of the covariate.

5.2.2 CoRF

An extention of the Random Forest algorithm for high-dimensional data is the *co-data moderated Random Forest (CoRF)*, proposed in te Beest et al. (2017).

In CoRF, first a base Random Forest is performed. In this base Random Forest, each covariate has uniform sampling probability to be selected in defining a test. In each

decision tree, the selection of covariates is different. Therefore, the number of times each covariate was used is summed across all decision trees. High counts of usage across all decision trees, might indicate that that covariate is important for the model.

The co-data provided is in the form of groups consisting of covariates. Then, *group-specific probabilities* w_g are calculated following the definition

$$(\cdot)^+ := \max(\cdot, 0), \quad w_g := \left(\hat{p}_g^{sel} - p^0\right)^+,$$

where $\hat{p}_g^{sel}$ is the proportion of selected covariates from group g across all decision trees divided by the size of group g and $p^0 = 1/p$ is the expected value of $\hat{p}_g^{sel}$ when the group structure is uninformative. The group-specific probabilities are normalized to sum to one, to obtain the *sampling probabilities* $\hat{w}_g$. These sampling probabilities are used in a new Random Forest, replacing the uniform sampling probability.

From the refitted Random Forest, once again counting usage across all decision variables, covariate importance for the model is obtained.

5.3 Performance evaluation

5.3.1 Underfitting and Overfitting

At first thought, we want our model to perform well on the data set. But this is where the dangers of 'Underfitting' and 'Overfitting' lie.

The data set is split in two parts: The training data and the test data. The model is trained on the training data, meaning parameters are estimated. Then, the model with the estimated parameters is used to see how well it performs on the test data. We say the model should generalize to new data as accurately as possible, preventing underfitting and overfitting.

Underfitting is the term used when the model obtained after estimation of parameters is too simple. Commonly, this happens when too many estimated parameters are equal to zero, thus resulting in a small amount of covariates in the model. Consequence of an underfitting model is generally a low performance on both the training and the test data. Even when the underfitting model performs reasonably well on the training data, the variability of the data might not be wholly captured, resulting in a poor performance on the test data

Overfitting is the term used when the model obtained after estimation of the parameters fits the training data too perfect. This commonly happens when too many covariate parameters are used in the obtained model. Using many covariate parameters increases the biased towards the training data, such that the model performs worse on new data.

We encountered techniques to prevent underfitting and overfitting in the models discussed in the previous Chapter.

In the Logistic regression model, the regression parameter β was estimated through Equation (5.9). This model is likely to overfit when many covariates are selected. In the extension to the Ridge logistic regression model, the penalty term in Equation (5.10) restricts the magnitude of the parameters in β. An estimator of β that resulted in

an overfit of the Logistic regression model might not have been chosen in the Ridge Logistic regression model, due to this restriction. Continuing with the ECPC model, the estimator for β is found through Equation (5.17). In the latter, the regularizer term is extended even more, signifying more control over β.

In the Random Forest models, the bootstrapping method creates randomization of the data. Each decision tree is generated with random covariates selected, upon in the end, all decision trees are averaged. Both procedures make sure the result of the Random Forest is not biased towards the original data set.

Also the depth of the decision tree is chosen to prevent overfitting and underfitting. When the depth is too small, underfitting might occur, while when the depth is too large overfitting might occur.

Chosing a depth that is between small and large forces the model to make minor mistakes in the prediction of the training data, but resulting in an increase of prediction performance on the test data.

5.3.2 Sensitivity and Specificity

In Section 5.2, we described the models with their underlying assumptions. Different models might give different results, therefore we need measures to compare the results.

Such a comparison can be on a classification test of any kind, therefore we will do this on the example test in Subsection 5.2.1, *Is for the data point the count of the Macrophages greater than 140?*.

Suppose we have an data set S containing 200 data points, 100 data points of class one and 100 data points of class zero. The test above then splits the original data according to the test into two data sets S_1 and S_0. Each data point for which the test is True is put in S_1 and the others in S_0.

Ideally, if the test is informative, the test splits the data set such that all data points with the same class are in either S_1 or S_0, and vice versa. For classification, a new data point's class would be predicted as one if the test is True, and zero otherwise.

Although, what generally happens, is that both S_1 and S_0 contain data points of each class. The performance of the prediction model are derived from counting different sets of data points in each class.

As an example, let us define the root as a vector $S = (100, 100)$, with elements of the number of data points with response variable one and zero respectively. The test above splits S in nodes S_1 and S_0, with distribution $S_1 = (80, 5)$ and $S_0 = (20, 95)$.

From the above distribution, we conclude the following

- 175 data points are correctly classified, namely
  - 80 data points with class one, so-called *True-Positives* (TP).
  - 95 data points with class zero, so-called *True-Negatives* (TN).
- 25 data points are are incorrectly classified, namely
  - 5 data points with class one, so-called *False-Positives* (FP).
  - 20 data points with class zero, so-called *False-Negatives* (FN).

These values can be summarized in a *confusion matrix*, visualized for this example in Figure 5.3. When the test has classified False-Positives, we say that a *type-I error* has been made. Tests with False-Negatives are tests with *type-II error*. Making errors is a problem of the test, but what type of error is worse for the prediction differs per field. In some medical settings making a type-I error is more tolerated than making a type-II error.

To quantify making errors in prediction models, many different measures are derived from the values in the confusion matrix. We will discuss the *sensitivity*, *specificity* and the *accuracy* for the purpose of this project. The sensitivity and the specificity are commonly used in constructing the ROC-curve, which we will discuss in Subsection 5.3.3.

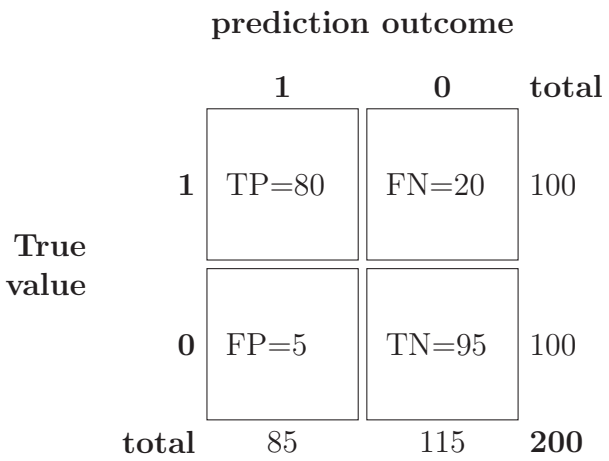


Figure 5.3: The confusion matrix for the example test described in the text.

Let $t \in \mathbb{R}$ be a threshold for a conducted test, $\mathrm{TP}(t)$ be the number of True-Positives and $FN(t)$ be the number of False-Negatives for the conducted test for threshold t. The *Sensitivity* or the *True Positive Rate (TPR)* of a test is a measure calculating the proportion of positives that were truly classified as positive. Therefore, the sensitivity TPR is defined as a function $TPR : \mathbb{R} \to [0, 1]$, where

$$TPR(t) := \frac{\mathrm{TP}(t)}{\mathrm{TP}(t) + \mathrm{FN}(t)}. \tag{5.19}$$

When it is clear from the context which test is conducted, we will drop the dependence of the measure on the threshold.

Let TN be the number of True-Negatives and FP be the number of False-Positives for the conducted test for threshold t. The *Specificity* or the *True Negative Rate (TNR)* of a test is a measure calculating the proportion of negatives that were correctly classified as negative. Thus, the specificity is defined as $TNR : \mathbb{R} \to [0, 1]$, where

$$TNR := \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}. \tag{5.20}$$

Let TP, TN, FN, FP be defined as above. The *Accuracy* of a test of threshold t is a measure calculating the proportion of the data set that is correctly classified. Thus, the accuracy is defined as $Accuracy : \mathbb{R} \to [0, 1]$, where

$$Accuracy := \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}}. \tag{5.21}$$

For the example, this means that the sensitivity for the test was $\frac{80}{80+5} \approx 0.94$, the specificity of the test was $\frac{95}{95+20} \approx 0.83$ and the accuracy of the test was $\frac{80+95}{80+95+20+5} \approx 0.86$.

The closer each measure to one, the better the test, where the ideal test would have all measures equal to one. Due to the definitions, an accuracy equal to one implies that both the sensitivity and the specificity are equal to one[1]. This is therefore an ideal test. In practice, ideal tests are not common. Consequently, a strategy should be provided for picking the best test.

The strategy for the test often depends on what type of error the test is allowed to make. Type-I errors are a result of False-Negatives, therefore, picking the test with the highest sensitivity decreases type-I errors. When we would like to reduce the amount of type-II errors, we should pick the test which results in the highest specificity.

5.3.3 ROC-curve and AUC

As described earlier, tests can be conducted with different thresholds. To compare these results for different thresholds the ROC-curve is plotted. First, the TNR and the TPR, so the specificity and the sensitivity respectively, are calculated for a domain of thresholds. For each threshold, the coordinate (TNR, TPR) is plotted in $[1, 0] \times [0, 1]$.

The ROC-curve is plotted by default with a diagonal line from corner to corner. This line indicates the test that classifies at random. The test performs better than random if the ROC-curve of the test is above the diagonal line. The ideal test would have a ROC-curve containing the coordinate $(1, 1)$, the closer the ROC-curve of the test to this coordinate, the better the test is.

The latter notion of visual inspection of the ROC-curve is quantified in the *Area under the curve* or AUC. The AUC is a common measure used to compare performance of prediction models, and calculates the area under the ROC-curve for a prediction model. The AUC ranges between zero and one, where an AUC of closer to one corresponds to a better prediction model. Intuitively, the prediction model that classifies at random has an AUC of 0.5.

The AUC of a ROC-curve is often used to compare performance of prediction models. Nevertheless, summarizing the ROC-curve in one measure results in a loss of information of the sensitivity and the specificity.

5.3.4 Cross-validation

As mentioned earlier, the data is divided in the training set and the test set. The dividing of the data into these sets is done at random. Commonly, the training set contains 80%

[1]This follows as $\text{FN} + \text{FP} = 0$ and the non-negativity of FN and FP implies that $\text{FN} = \text{FP} = 0$.

of the data set and the test set the leftover 20%. This can be problematic when data with one response variable only is assigned to either the training set or the test set. This creates bias in the prediction model. The prediction model would only train on the response variable available in the training set, and be unaware of the existence of the other response variable. Consequently, the measure of performance of the prediction model might not be representative. Definitely in unbalanced data sets where the counts of the response variables might not be uniform distributed, measure of performance is dependent on the training and test set division. This problem is solved by balancing the training and test set, such that the percentages of zeros and ones are equal in both sets.

In *k-fold cross-validation*, the prediction model is repeatedly trained on k different training sets. Each training set is generated from a different part of the data set, with complementary test set. An example of a 5-fold cross-validation splitting can be seen in Figure 5.4, originally from Muller and Guido (2016). The prediction model is trained and tested on each splitting, each splitting resulting in a performance measure, for example the AUC.

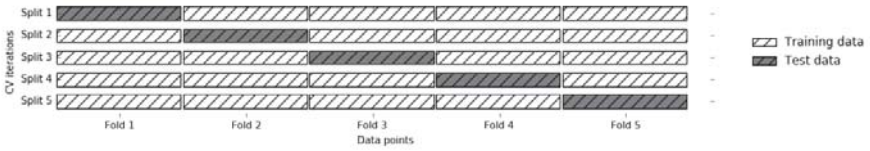


Figure 5.4: Splitting of the data set into 5-folds for cross validation. Figure originally from Muller and Guido (2016).

In this $k-$fold cross-validation, k AUCs are found. This vector of AUCs is then analyzed. Common is to calculate the mean and the standard deviation of the AUCs. Including statistics such as the 97.5% and the 2.5% quantile provides extra information on how stable the AUC is.

8 Discussion

In this project, there were some things that could be optimized for follow-up research.

First of all, the methods used in the scripts are optimized to simple phenotyping. We started off with complete phenotyping, but switched to simple phenotyping because simple phenotyping contained less unique phenotypes with more counts. The more complex spatial statistics, such as Ripley's functions, depend on having a reasonable number of counts to produce a reliable feature. In contrary, converting the patient MSIs from complete phenotyping to simple phenotyping removes possibly valuable information of the individual phenotypes in the complete phenotyping. It also goes against the exploitation of being able to identify many phenotypes through Vectra Imaging. Therefore, a possible solution to this, is to use less complex spatial statistics on the complete phenotyping. This would mean to exclude several statistics described in Section 4.5 and 4.6. As seen in the results of the comparison of the *simple spatial grouping*, there already is indication that this solution potentially improves prediction.

Secondly, the patient MSIs were picked on sight of the pathologist. This resulted in a variation in the number of obtained MSIs for each patient, for example: some patient had two MSIs while another patient had ten MSIs. To obtain a specific feature for one patient, we averaged the corresponding feature over all the MSIs available for that patient. The difference of number of MSIs influences on how representative the obtained feature is. Also important is that the method of averaging is known to be influenced by outliers. As adding an outlier has less impact on the average of a data set of increasing sample size, increasing the sample size results in a more representative average. Therefore, in follow-up research, it is recommended to obtain more or at least the same number of MSIs for each patient. Of course, this is an easy thing to suggest, although harder to apply as the biological tissue is very hard to come by. Surgeries to obtain more tissue would possibly take longer, putting unnecessary risk on patients. To prevent this, another method would be to take the median of the patient MSIs. The median is known to be less influenced by outliers, making it more suitable to summarize the features extracted from patient MSIs.

Initially, the idea was to have all patient MSIs to be used to generate features for prediction. In the process of reviewing the patient MSIs, we observed that the distribution of phenotypes for some patients was different over the MSIs from that patient. This was when we received the information that the pathologist obtained patient tissue on sight: The pathologist abducted patient tissue generally in the tumor region itself, and sometimes on the border region of the tumor.

Because generally the distribution of phenotypes was different for 'Tumor' MSIs than for 'Border' MSIs, this distinction would show in the obtained features. Consequently, taking averages over all these features obtained from the MSIs would give a non-reliable

feature for the patient. Therefore, we decided to label every MSI either 'Tumor' or 'Border', and treat each label as a different data set. Continuing with this in mind, we figured that we could construct three different prediction models:

1. A model that was solely based on patients with at least one 'Tumor' MSI. A feature for a patient is obtained by averaging the corresponding feature over the patient's 'Tumor' MSI(s),
2. A model that was solely based on patients with at least one 'Border' MSI. A feature for a patient is obtained by averaging the corresponding feature over the patient's 'Border' MSI(s),
3. A model based on patients with at least one 'Tumor' MSI and one 'Border' MSI. The same as above applies, but now a patient feature is generated for both the labels. This would result in at most two times more features in the model, compared to either model above.

Nevertheless, in the prediction model phase of the project, we chose to only include the analysis of the prediction model derived from the 'Tumor' MSI data set. We argued that, script-wise, the 'Border' MSI analysis was not any different than the 'Tumor' MSI analysis. We already planned on using different prediction methods for the analysis, as described in Section 5.1 and 5.2. Because of this reason, the results and analysis would already be complex on its own. Therefore we continued the project with only the 'Tumor' MSI data set. Consequently, follow-up research could include the analysis of the prediction model derived from the 'Border' MSI data set, and possibly the analysis of the prediction model derived from the 'Tumor'- and 'Border' MSI data sets.

In addition, a possible solution to the selection bias by the pathologist of the MSIs, is to extract features from the whole biopt. At the time of the start of the project, the analysis of the whole biopt was not possible. Meanwhile, recent increase in computational power has made this a viable solution. This would result in one MSI for one patient, which includes the 'Tumor' regions and 'Border' regions in whole. Consequently, more biological tissue is used for analysis. This could benefit the prediction because hypothetically more counts of the phenotypes in the complete phenotyping could be obtained. This could lead to more detailed results revolving the complete phenotypes. Another benefit would be that the more complex spatial statistics, as described in Section 4.5 and 4.6, could be used and possibly contribute as important features to prediction of survival.

For the extended regression models we choose a Ridge regression model. As explained in Section 5.1.2, this model controls the magnitude of the regression parameter β by forcing it to attain small non-zero values. But, for high-dimensional data, we do not expect all of the features to be important for prediction. Forcing these features of less importance to be included in the model seems counter-intuitive. Therefore, the performance of the prediction model could potentially benefit from the assumption that β is allowed to attain zero values. The latter is done in a LASSO regression model, popularized by Tibshirani (1996). Assuming that β can attain zero values, could benefit in the performance of the prediction model. Therefore, we suggest in follow-up research to experiment with a LASSO regression model.

9 Acknowledgements

I would to thank some people that contributed to this project.

First of all, Yongsoo Kim, Tim van de Brug and Marit Roemer, for their help as daily supervisors through the extended timeline of this project.

Thank you Yongsoo for your expert eye on the programming part of this project, and the early trust you have shown in me. I am thankful to be given the opportunity to work more with you at the Cancer Centrum Amsterdam.

I would like to thank Tim for your great feedback during this project and introducing me to the environment at the Department of Epidemiology & Data Science. It was wonderful to be able to work there, as were the brainstorms that we had that sparked my interest every time.

Thank you Marit Roemer for sharing the biomedical background with me and showing me biomedical techniques that I did not know existed.

I would like to show my appreciation to Daniella Berry for her contribution in the selection and labelling of the patient MSIs.

I would like to praise Mirrelijn van Nee for her excellent job on the ECPC-method and enlightening me on parts of the method that were not clear to me.

Lastly, my biggest gratitude goes to my parents and my girlfriend. Thank you for showing your interest in this project and me. Thank you for keeping me motivated to perform when I was down and cheering me on until the end.

Bibliography

Baddeley, A. and Gill, R. D. (1997). Kaplan-meier estimators of interpoint distance distributions for spatial point processes.

Baddeley, A., Rubak, E., and Turner, R. (2015). *Spatial Point Patterns: Methodology and Applications with R*. Chapman and Hall/CRC Press, London.

Baddeley, A. and Turner, R. (2005). spatstat: An R package for analyzing spatial point patterns. *Journal of Statistical Software*, 12(6):1–42.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Cai, Q., Fang, Y., and Young, K. H. (2019). Primary central nervous system lymphoma: Molecular pathogenesis and advances in treatment. *Translational Oncology*, 12(3):523 – 538.

Cho, H., Kim, S. H., Kim, S.-J., Chang, J. H., Yang, W. I., Suh, C.-O., Cheong, J.-W., Kim, Y. R., Lee, J. Y., Jang, J. E., Kim, Y., Min, Y. H., and Kim, J. S. (2017). The prognostic role of cd68 and foxp3 expression in patients with primary central nervous system lymphoma. *Annals of Hematology*, 96:1163–1173.

Feng, Z., Bethmann, D., Kappler, M., Ballesteros-Merino, C., Eckert, A., Bell, R., Cheng, A., Bui, T., Leidner, R., Urba, W., Johnson, K., Hoyt, C., Bifulco, C., Bukur, J., Wickenhauser, C., Seliger, B., and Fox, B. (2017). Multiparametric immune profiling in hpv- oral squamous cell cancer. *JCI Insight*.

Four, M., Cacheux, V., Tempier, A., Platero, D., Fabbro, M., Marin, G., Leventoux, N., Rigau, V., Costes-Martineau, V., and Szablewski, V. (2017). Pd1 and pdl1 expression in primary central nervous system diffuse large b-cell lymphoma are frequent and expression of pd1 predicts poor survival. *Hematological Oncology*, 35(4):487–496.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22.

Furuse, M., Kuwabara, H., Ikeda, N., Hattori, Y., Ichikawa, T., Kagawa, N., Kikuta, K., Tamai, S., Nakada, M., Wakabayashi, T., Wanibuchi, M., Kuroiwa, T., Hirose, Y., and Miyatake, S.-I. (2020). Pd-l1 and pd-l2 expression in the tumor microenvironment including peritumoral tissue in primary central nervous system lymphoma. *BMC Cancer*, 20:277.

Gillies, R. J., Kinahan, P. E., and Hricak, H. (2016). Radiomics: Images are more than pictures, they are data. *Radiology*, 278(2):563–577. PMID: 26579733.

Ishwaran, H. and Kogalur, U. (2020). *Fast Unified Random Forests for Survival, Regression, and Classification (RF-SRC)*. R package version 2.9.3.

Ishwaran, H., Kogalur, U., Blackstone, E., and Lauer, M. (2008). Random survival forests. *Ann. Appl. Statist.*, 2(3):841–860.

Johnson, K. S. (2020). *phenoptr: inForm Helper Functions*. R package version 0.2.7.

Kim, S., Nam, S. J., Park, C., Kwon, D., Yim, J., Song, S. G., Ock, C.-Y., Kim, Y. A., Park, S. H., Kim, T. M., and Jeon, Y. K. (2019). High tumoral pd-l1 expression and low pd-1+ or cd8+ tumor-infiltrating lymphocytes are predictive of a poor prognosis in primary diffuse large b-cell lymphoma of the central nervous system. *OncoImmunology*, 8(9):e1626653.

Kingman, J. (2005). *Poisson Processes*. American Cancer Society.

Li, Y.-L., Shi, Z.-H., Wang, X., Gu, K.-S., and Zhai, Z.-M. (2019). Tumor-associated macrophages predict prognosis in diffuse large b-cell lymphoma and correlation with peripheral absolute monocyte count. *BMC Cancer*, 19:1049.

Liaw, A. and Wiener, M. (2002). Classification and regression by randomforest. *R News*, 2(3):18–22.

Marcelis, L., Antoranz, A., Delsupehe, A.-M., Biesemans, P., Ferreiro, J. F., Debackere, K., Vandenberghe, P., Verhoef, G., Gheysens, O., Cattoretti, G., Bosisio, F. M., Sagaert, X., Dierickx, D., and Tousseyn, T. (2020). In-depth characterization of the tumor microenvironment in central nervous system lymphoma reveals implications for immune-checkpoint therapy. *Cancer Immunology, Immunotherapy*, 4(69):1751–1766.

Muller, A. C. and Guido, S. (2016). *Introduction to Machine Learning with Python: a Guide for Data Scientists*. O'Reilly Media, Inc., Sebastopol, CA.

Pya, N. (2020). *scam: Shape Constrained Additive Models*. R package version 1.2-6.

R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Robin, X., Turck, N., Hainard, A., Tiberti, N., Lisacek, F., Sanchez, J.-C., and Müller, M. (2011). proc: an open-source package for r and s+ to analyze and compare roc curves. *BMC Bioinformatics*, 12:77.

Schurch, C. M., Bhate, S. S., Barlow, G. L., Phillips, D. J., Noti, L., Zlobec, I., Chu, P., Black, S., Demeter, J., McIlwain, D. R., Samusik, N., Goltsev, Y., and Nolan, G. P. (2019). Coordinated cellular neighborhoods orchestrate antitumoral immunity at the colorectal cancer invasive front. *bioRxiv*. AACR.

Taylor, E. J. (2000). *Dorland's Illustrated medical dictionary*. Philadelphia: Saunders, 29 edition.

te Beest, D. E. (2020). *CoRF: Co-data guided RandomForest (CoRF)*. R package version 0.1.0.

te Beest, D. E., Mes, S. W., Wilting, S. M., Brakenhoff, R. H., and van de Wiel, M. A. (2017). Improved high-dimensional prediction with random forests by the use of co-data. *BMC Bioinformatics*, 18.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.

van de Wiel, M., Lien, T., Verlaat, W., van Wieringen, W., and Wilting, S. (2016). Better prediction by use of co-data: adaptive group-regularized ridge regression. *Statistics in Medicine*.

van Nee, M. M., Wessels, L. F., and van de Wiel, M. A. (2020a). *ecpc: Flexible Co-data Learning For High-dimensional Prediction*. R package version 2.0.

van Nee, M. M., Wessels, L. F., and van de Wiel, M. A. (2020b). Flexible co-data learning for high-dimensional prediction. *arXiv preprint arXiv:2005.04010*.

Yang, X. and Liu, Y. (2017). Advances in pathobiology of primary central nervous system lymphoma. *Chin Med J (Engl)*, pages 1973–1979.