

Hochschule  
für Technik  
Stuttgart

University of Applied Sciences

Master of Science Programme  
Photogrammetry and Geoinformatics  
Master Thesis  
Winter Term 2024/2025

Habitat selection of the northern  
lapwing (*Vanellus vanellus*) during  
breeding season in Europe

By:

Lady Johanna Esguerra Montaña

Supervisors:

Prof. Dr.-Ing. Angela Blanco-Vogt  
Dr. Elina Takola

# **Habitat selection of the northern lapwing (*Vanellus vanellus*) during breeding season in Europe**

**by**

**Lady Johanna Esguerra Montaña**

**A dissertation presented in partial fulfillment of the requirements for the degree of  
Master of Science in the Department of Geomatics, Computer Science and Mathe-  
matics, Stuttgart University of Applied Sciences**

## **Declaration**

This Master's thesis was written in my own words without any external assistance. All sources of literature used are listed at the end of the thesis. The study presented here was conducted under the supervision of Prof. Dr.-Ing. Angela Blanco-Vogt and Dr. Elina Takola, head of the research group BIOECOS, part of the Helmholtz Centre for Environmental Research – UFZ Leipzig.

I hereby grant to Stuttgart University of Applied Sciences permission to reproduce and to distribute publicly paper and electronic copies of this document in whole and in part.

Stuttgart, 28.02.2025

\_\_\_\_\_  
(Lady Johanna Esguerra Montaña)

Approved by:

\_\_\_\_\_  
(Prof. Dr.-Ing. Angela Blanco-Vogt)

## Acknowledgement

I would like to express my deepest gratitude to the German Academic Exchange Service (DAAD) for giving me the opportunity to study in Germany. Their support was essential in allowing me to pursue this master's degree and successfully complete this journey. I am also grateful to the Stuttgart Technology University of Applied Sciences and its professors for sharing their invaluable knowledge and experience. Each lesson has been a cornerstone in my academic and professional growth.

My heartfelt thanks go to my supervisor, Prof. Dr.-Ing. Angela Blanco-Vogt, for her constant guidance, always offered with kindness and generosity. Her trust in my abilities and words of encouragement were an invaluable source of motivation throughout this process. I am profoundly grateful to my supervisor, Dr. Elina Takola, whose generosity and steadfast support were invaluable throughout this journey. Her willingness to guide me at key moments, her ability to teach me new perspectives, and her patience in sharing her experience greatly enriched both my academic and personal growth.

I sincerely appreciate the Helmholtz Centre for Environmental Research (UFZ) and its team for welcoming me and providing the necessary resources to develop this thesis. In particular, I want to express my sincere gratitude to Dr. Ulrike Schlägel, whose guidance and insightful ideas helped me stay on track and achieve the objectives set for this study.

To my classmates and colleagues, with whom I shared almost two years of learning and unforgettable experiences, thank you for broadening my perspective and deepening my understanding of different cultures. A special thanks to Mónica Acosta, Fatemeh Rafiei, Emerson Martinez, and Arsalan Mukhtar for their unwavering friendship, for supporting me during the most challenging moments, and for becoming an essential part of my support network.

To Ole Tjardes, my deepest appreciation for his steadfast support and for walking this path with me. I could not have completed this journey without his continuous encouragement, insightful ideas, and, above all, his belief in me every step of the way. His presence has been an inspiration, always pushing me to give my best.

Finally, to my family, for being my greatest source of inspiration, for their confidence in me, and for their unconditional support. Their encouragement and belief in my abilities have given me the strength to pursue my dreams with determination. To Carolina Bernal, for being the family we built together over time, always standing by my side despite the distance, and for supporting me in every stage of my life.

Master Course Photogrammetry and Geoinformatics

# Habitat selection of the northern lapwing (*Vanellus vanellus*) during breeding season in Europe

## Abstract

The ongoing population reduction of northern lapwings across Europe has raised serious conservation concerns. One of the primary drivers of this trend is the continuous transformation of their habitats. Land-use changes, particularly the expansion of agricultural fields and intensively managed pastures, have altered the species' breeding environments, significantly affecting reproductive success. Eggs and chicks are especially vulnerable, facing increased risks due to habitat modifications.

To investigate how landscape structure influences habitat selection, this study, conducted in collaboration with Helmholtz Centre for Environmental Research - UFZ, analyzed tracking data from 13 northern lapwings alongside land use and land cover and elevation data. A spatial complexity assessment was performed using landscape metrics, which were then integrated into step-selection function (SSF) models to quantify habitat preferences.

Findings from the SSF models indicate that, under the tested scenarios, habitats characterized by homogeneous resource diversity and spatial configuration increase the relative probability of habitat selection. These results provide key insights into how northern lapwings interact with their environment, offering a foundation for designing more effective conservation strategies tailored to their ecological needs.

**Keywords:** northern lapwing, habitat selection, landscape metrics, GIS, step-selection functions, SSF.

# Contents

<b>Acknowledgement .....</b>	<b>2</b>
<b>Abstract.....</b>	<b>3</b>
<b>Contents .....</b>	<b>4</b>
<b>Index of Figures.....</b>	<b>6</b>
<b>Index of Tables .....</b>	<b>8</b>
<b>Abbreviations .....</b>	<b>9</b>
<b>1 Introduction.....</b>	<b>10</b>
1.1 Theoretical Background.....	11
1.1.1 Habitat characteristics of the northern lapwing .....	12
1.1.2 Modelling animal habitat selection.....	14
1.2 State of the art for the analysis of habitat selection .....	15
1.3 Objectives and Key Questions.....	18
1.3.1 Key questions.....	18
1.3.2 Objectives .....	18
<b>2 Methodology .....</b>	<b>19</b>
2.1 Data.....	19
2.1.1 Northern lapwing tracking data .....	19
2.1.2 Land cover and land use dataset .....	21
2.1.3 Digital Elevation Model (DEM).....	25
2.2 Software and Technology .....	27
2.2.1 R .....	27
2.2.2 Google Earth Engine (GEE) .....	29
2.2.3 Python.....	29
2.2.4 ArcGIS Pro .....	30
2.3 Methods .....	30
2.3.1 Methodological framework.....	30
2.3.2 Defining study area.....	32
2.3.3 Assessing changes in land cover and land use.....	33
2.3.4 Calculating local landscape metrics.....	35
2.3.5 Assessing collinearity in environmental variables.....	41
2.3.6 Implementation step-selection function (SSF) model.....	43
2.3.7 Model performance.....	47
<b>3 Results .....</b>	<b>50</b>

3.1	Landscape complexity assessment.....	50
3.1.1	Global landscape metrics .....	50
3.1.2	Local landscape metrics.....	51
3.1.3	Used steps distribution in landscape metrics .....	57
3.2	Estimation parameters of step-selection functions .....	59
3.2.1	General findings.....	60
3.2.2	The effect of landscape diversity on the habitat use of the northern lapwing .....	60
3.2.3	The effect of the patch size on the habitat use of the northern lapwing .....	62
3.2.4	The effect of the aggregation of patches on the habitat use of the northern lapwing .....	65
3.3	Model performance.....	67
3.3.1	Akaike’s Information Criterion for model selection.....	67
3.3.2	K-Fold cross validation.....	68
<b>4</b>	<b>Discussion .....</b>	<b>70</b>
<b>5</b>	<b>Prospects for future study .....</b>	<b>73</b>
5.1	Movement simulation approach.....	73
5.2	Multitemporal analysis .....	73
<b>6</b>	<b>Conclusions.....</b>	<b>74</b>
<b>7</b>	<b>References.....</b>	<b>76</b>
<b>8</b>	<b>Appendix.....</b>	<b>85</b>
8.1	Appendix. Google Earth Engine code developed in JavaScript, for generating, visualizing, and exporting Dynamic World 2021 data across the interest area. ....	85
8.2	Appendix. Code developed in python to extract tracking data of breeding season. ....	88
8.3	Appendix. JavaScript code for Shannon diversity index calculation in environment GEE. ....	89
8.4	Appendix. Python (Colab) code for Mean Patch Area index calculation. ....	91
8.5	Appendix. Python (Colab) code for Contagion index calculation. ....	94
8.6	Appendix. R code developed for SSF model fitting. ....	97
8.7	Appendix. R script to perform K-Fold cross validation of SSF model fitting, using AUC metric.....	101
8.8	Appendix. R script to perform AIC calculation iteratively in a set of fitted models using the generic function <i>AIC()</i> .....	103
8.1	Appendix. Standard outputs of SSF fitted models.....	104

## Index of Figures

Figure 1. Photo of a northern lapwing: This bird is distinguished for its metallic green, blue, and violet colors, as well as its distinctive crest (Source: Derer, n.d.).	12
Figure 2. Left photo: A northern lapwing nesting on the ground in a grassland area (Source: NABU, n.d.). Right photo: A northern lapwing nest with three eggs in a grassland area (Source: Grundmann, n.d.)	13
Figure 3. Geographic distribution map of the northern lapwing ( <i>Vanellus vanellus</i> ): The legend details presence and conservation status of the species (Sourced: BirdLife International, 2016).	13
Figure 4. Visualization of map tracks of the 13 individuals from the Moveback repository, with each individual's name displayed on the left. The region of the tracked movements covers southern Spain, France, Belgium, the Netherlands and England.	20
Figure 5. Tracking timeline chart showing the recording periods (in months) on the x-axis for each individual's name on the y-axis. The bar labels indicate the number of recorded positions.	21
Figure 6. Dynamic World dataset of the area of interest retrieved from Google Earth Engine Code Catalog.	24
Figure 7. Map of Digital Elevation Model (DEM) dataset of the area of interest retrieved from Google Earth Engine catalog.	26
Figure 8. Workflow diagram showing the structure of the methodology used to conduct the study.	31
Figure 9. Map of the study area for the northern lapwing habitat preference analysis: The polygon outlines the region of positions recorded for the 13 individuals during breeding season.	32
Figure 10. Code snippet developed in Python to filter geospatial data for the breeding seasons (2021-2023), performed in Google Colab.	33
Figure 11. Maps illustrating changes in land cover and land use classes across the study area between 2021, 2022, and 2023: The areas in green represent pixels where the LULC classes remained the same between years and in orange where there were changes.	35
Figure 12. Code snippet developed in JavaScript for computation of class proportions.	37
Figure 13. Code snippet developed in JavaScript for Shannon diversity index computation from class proportions.	37
Figure 14. The SHDI, AREA_MN, and CONTAG landscape metrics are calculated within the extent of the moving window, and the results are assigned to the corresponding output pixel for each metric.	40
Figure 15. R code snippet of the creation of subset of positions per individual using the "id" as criteria to filter the data.	44
Figure 16. R code snippet of the creation of tracks: A differentiated resampling rate and tolerance was configured for each individual.	45
Figure 17. R code snippet for generating 10 random (available) steps from used steps. The input consists of previously created tracks and bursts: The movement characteristics <i>log_sl_</i> (logarithm of the step length) and <i>cos_ta_</i> (cosine of the turning angles) are calculate.	45

Figure 18. Fitting of SSF models using the function in <i>fit_issf</i> in R: For each landscape metric a separate model is fitted.....	46
Figure 19. Code snippet taken from the k-Fold cross validation for SSF models in R: This segment predicts the probability of selection of a step by comparing it with available options.....	48
Figure 20. Shannon diversity index - SHDI (local scale) in the study area generated with the moving window approach (7 × 7 pixel) in Google Earth Engine.....	52
Figure 21. Mean patch area index – AREA_MN (local scale) in the study area generated with the moving window approach (7 × 7 pixel) in Python Colab +. ....	54
Figure 22. Contagion index - CONTAG (local scale) in the study area generated with the moving window approach (7 × 7 pixel) in Python Colab +. ....	56
Figure 23. Distribution of mean Shannon diversity index across used steps per individual. Each individual is identified by the “id” number in the legend on the right. ....	57
Figure 24. Distribution of mean patch area index across used steps per individual. Each individual is identified by the “id” number in the legend on the right. ....	58
Figure 25. Distribution of Contagion index across used steps per individual. Each individual is identified by the “id” number in the legend on the right. ....	59
Figure 26. <b>Estimated parameters in fitted SSF model 1 (landscape diversity) per individual for the years 2021 and 2022.</b> The x-axis displays the ranges of relative selection strength (RSS) values for all variables considered in Model 1. Each color represents an individual, as indicated in the legend on the right. Data from 2021 is represented by circles, while data from 2022 is represented by diamond symbols. ....	61
Figure 27. <b>Estimated parameters in fitted SSF model 2 (size of patches) per individual for the years 2021 and 2022.</b> The x-axis displays the ranges of relative selection strength (RSS) values for all variables considered in Model 2. Each color represents an individual, as indicated in the legend on the right. Data from 2021 is represented by circles, while data from 2022 is represented by diamond symbols. ....	64
Figure 28. <b>Estimated parameters in fitted SSF model 3 (aggregation of patches) per individual for the years 2021 and 2022.</b> The x-axis displays the ranges of relative selection strength (RSS) values for all variables considered in Model 3. Each color represents an individual, as indicated in the legend on the right. Data from 2021 is represented by circles, while data from 2022 is represented by diamond symbols. ....	66

## Index of Tables

Table 1. Table of attributes description of tracking data retrieved in shapefile format from Movebank repository across the 13 individuals. ....	20
Table 2. Dynamic World land use Land cover classification taxonomy. <i>Data sourced from Dynamic World, Near Real-Time Global 10 m Land Use Land Cover Mapping</i> (Brown et al., 2022). ....	22
Table 3. Technical specifications of Dynamic World: Land cover and land use dataset. ....	24
Table 4. Copernicus DEM GLO-30: Global 30m Digital Elevation Model data specification. ....	25
Table 5. Correlation matrix of the predictor variables: Shannon diversity index (SHDI), mean patch area index (AREA_MN), digital elevation model (DEM) and contagion index (CONTAG). ....	42
Table 6. Variance inflation factors (VIF) for predictor variables. The SHDI was treated as the dependent variable for the calculation. ....	43
Table 7. Global landscape metrics and mean patch area index for each LULC in the study area for the year 2021 (spatial resolution: 10 meters). ....	50
Table 8. Distribution table of Shannon diversity index (SHDI) ranges: Absolute frequency (pixel count) and relative frequency (percentage). ....	53
Table 9. Distribution table of mean patch index (AREA_MN) ranges: Absolute frequency (pixel count) and relative frequency (percentage). ....	55
Table 10. Distribution table of contagion index (CONTAG) ranges: Absolute frequency (pixel count) and relative frequency (percentage). ....	57
Table 11. Compiled AIC scores for models by individual for the years 2021–2022. Model 1 included the Shannon diversity index, Model 2 incorporated the mean patch area index, and Model 3 used the contagion index. ....	67
Table 12. Results of the 10-Fold Cross-Validation using the AUC metric to evaluate model performance by individuals for the years 2021–2022. Model 1 included the Shannon diversity index, Model 2 incorporated the mean patch area index, and Model 3 used the contagion index. ....	68

## Abbreviations

AES	Agri-Enviromental Schemes
AIC	Akaike's Information Criterion
AUC	Area under the Curve
DEM	Digital Elevation Model
GEE	Google Earth Engine
GIS	Geographic Information System
GPS	Global Positioning System
LULC	Land Use and Land Cover
RSF	Resource Selection Function
RS	Remote Sensing
RSS	Relative Selection Strength
SFF	Step Selection Function
VIF	Variance Inflation Factor

# 1 Introduction

Landscape transformation, driven by the expansion of agricultural and urban areas, leads to habitat loss, reduced species diversity, and increased population isolation (Morera et al., 2008). Among the species undergoing a continuous population decline is the northern lapwing (*Vanellus vanellus*) which in 2021 was classified as a Vulnerable species on the *IUCN Red List of Threatened Species* (BirdLife International, 2021). Given the ongoing modification of suitable habitats, it becomes crucial to understand how species like the northern lapwing respond to environmental changes. One way to approach this is by examining habitat selection, a process that shapes species' behavior, ecology, and survival.

Habitat selection plays a fundamental role in species preservation. It influences key factors such as food availability, predator avoidance, and competition, ultimately affecting population dynamics and ecosystem stability. In ecological research, resource selection frameworks are widely used to explore how individuals interact with their environment, shedding light on habitat preferences and species adaptations to changing landscapes. This is typically assessed by comparing where the animals are found (i.e., habitat use) with the availability of resources in the surrounding area (Florko et al., 2024).

For species experiencing population declines, studying habitat selection provides key insights into how they interact with their environment and respond to habitat changes. The northern lapwing, a medium-sized bird distributed across Europe, has experienced a steady decline, raising conservation concerns. Despite adaptability of the species to diverse habitats, its numbers have been decreasing over time. This population trend has drawn the interest of academics, researchers, and conservation organizations, prompting a deeper investigation into the factors driving this decline.

Research in the field of ecology has identified consistent patterns affecting northern lapwing populations. One of the most pressing concerns is low reproductive success, which has been linked to agricultural intensification and land-use changes, particularly in breeding habitats such as grasslands and wetlands (Schekkerman et al., 2009). Additionally, predation has been identified as a significant threat, as eggs and chicks are especially vulnerable during the breeding season (Jackson, 2001), further impacting population numbers.

While previous researches have provided valuable insights into the factors contributing to northern lapwing decline, there is still a need for a more detailed understanding of how landscape complexity influences habitat selection. This thesis addresses that gap by adopting a fine-scale analytical approach to examine habitat selection of the northern lapwing. In collaboration with

the **BIOECOS** research group, which focuses on biodiversity and ecosystem services and is part of the **Helmholtz-Centre for Environmental Research -UFZ Leipzig**, this study explores how the landscape complexity influences the habitat selection process of this species during the breeding season. The following sections will provide an overview of the species' ecological context and the current state of the art, establishing the foundation for the key questions and objectives of this study.

## 1.1 Theoretical Background

The study followed two main lines of analysis: landscape complexity and integration of local landscape metrics with real movement data. The first line of analysis, landscape complexity was quantified using land use and land cover data. To capture habitat diversity and spatial configuration, the study incorporates three key landscape metrics: Shannon diversity index, mean patch area index, and contagion index, based on the 10-meter resolution land use and land cover dataset named *Dynamic World*. These metrics provide a detailed assessment of landscape complexity, which will be further explained in the section 2.3.4 *Calculating local landscape metrics*. The analysis examines habitat selection at a scale relevant to the species' immediate surroundings using a moving window approach. This method ensured that each output pixel reflects landscape metrics within a defined local extent, providing a detailed spatial representation of habitat structure.

The second stage of the analysis integrates local landscape metrics with real movement data. The study uses tracking data from 13 northern lapwing individuals, sourced from the *Movebank* platform, which leverages GPS devices to ensure data accuracy and continuous monitoring. To achieve this integration, the analysis applies statistical models commonly used in resource selection studies. In addition to landscape metrics, the methodology incorporates elevation and land use and land cover information as potential factors influencing habitat selection in northern lapwing.

These two analytical approaches aim to generate insights into how landscape structure influences habitat selection during the breeding season, within the spatial and temporal scales defined by the available data. Understanding this relationship is essential for assessing the ecological factors shaping northern lapwing habitat use. The following subsections will further explore key concepts relevant to this study, along with related research that provides context for the findings.

### 1.1.1 Habitat characteristics of the northern lapwing

The northern lapwing is a semi-colonial species (see reference photo of a lapwing in Figure 1), with group sizes ranging from 1 to 8 nests (Watson et al., 2006). Northern lapwing are ground-nesting birds (see reference photos in Figure 2), and the median clutch size is four (Sheldon et al., 2013). Although the young are precocial (Parish & Coulson, 1998), they follow their parents to feeding grounds (Korner et al., 2024), before eventually becoming independent. In Europe, northern lapwing migrate during winter from north and northeast to the south and southwest, but they return to their natal grounds in order to breed (Lislevand et al., 2009). (See distribution map of the species in Figure 3).



Figure 1. Photo of a northern lapwing: This bird is distinguished for its metallic green, blue, and violet colors, as well as its distinctive crest (Source: Derer, n.d.).



Figure 2. Left photo: A northern lapwing nesting on the ground in a grassland area (Source: NABU, n.d.). Right photo: A northern lapwing nest with three eggs in a grassland area (Source: Grundmann, n.d.)



Figure 3. Geographic distribution map of the northern lapwing (*Vanellus vanellus*): The legend details presence and conservation status of the species (Sourced: BirdLife International, 2016).

As described by Eglinton et al. (2010) and the Royal Society for the Protection of Birds (RSPB) (2017), the northern lapwing is primarily distributed on wetlands and grasslands in both lowland and upland areas, with a smaller presence in arable farmland and intensively managed pastures. These habitats provide key food resources, including earthworms, crane fly larvae, surface-dwelling insects, and other invertebrates. Nesting typically occurs in open fields with short vegetation, which allows for early predator detection. Ideal nesting sites include sparsely vegetated grasslands and areas with open water, offering both safety and proximity to food sources. This species ex-

hibits strong site fidelity, returning to the same fields year after year. However, widespread drainage of wetlands and grasslands, the reseeded of faster-growing grass species, increased fertilizer use, intensive grazing, and mechanized farming have altered these landscapes. These changes in environmental conditions reduce both the availability and the quality of food resources essential for the species.

Although management strategies have been implemented to slow the decline of the northern lapwing population, such as agri-environment schemes (AES) in the Netherlands, where delayed mowing was introduced to protect nests and chicks, the results have not met expectations (Barba-Escoto et al., 2024). This contrasts with similar studies in the UK, which reported benefits for some species, but the results were influenced by biases in data interpretation (Breeuwer et al., 2009) (Breeuwer et al., 2009). To develop more effective conservation strategies, it is crucial to understand the complex relationship between the northern lapwing and the landscape, particularly agricultural environments, and to address the factors contributing to its population decline.

### **1.1.2 Modelling animal habitat selection**

A key aspect of understanding species decline is analyzing how individuals interact with their environment. Animal movement models address this by using stochastic processes to describe movement trajectories, represented as sequences of steps or transitions between consecutive positions. These movements are influenced by factors such as speed, direction, resource availability, and other environmental variables (Florko et al., 2024). This approach helps researchers study the determinants of animal selection for specific environmental characteristics, such as proximity to food sources or shelter (Thurfjell et al., 2014).

A movement-based approach to resource selection studies often rely on step-selection functions (SSF), which are an advanced extension of resource selection functions (RSF). While RSF are based on comparisons between specific locations used and available (Boyce et al., 2002), the SSF defines the selection of locations by considering the individual's location at a given time and its potential movement between observations (Buderman et al., 2023). This approach is particularly useful when there is serial dependence in an individual's locations and when the time between observations is short relative to the size of its home range (Fieberg et al., 2021).

In SSF implementation, steps are defined as the linear displacement between two consecutive locations of the animal. For each observed step, random steps are generated that represent alternatives that the animal could have taken, and the environmental attributes of the used steps are compared to the random steps. The SSF are particularly useful for studying how animals move

and interact across landscapes, identifying movement routes and migratory corridors that may be essential for conservation (Thurfjell et al., 2014).

The SSF models implemented contribute to predicting habitat selection; however, deriving clear insights from their results can present challenges. This is where the Relative Selection Strength (RSS) becomes relevant. RSS is a metric that quantifies the strength with which an animal selects one habitat over another (Fieberg et al., 2021). Essentially, it indicates how preferable a given location is compared to another while keeping all other conditions constant. Using RSS is crucial because it provides a practical interpretation of the estimated coefficients in SSF models, allowing for comparisons of the significance of different factors influencing habitat selection both within and across studies (Avgar et al., 2017). RSS enhances the clarity of results and facilitates their application in wildlife management and conservation.

## 1.2 State of the art for the analysis of habitat selection

Researchers have sought to understand the factors driving the population decline of the northern lapwing. Recent studies have focused on key ecological and reproductive challenges affecting northern lapwing populations. For instance, van den Berg (2024), in the Ph.D. thesis "*The breeding ecology of the northern lapwing (Vanellus vanellus) in France: investigating the decline of a widely-distributed wader*" examined hatching success rates in France and the environmental factors influencing reproductive success.

The research by van den Berg (2024) also found that agricultural activities such as land preparation, seeding and mechanical weeding are often carried out during the birds' breeding season, which can result in the direct destruction of nests. Low wind speeds were also found to increase thermoregulation costs for the birds and higher exposure to predators by making it difficult to dissipate odors generated around the nest.

The study of animal-habitat interactions has advanced significantly in recent years, driven by improvements in GPS telemetry and the availability of high-resolution movement data. These advancements have helped identify critical movement corridors and key conservation areas by shedding light on how animals respond to habitat fragmentation (Thurfjell et al., 2014). Additionally, the integration of geographic information systems (GIS) and remote sensing (RS) technologies with animal movement data has further enhanced the understanding of their movement patterns (Craft et al., 2025). Movebank stands as one of the most successful integrations of these technologies for ecological research, serving as a platform for sharing and managing movement

data collected from worldwide multi-sensor datasets. Together, these technological advancements provide invaluable tools for understanding species movement.

Fieberg et al. (2021) and Thurfjell et al. (2014) provide a detailed framework in their papers “A ‘How to’ guide for interpreting parameters in habitat-selection analyses” and “Applications of step-selection functions in ecology and conservation” for integrating GIS and RS with animal tracking data through the implementation of SSF as statistical models. Their approach compares the environmental attributes of the “used steps” taken by the animal with alternative movement options, making it applicable to a wide range of species, including studies on northern lapwing-environment interactions. Together, these papers serve as a step-by-step guide for data preparation, outlining how to process environmental variables for analysis, generate available locations, and fit SSF models in R. Additionally, they provide insights into the tools and resources available to implement these models effectively

The paper “Using step and path selection functions for estimating resistance to movement: pumas as a case study” (Zeller et al., 2016) presents a case study that effectively applies SSF to model animal movement and assess landscape resistance in a population of pumas (*Puma concolor*) in southern California. The study incorporated movement steps obtained from GPS-collared individuals as input data along with a 30-meter resolution land cover raster. The results revealed that pumas exhibit habitat preferences for grasslands, barren areas, chaparral, coastal scrub, and perennial grasslands. Additionally, the study underscored the influence of spatial scale on the outcomes, highlighting the importance of considering scale effects when analyzing movement patterns.

Other studies have explored northern lapwing habitat preferences and the role of landscape features in nesting success. The study “Habitat Selection and Diet of Lapwing *Vanellus vanellus* Chicks on Coastal Farmland in S.W. Sweden” by (Johansson & Blomqvist, 1996) examined how lapwing chicks use coastal habitats by comparing the composition of their breeding areas with the availability of surrounding resources. For this, there was a monitoring of their movement, diet, and food availability across different grassland types. The study determined that chicks preferred coastal grasslands, linked to the fact that agricultural lands do not provide adequate foraging resources for lapwing chicks, highlighting potential habitat limitations.

The study “Landscape effects on nest site selection and nest success of northern Lapwing *Vanellus vanellus* in lowland wet grasslands” by (Bertholdt et al., 2017) investigated whether forest patches in wet grasslands influence the nesting distribution and predation rates of the northern

lapwing. Researchers mapped nest locations, monitored nesting success using temperature loggers, and analyzed the distance between nests and forest patches. This study revealed that the northern lapwing tends to avoid nesting near forests, and that predation rates did not vary significantly when the distance to forest patches changed. This suggests that their natural predators are not concentrated in near forest edges.

The decline of northern lapwing populations is partially explained by the influence of environmental changes, agricultural expansion, and habitat dynamics. The state of the art discussed provides in this section address a strong foundation for understanding how habitat characteristics shape site selection and, ultimately, species survival. Advances in tracking technologies, GIS, and statistical models have made it possible to analyze these relationships in greater detail, offering insights about habitat use. Based on this framework, it is possible carry more detailed studies of the species' habitat interactions and contribute inputs to designing effective conservation strategies.

## 1.3 Objectives and Key Questions

### 1.3.1 Key questions

This thesis explores the following research questions, each designed to provide deeper insight into the problem at hand:

- I. How is the landscape complexity of the interest area at a local context of the northern lapwing by considering its diversity and spatial structure?
- II. Do northern lapwings show a preference of habitat based on the complexity of the landscape?

### 1.3.2 Objectives

Based on the literature review, two main methods are used to address the research questions. The methods are computing local landscape metrics and implementation of step-selection function models. The main goal of this thesis is to evaluate how landscape complexity influences the habitat selection of the northern lapwing during breeding season by examining whether the species exhibits a preference for a specific spatial configuration and composition of the landscape. To achieve this goal, the study is addressed to the specific objectives presented below:

- I. To evaluate spatial structure and resources diversity of landscape in the study area considering the step length of the northern lapwing as a local scale reference.
- II. To identify the habitat preferences of northern lapwings by modeling step selection functions, integrating animal movement data with environmental variables.

## 2 Methodology

This chapter presents the data, tools and methods used throughout the study. A description of the datasets is given, detailing their main characteristics and relevance to addressing the research questions. Next, the software tools used are presented, explaining their specific role in analyzing and processing the information. Finally, the methodological procedures adopted are described, ensuring that the analysis is in line with the objectives set out in Chapter 1.3.

### 2.1 Data

For this project, two main types of data were needed to meet the objectives. First, a land cover and land use dataset, together with an elevation dataset, provided the inputs to assess landscape complexity and serve as key environmental variables. Second, GPS tracking data from a group of northern lapwings were integrated to model their habitat selection preferences.

#### 2.1.1 Northern lapwing tracking data

The movement dataset for the northern lapwing was obtained from the Movebank repository, an open-access online database for animal tracking data hosted by the Max Planck Institute of Animal Behavior. This dataset comprises information on the movements of individual animals, gathered through GPS tags that accurately record their locations at specific timestamp based on satellite positioning. These locations were transmitted to the user via the GSM network, offering high-resolution and precise tracking of animal movements.

The dataset used in this study, titled “*Lapwing NFW Vanellus vanellus*”, was conducted under the guidance of principal investigator Jelle Loonstra and tracks the movements of the species *Vanellus vanellus*. This dataset, hosted on Movebank (Study ID: 1448409403), includes location data for 13 individual animals, with a total of 362,603 location records (See Figure 4). Data were accessed on October 07 of 2024, via the Movebank repository, under a Creative Commons Attribution-NonCommercial (CC BY-NC) license that allow public access.

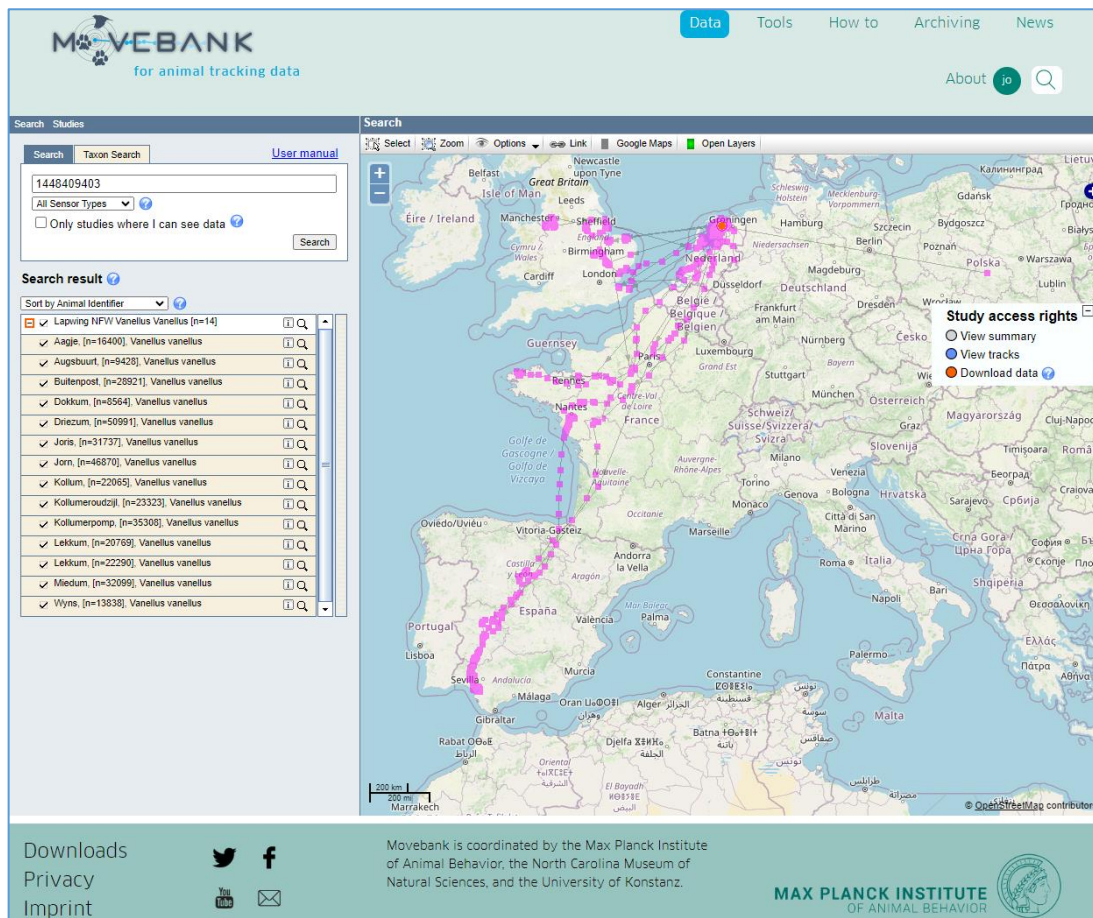


Figure 4. Visualization of map tracks of the 13 individuals from the Moveback repository, with each individual's name displayed on the left. The region of the tracked movements covers southern Spain, France, Belgium, the Netherlands and England.

The data were retrieved in shapefile format, including two files: a point-geometry shapefile representing the positions of the 13 individuals, and a second polyline-geometry shapefile representing the movement steps between positions. Relevant attributes in these datasets for the study summarize in Table 1.

Table 1. Table of attributes description of tracking data retrieved in shapefile format from Movebank repository across the 13 individuals.

Field name	Description	Data type
<b>FID</b>	It is a unique identifier for each feature in the shapefile.	Numeric
<b>tag_idnt</b>	Identifier of the device attached to the animal to collect and transmit data (numeric).	Text

<b>ind_ident</b>	Identifier of the animal by a given name.	Text
<b>individual</b>	Species observed, in this study all of the individuals are <i>Vanellus vanellus</i> .	Text
<b>sensor_typ</b>	Type of sensor used to collect positioning data, in this study GPS was the sensor used with all individuals.	Text
<b>timestamp</b>	Date and time at which the individual's position was recorded.	Date
<b>Long</b>	Longitude of the position recorded.	Double
<b>lat</b>	Latitude of the position recorded.	Double

The tracking period spans from March 20, 2021, to March 15, 2023, where the timestamp differs across individuals and thus the number of positions recorded as shown in Figure 5. The individual “Lekkum” had two different deployments (d1 and d2) with partially overlapping timestamps.

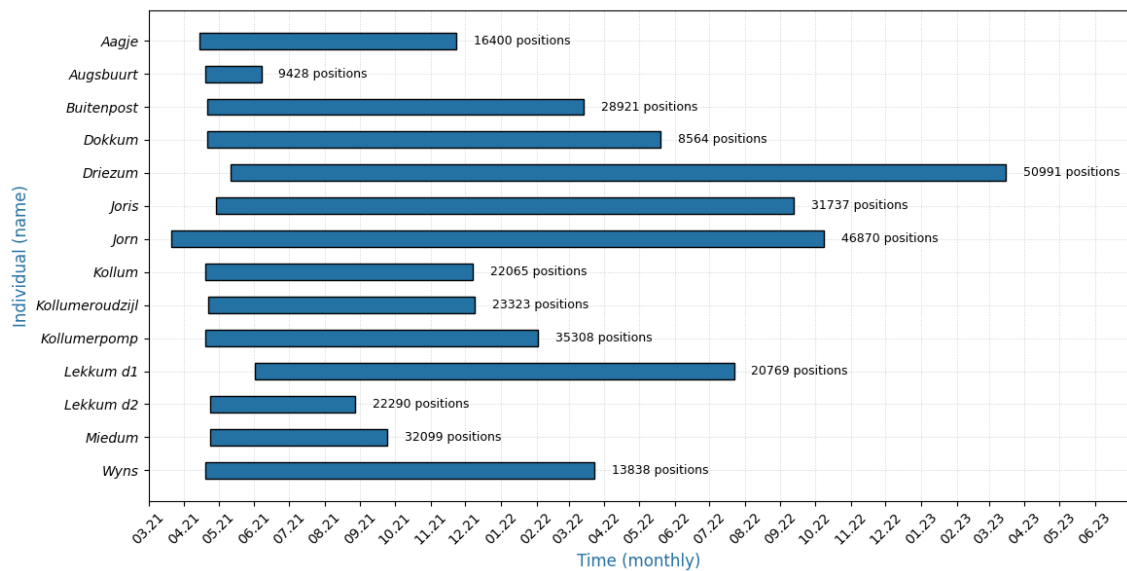


Figure 5. Tracking timeline chart showing the recording periods (in months) on the x-axis for each individual's name on the y-axis. The bar labels indicate the number of recorded positions.

### 2.1.2 Land cover and land use dataset

To accurately analyze land cover and landscape complexity in this study, the dataset Dynamic World was used. Dynamic World is a near real-time 10m resolution global land use land cover dataset, produced using deep learning, freely available and openly licensed. resulting of a partnership between Google and the World Resources Institute (Google & the World Resources Institute, n.d.).

Dynamic World predictions are produced for Sentinel-2 L1C images where the percentage of cloud-covered pixels is 35% or lower. To enhance accuracy, clouds and their shadows are filtered out using a combination of S2 Cloud Probability, the Cloud Displacement Index, and the Directional Distance Transform (Google Developers, n.d.)

The model training for Dynamic World employs a Fully Convolutional Neural Network (FCNN), specifically designed for semantic segmentation, to analyze spatial context in Sentinel-2 imagery and interpret global land cover types. The FCNN predicts 9-class probabilities for each pixel by combining spatial and spectral data, following the taxonomy outlined in Table 2. The model delivers high accuracy worldwide, aligning closely with expert annotations (Brown et al., 2022)

Table 2. Dynamic World land use Land cover classification taxonomy. *Data sourced from Dynamic World, Near Real-Time Global 10 m Land Use Land Cover Mapping* (Brown et al., 2022).

Class ID	Land use/land cover (LULC)	Description	Examples
0	Water	<ul style="list-style-type: none"> <li>Water is present in the image.</li> <li>Contains little-to-no sparse vegetation, no rock outcrop, and no built-up features like docks.</li> <li>Does not include land that can or has previously been covered by water.</li> </ul>	<ul style="list-style-type: none"> <li>Rivers</li> <li>Ponds &amp; Lakes</li> <li>Ocean</li> <li>Flooded Salt Pans</li> </ul>
1	Trees	<ul style="list-style-type: none"> <li>Any significant clustering of dense vegetation, typically with a closed or dense canopy.</li> <li>Taller and darker than surrounding vegetation (if surrounded by other vegetation).</li> </ul>	<ul style="list-style-type: none"> <li>Wooded vegetation</li> <li>Dense green shrubs</li> <li>Cluster of dense, tall vegetation within savannas</li> <li>Plantations such as apples, bananas, citrus, and rubber</li> <li>Swamp (dense/tall vegetation with no obvious water)</li> <li>Any mix of the above</li> <li>Any burned areas of the above</li> </ul>
2	Grass	<ul style="list-style-type: none"> <li>Open areas covered in homogenous grasses with little to no taller vegetation.</li> <li>Other homogenous areas of grass-like vegetation (blade-type leaves) that appear different from trees and shrubland.</li> <li>Wild cereals and grasses with no obvious human plotting (i.e. not a structured field).</li> </ul>	<ul style="list-style-type: none"> <li>Natural meadows and fields with sparse or no tree cover</li> <li>Open savanna with little to no tree cover</li> <li>Parks, golf courses, human manicured lawns, including large fields in urban settings like soccer and baseball.</li> <li>Tree cut-throughs for power lines, gas etc.</li> <li>Pastures</li> <li>Reeds and marshes with no obvious flooding</li> </ul>
3	Flooded vegetation	<ul style="list-style-type: none"> <li>Areas of any type of vegetation with obvious</li> <li>intermixing of water.</li> </ul>	<ul style="list-style-type: none"> <li>Flooded mangroves</li> <li>Emergent vegetation</li> </ul>

		<ul style="list-style-type: none"> <li>Do not assume an area is flooded if flooding is observed in another image.</li> <li>Seasonally flooded areas that are a mix of grass/shrub/trees/bare ground.</li> </ul>	
4	Crops	<ul style="list-style-type: none"> <li>Human planted/plotted cereals, grasses, and crops.</li> </ul>	<ul style="list-style-type: none"> <li>Corn, wheat, soy, etc.</li> <li>Hay and fallow plots of structured land</li> </ul>
5	Shrub & Scrub	<ul style="list-style-type: none"> <li>Mix of small clusters of plants or individual plants dispersed on a landscape that shows exposed soil and rock.</li> <li>Scrub-filled clearings within dense forests that are clearly not taller than trees. Appear grayer/browner due to less dense leaf cover.</li> </ul>	<ul style="list-style-type: none"> <li>Moderate to sparse cover of bushes, shrubs, and tufts of grass</li> <li>Savannas with very sparse grasses, trees, or other plants</li> </ul>
6	Built area	<ul style="list-style-type: none"> <li>Clusters of human-made structures or individual very large human-made structures.</li> <li>Contained industrial, commercial, and private building, and the associated parking lots.</li> <li>A mixture of residential buildings, streets, lawns, trees, isolated residential structures or buildings surrounded by vegetative land covers.</li> <li>Major road and rail networks outside of the</li> <li>predominant residential areas.</li> <li>Large homogeneous impervious surfaces, including parking structures, large office buildings, and residential housing developments containing clusters of cul-de-sacs.</li> </ul>	<ul style="list-style-type: none"> <li>Cluster of houses, can include small lawns or small patches of trees can be included</li> <li>Dense villages, town, and cityscape (buildings and roads together)</li> <li>Clusters of paved roads and large highways</li> <li>Asphalt and other human-made surfaces</li> </ul>
7	Bare ground	<ul style="list-style-type: none"> <li>Areas of rock or soil containing very sparse to no vegetation.</li> <li>Large areas of sand and deserts with no to little vegetation.</li> <li>Large individual or dense networks of dirt roads.</li> </ul>	<ul style="list-style-type: none"> <li>Exposed rock</li> <li>Exposed soil</li> <li>Desert and sand dunes</li> <li>Dry salt flats and salt pans</li> <li>Dried lake bottoms</li> <li>Mines</li> <li>Large empty lots in urban areas</li> </ul>
8	Snow & Ice	<ul style="list-style-type: none"> <li>Large homogenous areas of thick snow or ice, typically only in mountain areas or highest latitudes.</li> <li>Large homogenous areas of snowfall.</li> </ul>	<ul style="list-style-type: none"> <li>Glaciers</li> <li>Permanent snowpack</li> <li>Snowfall</li> </ul>

The dataset was accessed and downloaded using the Google Earth Engine Code Editor. A script was written (see Appendix 8.1) in JavaScript syntax, specifying parameters such as the dataset name, the date range for the images within the collection to be included in the output, and the area of interest for data extraction (see Figure 6). The dataset specifications are described in Table 3. The download covering the area of interest contains the tiles for each year considered.

Table 3. Technical specifications of Dynamic World: Land cover and land use dataset.

<b>ID Google Earth Engine</b>	GOOGLE/DYNAMICWORLD/V1
<b>Theme:</b>	Land cover and land use
<b>Classes:</b>	9
<b>Spatial resolution:</b>	10 m
<b>Format:</b>	GeoTIFF
<b>Coverage:</b>	Global
<b>Dataset Provider:</b>	World Resources Institute Google

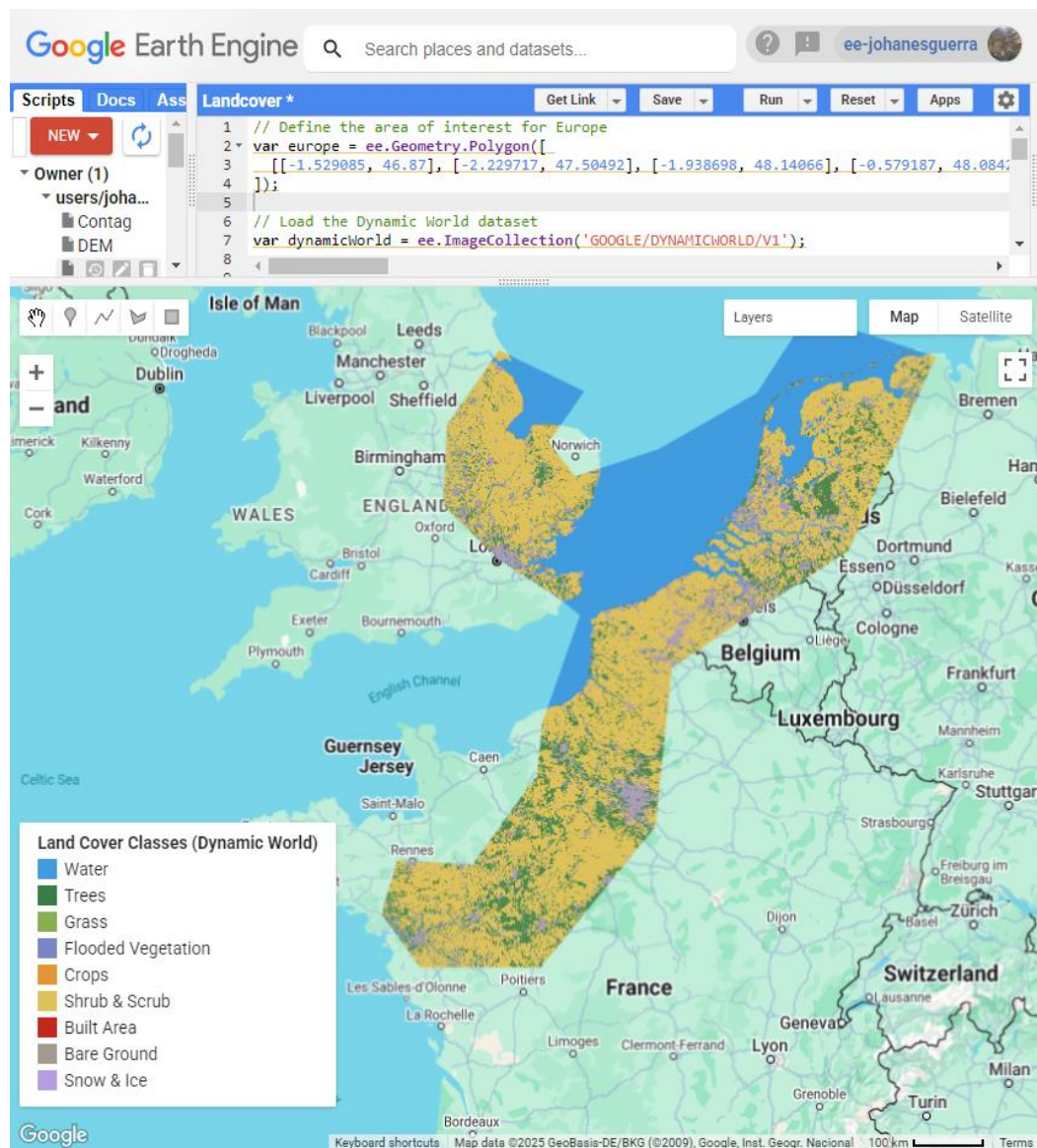


Figure 6. Dynamic World dataset of the area of interest retrieved from Google Earth Engine Code Catalog.

### 2.1.3 Digital Elevation Model (DEM)

Elevation is a key variable in animal movement studies as it helps assess how topography influences an individual's selection of resources and locations. Factors such as slope and elevation can impact the energetic cost of movement as well as the accessibility and preference for different habitats. The Copernicus DEM GLO-30 was integrated in the project, obtained from the Google Earth Engine catalog. This dataset, freely available worldwide under a Copernicus open license (European Space Agency & Airbus, 2022), provides elevation data for the study area. More details about the dataset are presented in Table 4. The DEM contains elevation values ranging from -137.97 to 493.58 meters, with negative values likely corresponding to natural depressions. A visual representation of the dataset is shown in Figure 7.

Table 4. Copernicus DEM GLO-30: Global 30m Digital Elevation Model data specification.

<b>ID Google Earth Engine</b>	COPERNICUS/DEM/GLO30
<b>Theme:</b>	Elevation
<b>Spatial resolution:</b>	30 m
<b>Format:</b>	GeoTIFF
<b>Coverage:</b>	Global
<b>Dataset Provider:</b>	Copernicus

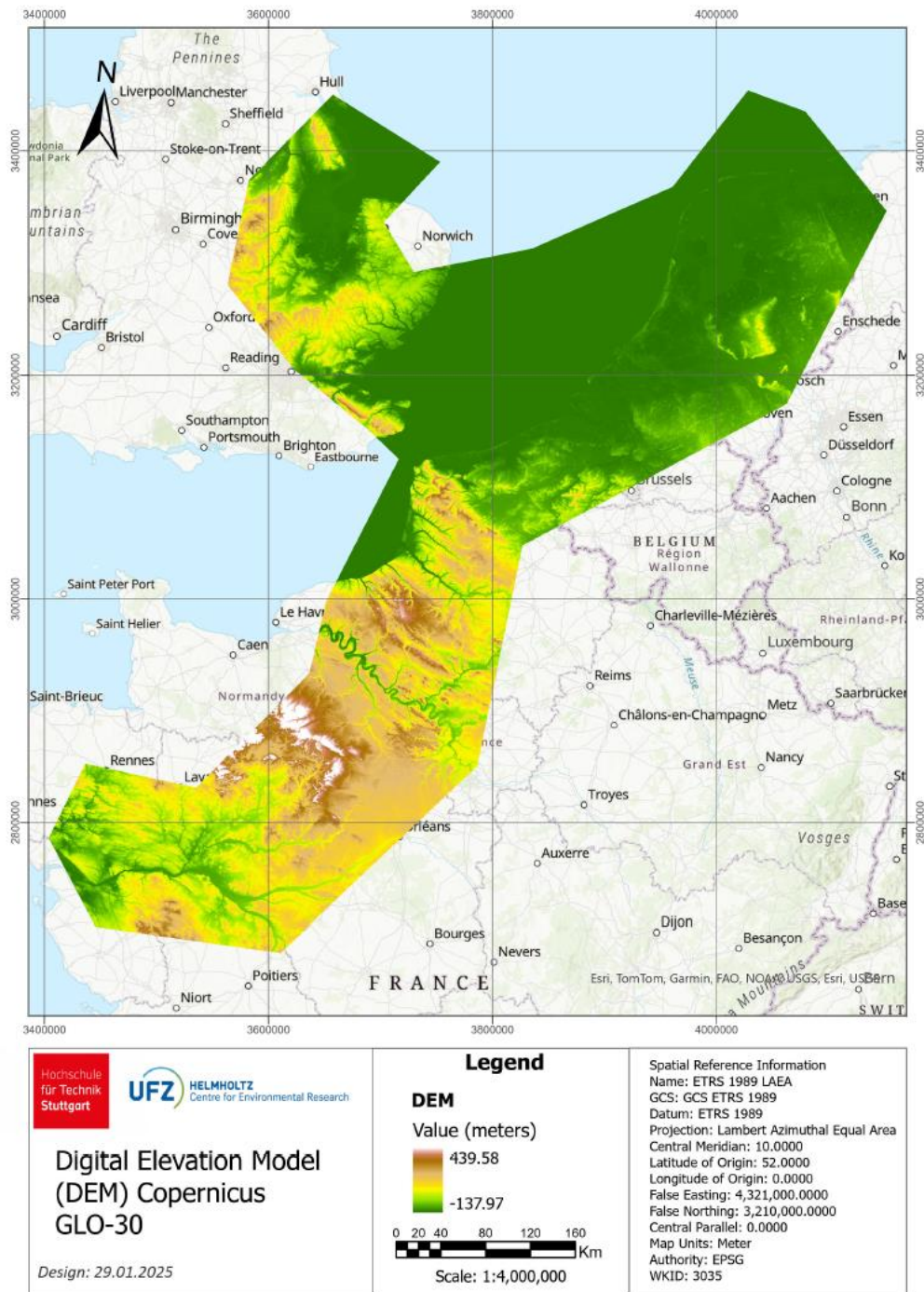


Figure 7. Map of Digital Elevation Model (DEM) dataset of the area of interest retrieved from Google Earth Engine catalog.

## 2.2 Software and Technology

This section dives into the software, programming languages and technologies needed to visualize and process the source data. The R programming language was used to model habitat selection from tracking data and evaluate the performance of the models. In addition, Python and the Google Earth Engine (GEE) platform allow the calculation of metrics that assess landscape complexity from land cover and land use dataset. To complement these analyses, ArcGIS Pro was used to visualize data, generate maps and perform spatial operations. The following sections detail each of these tools.

### 2.2.1 R

R is an open-source programming language for statistical computing and data visualization. R allows to perform advanced statistical analyses on processed data, including the modeling of selection functions, critical to understanding ecological dynamics, leveraging creation of detailed graphs and maps that enhance and contextualize the geospatial analysis. It is widely used in animal movement analysis due to its comprehensive set of packages specifically designed for handling, analyzing, and visualizing movement data.

R Packages like amt (animal movement tools) allows users to preprocess tracking data, calculate movement metrics (such steps length and turning angles), and fit statistical models to understand movement patterns. Additionally, spatial analysis packages, like sp, sf, terra, and raster, facilitate the integration of movement data with environmental and geographic layers. Visualization packages such as ggplot2 provides dynamic, interactive mapping options that help illustrate movement trajectories and habitat interactions over time.

More information about the main R packages employed in this thesis are described below:

Package name	Description
<b>amt</b>	This package offers functionalities to explore and analyze animal movement, including telemetry, and provides tools for calculating home ranges, tracking movement statistics (such as step length, speed, and turning angles), preparing data for habitat selection modeling, and simulating space use based on fitted step-selection functions (Signer, 2024).
<b>landscape-metrics</b>	It provides functions to calculate landscape metrics for categorical landscape patterns in a tidy workflow. The landscapemetrics package reimplements key metrics from the program FRAGSTATS and introduces additional metrics based on recent research in landscape analysis. It supports terra raster objects

	as input and also offers tools for visualizing landscape patches, selecting relevant metrics, and creating custom metrics by combining foundational elements (Hesselbarth et al., 2024)
<b>terra</b>	This package includes functions for creating, reading, modifying, and exporting vector and raster data. Among its capabilities are versatile tools for general raster data manipulation, which can also be adapted to develop custom functions. Also, it provides methods that allow users perform geometric operations in vector data such as points, lines and polygons, and their attributes (Spatial Data Science Web Site, n.d.).
<b>raster</b>	The raster package provides tools and classes for working with geographic (spatial) data in raster format, it is particularly useful for creating, analyzing, and managing raster data. It supports large datasets, spatial operations like distance and neighborhood analysis, data conversion, model predictions, value extraction, visualization, and efficient handling of multiple raster file formats (Hijmans, 2024)
<b>sp</b>	This package provides classes and methods for working with spatial data, designed to manage spatial location information for both 2D and 3D datasets. The package includes utility functions for tasks such as plotting data as maps, performing spatial selection, retrieving coordinates, and operations like subsetting, printing, and generating summaries (Pebesma & Bivand, 2005)
<b>sf</b>	It provides support for simple features, a standardized format for encoding spatial vector data. It integrates with 'GDAL' for reading and writing spatial files, 'GEOS' for performing geometric operations, and 'PROJ' for handling projection conversions and datum transformations (Pebesma, 2018)
<b>ggplot2</b>	ggplot2 offers tools for the creation of visualizations by combining modular components rather than relying on predefined chart types. With defaults for scales, facets, and themes, ggplot2 simplifies the creation of flexible and precise visualizations. In this project, it is particularly useful to examine the input data and visualize outputs generated by regression models, simplifying result interpretation (Wickham et al., n.d.).
<b>survival</b>	This package provides functions for conducting survival analysis, which is used to model the time until an event occurs. It includes tools for fitting Cox models, parametric survival models, and performing survival curve comparison tests. The package also offers diagnostic tools, prediction functions, and visualization capabilities for survival models (Therneau, 2020).
<b>lubridate</b>	This package simplifies working with dates and times in R. It provides functions to parse, manipulate, and perform calculations with dates and times in various formats. lubridate also facilitates the conversion between different date-time formats (Grolemund & Wickham, 2011).
<b>tibble</b>	It introduces an enhanced version of the data frame, known as 'tbl_df' or tibble. Tibbles enforce stricter data validation and offer a cleaner, more readable printing format compared to traditional data frames. As the core data structure of the tidyverse, a collection of packages designed for data science, tibbles also display column types, helping users distinguish between character and factor variables (Müller & Wickham, 2024).
<b>dplyr</b>	This package provides a set of functions for manipulating tabular data. It includes tools for filtering, selecting, arranging, modifying, and summarizing data. Designed to be intuitive and user-friendly, dplyr simplifies data transformation for analysis (Wickham et al., 2022).

<b>spatstat</b>	This is a collection of packages designed for the statistical analysis of spatial data, with a primary focus on two-dimensional spatial point patterns. It offers a wide range of tools for exploratory data analysis, model fitting (Cox, Poisson, Gibbs, etc.), simulation, model diagnostics, and inference. spatstat includes functions for creating and manipulating spatial data, visualization, smoothing, and relative risk estimation. It is particularly useful for modeling spatial intensity and interactions between points (Baddeley et al., 2015).
-----------------	---

### 2.2.2 Google Earth Engine (GEE)

GEE is a cloud-based platform designed for large-scale environmental data analysis, providing access to a wide collection of satellite imagery and thematic datasets (Amani et al., 2020). It offers an explorer composed of an integrated data catalog and workspace and leverages Google's high-speed, parallel processing capabilities and machine learning tools, allowing for efficient analysis on a large scale (Tamiminia et al., 2020).

The code editor is a key component of this study because it allows access to the Dynamic World and DEM datasets using the computational resources available in GEE. It also provides features for developing complex geospatial (United States Department of Agriculture, 2019), which is useful for calculating landscape metrics that require operations such as counting pixels per class in specific extents, applying mathematical calculations based on pixel values, and exporting the results in a raster format.

### 2.2.3 Python

Python is a high-level, object-oriented programming language designed intuitive learning. Its dynamic typing and built-in data structures make it great for fast development and smooth integration of software components. With strong support for modularity and code reuse, Python is widely used across various fields, from scripting to large-scale applications (Van Rossum & Python development team, 2018). What stands out most is its versatility in handling data types of interest to the study, such as raster and vector.

Python provides several tools for working with large raster files and performing custom spatial analysis. It allows you to develop code to calculate landscape metrics at the local level and apply block processing approaches to optimize performance. In particular, the NumPy, Rasterio, and Gdal libraries provide functionality that meets the needs of this thesis:

- **NumPy** provides optimized data structures for working with large numeric arrays, facilitating pixel-level raster image processing and fast execution of spatial computations (Harris et al., 2020).
- **Rasterio** simplifies reading, writing, and manipulating geospatial raster data and integrates with NumPy to perform operations without losing spatial reference (Gillies & others, 2013).
- **Gdal** provides advanced tools for managing geospatial data in multiple formats, allowing reprojection, metadata querying, and operations on both raster and vector files (GDAL/OGR contributors, 2025).

#### 2.2.4 ArcGIS Pro

This geographic information system (GIS) software allows users to perform advanced spatial analysis and data visualization in geospatial research (NobleProg, n.d.). ArcGIS Pro provides a robust suite of tools for processing and preparing spatial datasets, supporting the exploration and manipulation of both animal movement data in vector format and land cover data in raster format. Additionally, it simplifies tasks like defining and refining the area of interest. Through its intuitive interface and geoprocessing tools, ArcGIS Pro enables precise spatial analyses that are essential for understanding spatial relationships.

### 2.3 Methods

In this subchapter, the methods are presented in a methodological framework that were used in the study, including the approach used to define the study area for the lapwing breeding season and the criteria for selecting a meaningful land use and land cover (LULC) dataset. It also describes the calculation of landscape metrics and the assessment of collinearity among environmental variables. Lastly, it outlines the process of fitting SSF models and evaluating their performance.

#### 2.3.1 Methodological framework

The methodological framework developed to achieve the two specific objectives is illustrated in Figure 8. The workflow consists of two main sections. The left column of the diagram represents the process of evaluating landscape complexity through the calculation of local landscape metrics. The second section focuses on the design and fitting of SSF models. The key elements in the

diagram include the main processes, their associated subtasks, intermediate outputs, and the final result. Each stage is explained in detail in the following subsections.

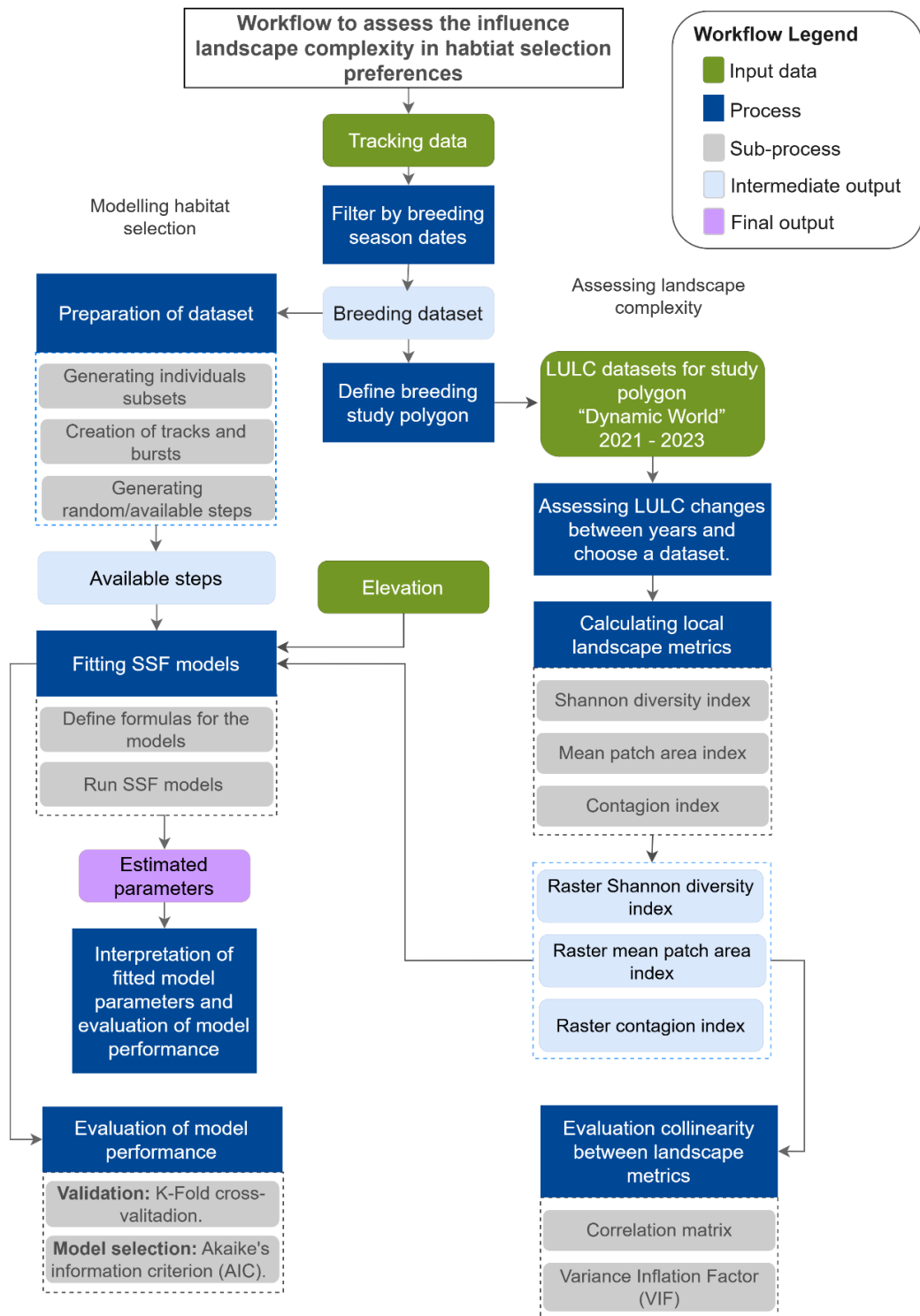


Figure 8. Workflow diagram showing the structure of the methodology used to conduct the study.

### 2.3.2 Defining study area

The analysis was carried out in the Western Europe and Northern Europe sub-regions. This area covers part of the southern United Kingdom, the Netherlands, north-western Germany, northern Belgium and northern France, covering of 251,129 km<sup>2</sup> (See **Error! Reference source not found.**). This area was delimited based on the extent of tracking data collected during the northern lapwing breeding season.

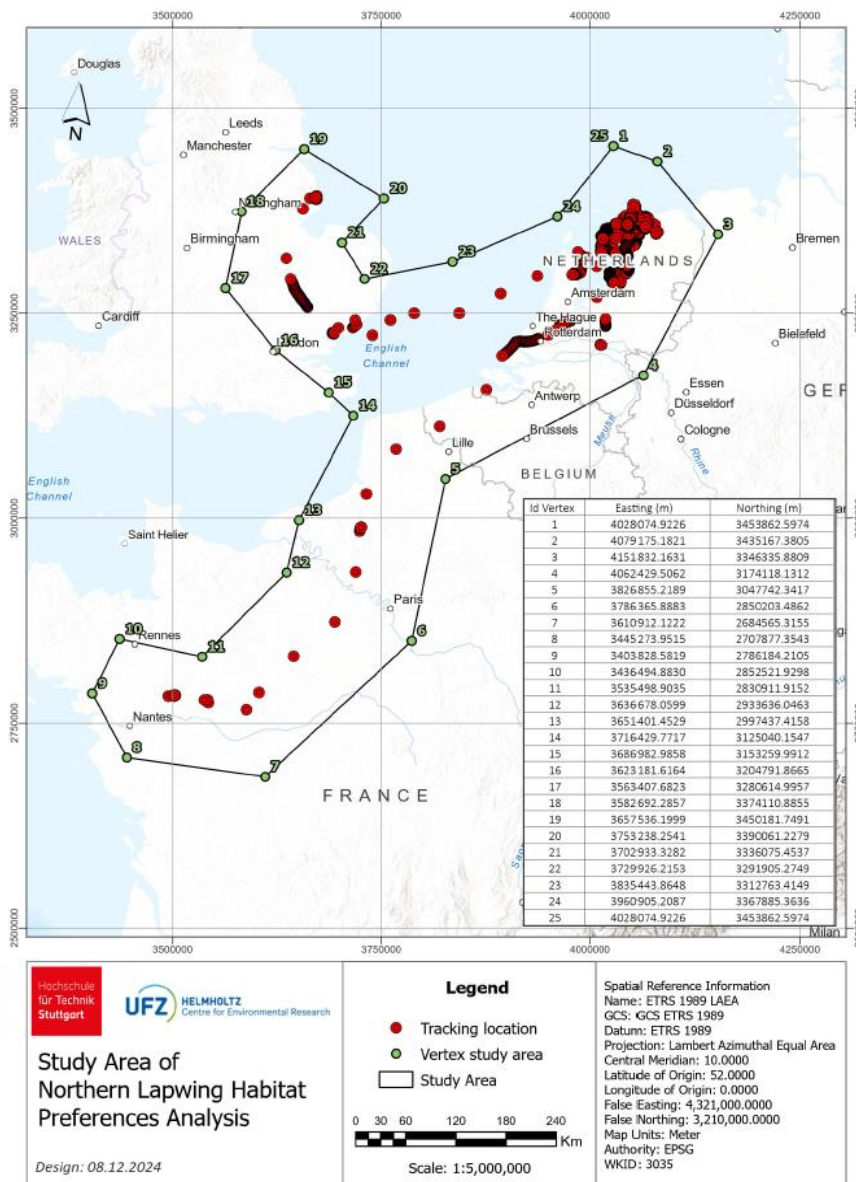


Figure 9. Map of the study area for the northern lapwing habitat preference analysis: The polygon outlines the region of positions recorded for the 13 individuals during breeding season.

The breeding season was the reference for the definition of the study scope. This criterion was defined considering that one of the main causes of population decline is problems in the reproductive aspects of the species' population dynamics (van den Berg, 2024). Some of the most common threats are predation of eggs, chicks and incubating adults (Berg et al., 1992). The breeding period of this species can vary depending on the colony and its area of distribution. According to the behavior of the colonies observed in Switzerland, the breeding season starts in mid to late March and ends in June (Swiss Ornithological Institute, 2023), so these months were used as a timeline reference for the thesis development.

Tracking data recorded in the timeline reference, corresponding to the breeding season were filtered and used for the analysis, excluding positions recorded during the migration season. For this, the *Geopandas* object *GeoDataFrame* (*gdf*) was used in Python, allowing handle the spatial data and filter the rows of the timestamp field "timestamp" that meet the timeline condition, as shown in the script fragment of Figure 10. For the full script, see 8.2 *Appendix. Code developed in python to extract tracking data of breeding season.*

```
filtered_gdf = gdf[
    ((gdf[timestamp] >= '2021-03-15') & (gdf[timestamp] <= '2021-06-30')) |
    ((gdf[timestamp] >= '2022-03-15') & (gdf[timestamp] <= '2022-06-30')) |
    ((gdf[timestamp] >= '2023-03-15') & (gdf[timestamp] <= '2023-06-30'))
]
```

Figure 10. Code snippet developed in Python to filter geospatial data for the breeding seasons (2021-2023), performed in Google Colab.

After filtering the original tracking dataset for the 13 monitored individuals, a refined subset of 206,622 positions was extracted for analysis. This dataset served as a reference for delineating the study area polygon and as input for fitting the SSF models.

### 2.3.3 Assessing changes in land cover and land use.

Assessing the differences between land use and land cover raster data from different years in which tracking data were recorded (2021- 2023) is a key preliminary step in determining what meaningful data to incorporate into the study, while considering the computational cost of processing in the subsequent stages of the analysis. In this context, simplifying the analysis by using a single representative year would optimize resources and avoid redundancies in the results. To address this analysis, a map of differences between the raster maps for each year was generated, allowing visualized and quantified the variations.

This analysis evaluates whether LULC classes change for each pixel over time. This involves verifying whether the class value, which is an integer between 0 and 8, remains the same or changes between 2021, 2022, and 2023. To perform this assessment, each pixel is compared year by year. If the class remains unchanged, the output pixel is assigned a value of 0. If the class changes in any year, the output pixel is assigned a value of 1. This approach allows the precise identification of areas where the landscape has changed, facilitating the interpretation of their relevance for the analysis. The maps resulting from these comparisons are presented in Figure 11.

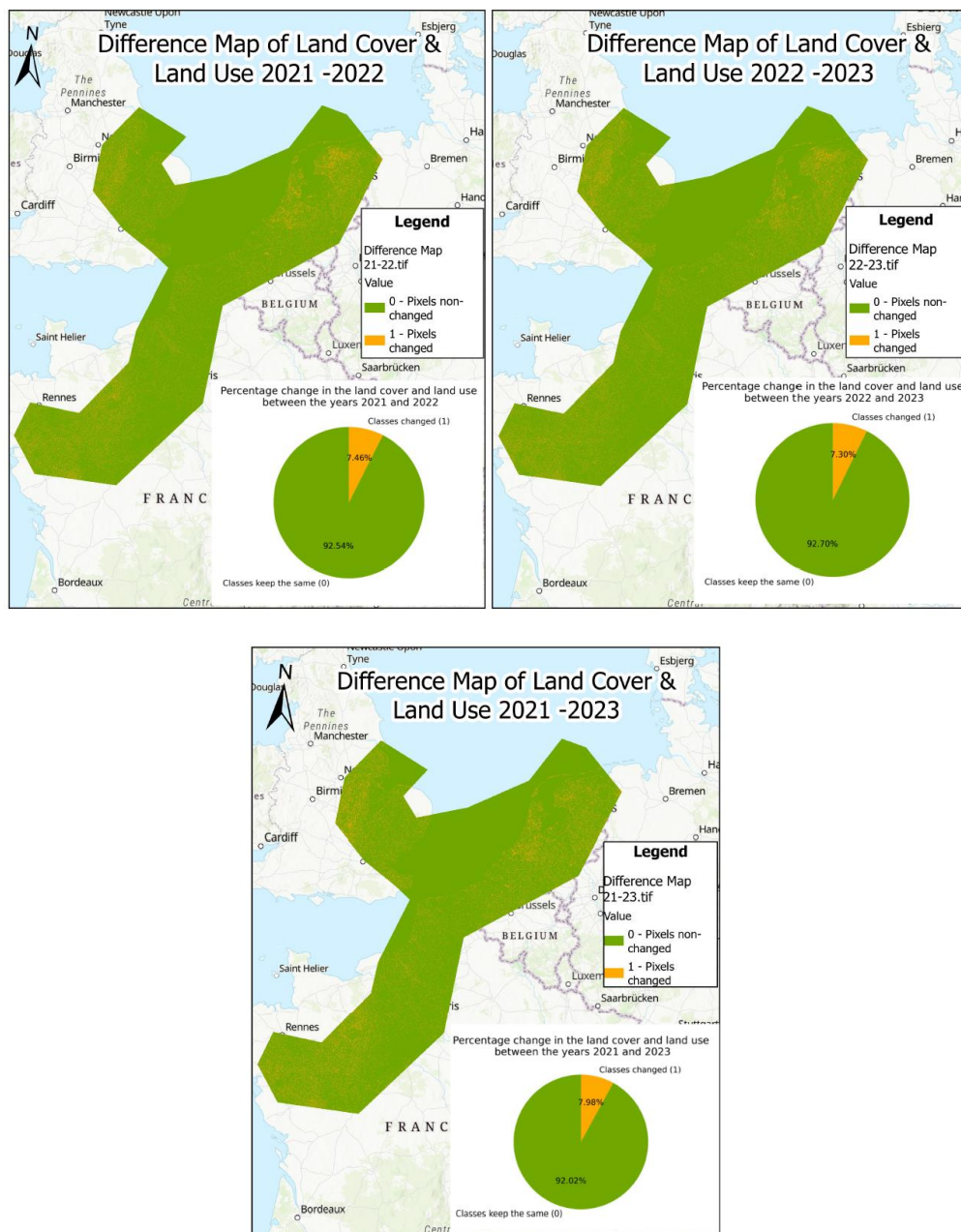


Figure 11. Maps illustrating changes in land cover and land use classes across the study area between 2021, 2022, and 2023: The areas in green represent pixels where the LULC classes remained the same between years and in orange where there were changes.

The analysis of the differences in the land cover and land use raster showed that annual changes accounted for 7.30 to 7.98% of the total pixels. This percentage indicates that variations in the classes represented in each pixel do not generate a significant analytical contribution to the study. Furthermore, such non-significant variations would produce landscape metrics with similar results between years. Consequently, the land cover and land use dataset corresponding to the intermediate year, 2022, was selected as it adequately represents the general characteristics of the study area.

### 2.3.4 Calculating local landscape metrics

The assessment of landscape complexity requires consideration of various aspects, including the analysis of landscape composition, determined by the proportion of each land cover and land use class; landscape configuration, which examines the spatial patterns of these classes; and landscape connectivity (Karimi et al., 2021). One of the most recent approaches to achieving a detailed assessment of landscape complexity involves combining multiple metrics (Machado et al., 2018), providing a comprehensive evaluation of the spatial characteristics of the landscape. This approach was adopted in this thesis, incorporating the analysis of various metrics.

The study focused on selecting metrics that capture key landscape features influencing habitat selection during the breeding season. It considered features such as the distribution of land cover and land use patches, which may be related to ground nest success, distribution, abundance, or predator activity. These factors could ultimately affect the risk of predators locating nests (Bertholdt et al., 2017).

Habitat diversity is essential for the northern lapwing, as this species selects areas with a mix of cover types such as grasslands, crops, and wetlands for nesting and foraging. A preference for diverse habitats likely ensures access to the resources necessary for survival and reproductive success (Horvat & Denac, 2019). For this thesis the landscape metrics Shannon diversity index, mean patch area, and contagion indices were used to analyze landscape complexity, addressing key aspects of habitat composition and configuration. The detailed methodology and application of each metric are presented in the following sections.

### 2.3.4.1 Shannon diversity index (SHDI)

**I. Definition:** This index measures species diversity by calculating the proportion between the number of species and the number of individuals, which for this particular study is the proportion between the number of LULC classes and the number of pixels in the study area (Spellerberg & Fedor, 2003). For a given number of species, diversity has a minimum value when abundance is concentrated in only one species while all others are left with only one individual, and has a maximum value when all species are equally common (Moreno et al., 2011). The Shannon index is determined by Equation 1:

Equation 1. Shannon diversity index.

$$SHDI = - \sum_{i=1}^s p_i \ln(p_i)$$

where,  $p_i$  is the relative abundance of species  $i$ , that is, the abundance of species  $i$  divided by the sum of the abundances of the  $s$  species that compose the community.  $\ln(p_i)$  is the natural logarithm of  $p_i$ ; and the summation ( $\Sigma$ ) extends to  $s$  species (Jost & González-Oreja, 2012). That is to say, the calculation takes the proportion of individuals for each class relative to the total abundance of all classes, multiplying this proportion by the logarithm of itself, and then summing these values for all species.

**II. Implementation:** Google Earth Engine's platform enables the calculation of the Shannon diversity index from the Dynamic World dataset for 2022. This platform provides both a computing environment and a JavaScript code editor to design and run the index code. A  $7 \times 7$  pixel neighborhood was defined to assess local diversity, defined by moving windows with an additional buffer to minimize border effects.

To calculate the index by breaking down the individual components of Equation 1. Shannon diversity index. Firstly, the frequency of each land cover and land use within the defined window is counted using the *reduceNeighborhood* function, which sums the number of pixels corresponding to each class in the neighborhood. These frequencies are then normalized by dividing by the total number of pixels in the neighborhood, which generates the proportions of each of the 9 classes, as proposed in the code snippet in Figure 12. Code snippet developed in JavaScript for computation of class proportions. (see full code in 8.3 Appendix. *JavaScript code for Shannon diversity index calculation in environment GEE.*):

```
// Generate proportions for each class and assign explicit names
var proportions = ee.Image.cat([
  image.eq(0).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(totalPixels).rename('class_0'),
  image.eq(1).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(totalPixels).rename('class_1'),
  ...,
  image.eq(7).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(totalPixels).rename('class_7'),
  image.eq(8).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(totalPixels).rename('class_8')
]);
```

Figure 12. Code snippet developed in JavaScript for computation of class proportions.

From these proportions, the logarithmic expression is applied to derive the Shannon index and the sum is iterated over all land cover and land use classes. The block approach was adopted to make parallel processing feasible and to handle large spatial extents. Within each block, the Shannon Index calculation is looped locally, ensuring that each neighborhood is evaluated independently, the code fragment of Shannon diversity index calculation is presented in the Figure 13. Code snippet developed in JavaScript for Shannon diversity index computation from class proportions. (see the complete code in the 8.3 Appendix. *JavaScript code for Shannon diversity index calculation in environment GEE.*). The result of each block is cropped to its original geometry and prepared for export in GeoTIFF format at high resolution (10 meters).

```
// Calculate the Shannon Index
var shannonIndex = proportions.expression(
  '-1 * (class_0 * log(class_0 + 1e-6) + ' +
  'class_1 * log(class_1 + 1e-6) + ' +
  'class_2 * log(class_2 + 1e-6) + ' +
  ...
  'class_7 * log(class_7 + 1e-6) + ' +
  'class_8 * log(class_8 + 1e-6))', {
  'class_0': proportions.select('class_0'),
  'class_1': proportions.select('class_1'),
  'class_2': proportions.select('class_2'),
  ...
  'class_7': proportions.select('class_7'),
  'class_8': proportions.select('class_8')
});
```

Figure 13. Code snippet developed in JavaScript for Shannon diversity index computation from class proportions.

#### 2.3.4.2 Mean Patch Area index (AREA\_MN)

**I. Definition:** This metric calculates the average size of all patches within a specific land cover class. This metric provides a straightforward approach to characterizing the composition of the landscape. When combined with the total area of the class, it offers valuable insights into the patch structure, such as whether the landscape consists of numerous small patches or a few large one. Mathematically, it is expressed in Equation 2.

Equation 2. Mean Patch Area index

$$AREA_{MN} = \frac{\sum_{i=1}^s A_i}{n}$$

Where  $A_i$  is the area of patch  $i$  and  $n$  is the total number of patches in the class concerned (Hesselbarth et al., 2024). Establish a relationship between the area occupied by a class and the number of fragments corresponding to that class (Subirós et al., 2006). High values indicate that the patches of the analyzed cover class are, on average, large, suggesting the presence of more continuous habitats. On the other hand, low values show that the cover class patches are, on average, small, which may be indicative of a highly fragmented landscape (Midha & Mathur, 2010).

**II. Implementation:** For this index calculation, the programming environment was changed to a Python in Colab Pro+, which provided access to greater computational resources. Following a similar scheme to that employed with the SHDI, the block processing approach was implemented a for all landscape metrics, running the `mpa_by_pixel` (see the code in the 8.4 Appendix. Python (Colab) code for Mean Patch Area index calculation.) function on each of them. In this way, the local habitat could be analyzed using  $7 \times 7$  pixel moving windows allowing the identification and measurement of specific patches.

In order to optimize the spatial characterization and refine the quantification of the size of local patches, a specialized technique was employed for extracting and labelling valid classes. Within each window, these classes are identified using the label function, which allows the area of each patch to be measured in square meters and the average of these areas to be calculated. In order to speed up the processing, the function `mpa_by_block` has been developed (see the code in the 8.4 Appendix. Python (Colab) code for Mean Patch Area index calculation.), which is responsible for traversing the blocks and running the analyses in parallel. Lastly, the results are assigned to a block and exported in GeoTIFF format, thus ensuring a detailed and efficient analysis of the average size of local patches across the entire image.

### 2.3.4.3 Contagion index (CONTAG)

**I. Definition:** The landscape contagion index measures the degree of clumping of attributes on raster maps. This index measures how pixels of the same class tend to cluster in the landscape. When there is clustering or ‘clumping’ in the landscape, i.e. pixels of the same class are together rather than scattered, the contagion index reflects this. The metric works because it is influenced by the greater frequency with which pairs of adjacent pixels that have the same attribute (such as

two forest pixels together) appear. In other words, if many of the same pixels are clustered together, the value of the contagion index will be higher (Riitters et al., 1996). The contagion Index is given by Equation 3. Contagion index.

Equation 3. Contagion index.

$$CONTAG = 1 + \frac{\sum_{i=1}^m \sum_{k=1}^m \left[ p_i \frac{g_{ik}}{\sum_{k=1}^m g_{ik}} \right] \left[ \ln \left( p_i \frac{g_{ik}}{\sum_{k=1}^m g_{ik}} \right) \right]}{2 \ln(m)}$$

Where  $p_i$  corresponds to the proportion of the landscape occupied by patch type (class)  $i$ ,  $g_{ik}$  is the number of adjacencies (joins) between pixels of patch types (classes)  $i$  and  $k$  based on the double-count method.  $m$  is the number of patch types (classes) present in the landscape. That is to say, the index is calculated by combining the proportion of a patch type and the proportion of its adjacencies with other types, multiplied by the logarithm of that value. This sum is then divided by twice the logarithm of the total number of patch types. Contagion values approach 0 when patch types are fully disaggregated (i.e. each cell is of a different patch type) and sparse (with equal proportions of all adjacencies between pairs), and reach 1 when all patch types are fully aggregated (McGarigal et al., 2002).

**II. Implementation:** The methodology implements the function `calculate_contagion` (see the code in the 8.5 Appendix. Python (Colab) code for Contagion index calculation.), which quantifies the spatial connectivity of categorical classes by analyzing their 8-directional adjacencies. Initially, each local window evaluates how many times each category borders with another, thus constructing a contiguity matrix that quantifies the closeness between classes. By normalizing these frequencies, relative probabilities are obtained and subjected to logarithmic transformation, this step allows that the structural complexity of the landscape to be expressed in a synthetic index. Consequently, the result is a quantitative value able to reflect the intensity of the spatial intertwining of the patches.

The connectivity calculation is integrated into a workflow designed to process large amounts of geospatial data. It maintains the technique of processing the image in blocks and, within each block, applies the moving window that enables the extraction and labelling of patches, thus adjusting to the scale of analysis required, while the parallel execution optimizes time.

### 2.3.4.4 Localized landscape metrics through moving windows

The study incorporates a moving window technique to analyze landscape characteristics at a specific scale. In moving window technique, each location is associated with the landscape patterns present within the surrounding spatial window, where the window size determines the scale of the analysis (Hagen-Zanker, 2016). The moving window iterates over each pixel in the image, calculating the landscape metric based on the land cover and land use data within the window. The result of the calculation is assigned to the center output pixel (see Figure 14. The SHDI, AREA\_MN, and CONTAG landscape metrics are calculated within the extent of the moving window, and the results are assigned to the corresponding output pixel for each metric.).

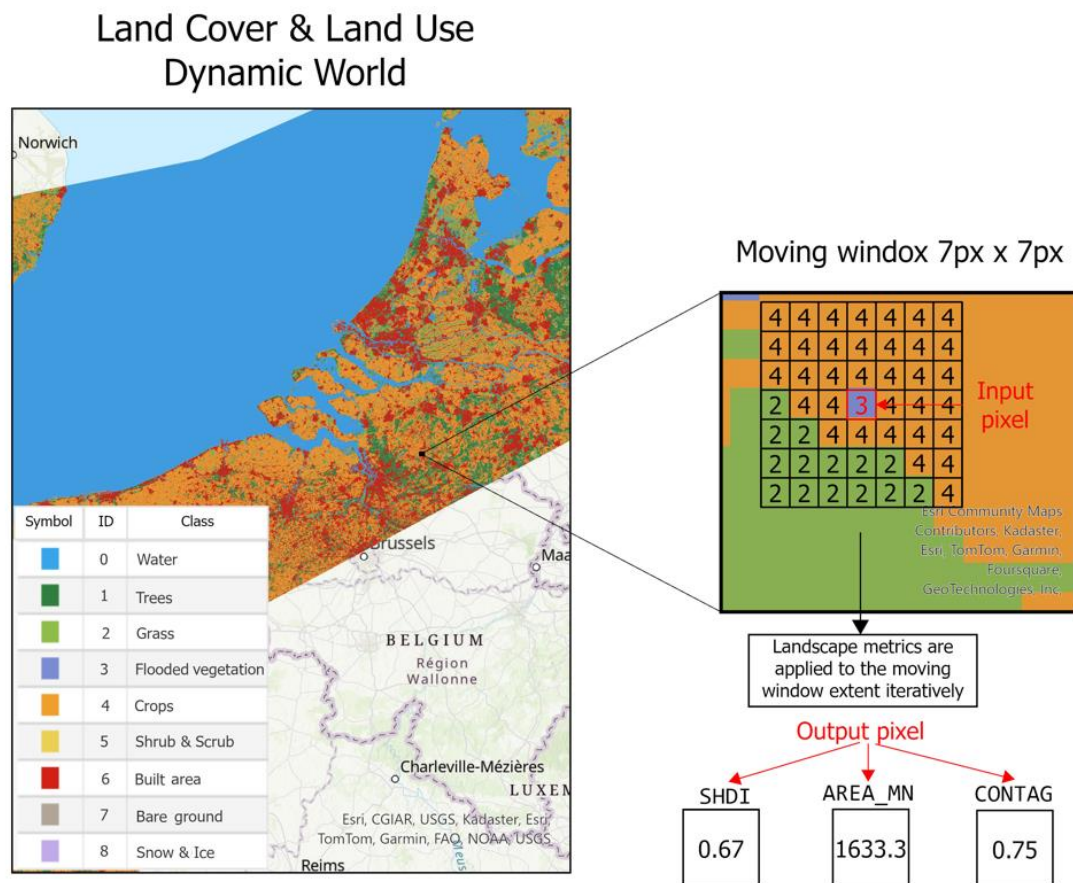


Figure 14. The SHDI, AREA\_MN, and CONTAG landscape metrics are calculated within the extent of the moving window, and the results are assigned to the corresponding output pixel for each metric.

This approach provides a more detailed analysis as it considers the local context of the northern lapwing. To define the size of the moving window, the overall mean step length between positions

for all individuals was used as the criterion. This reference value offers an indicative scale that reflects the bird's immediate decision-making process regarding how far it should move. Thurfjell et al. (2014) analyzed the use of step-selection functions to study resource selection by animals moving through the landscape, indicating that movement behavior is closely linked to habitat perception.

The analysis of the mean step length for the 13 individuals revealed an average of approximately 70 meters. Considering the spatial resolution of the land cover and land use dataset, which is 10 meters per pixel, the moving window size of 7 x 7 pixels was adopted for the calculations. By aligning the window size with the observed movement patterns of the species, the analysis of landscape complexity incorporates the birds' interaction dynamics with their environment in a more representative way.

To calculate local landscape metrics the function to divide the raster into smaller blocks was implemented. This approach ensured efficient computation since the available resources could not process the full 10-meter resolution raster at once. By working with manageable sections, the analysis maintained accuracy while optimizing performance.

### **2.3.5 Assessing collinearity in environmental variables**

In non-experimental research environments, i.e. those where conditions are not directly intervened upon, predictor variables are often correlated with each other (collinearity). This makes difficult to accurately estimate the individual effect of each variable in multiple regression models (Fieberg, 2024), such as the logistic models commonly used in SSF fitting, increasing the magnitude of standard errors. When two variables are closely linked, distinguishing their separate influence on the variable of interest is complex, as changes in one are often accompanied by alterations in the other. These limitations in identifying the specific role of each predictor lead to further reflection on the role of collinearity in the interpretation of the results (Morrissey & Ruxton, 2018).

Models are commonly used to test hypotheses by assessing whether predictors have a statistically significant effect on the variable of interest (i.e. habitats). When there is high collinearity it is possible to estimate coefficients, but with inflated standard errors and small changes in the data can alter the results considerably, as the model becomes unstable and it makes difficult to accurately determine the relative importance of each variable (Dormann et al., 2013). To assess the collinearity between the environmental variables considered in the study, the correlation matrix and Variance inflation factors (VIF) were calculated.

### 2.3.5.1 Correlation matrix for environmental variables

A correlation matrix indicates the linear association (collinearity) between each pair of variables, where the rows and columns of the matrix correspond to specific variables of interest, showing the correlation coefficients between them (Hadd & Rodgers, 2021). This is useful for identifying possible redundancies between predictor variables in statistical analyses, helping to avoid collinearity problems in the models (Juárez, 2022). In this study, the correlation matrix was applied to the variables Shannon diversity index, mean patch area index, contagion index and elevation, as shown in Table 5.

Table 5. Correlation matrix of the predictor variables: Shannon diversity index (SHDI), mean patch area index (AREA\_MN), digital elevation model (DEM) and contagion index (CONTAG).

	SHDI	AREA_MN	DEM	CONTAG
SHDI	1	-0.804	0.093	-0.869
AREA_MN	-0.804	1	-0.084	0.762
DEM	0.093	-0.084	1	-0.069
CONTAG	-0.869	0.762	-0.069	1

The correlation matrix provides important insights about how the predictor variables relate to each other, helping to understand their interaction within the landscape. A high negative correlation is observed between SHDI (diversity) and CONTAG (connectivity), which could indicate that more diverse landscapes tend to be more fragmented and less connected. In contrast, the variable DEM (elevation) shows very low correlations with the others, suggesting that, statistically, it does not strongly influence the landscape structure. This information is fundamental for the definition and fitting of the SSF model, allowing to define a strategy for the selection of the most relevant predictor variables and to minimize the effects of multicollinearity.

### 2.3.5.2 Variance inflation factors (VIF) of environmental variables

A complementary path for estimating multicollinearity is the calculation of variance inflation factors (VIF), which assesses how much the variance of an estimated regression coefficient increases when predictor variables are correlated. The VIF quantify the impact of strong relationships between predictors, indicating how much the precision of the coefficients is degraded due to this collinearity. It is calculated using the Equation 4.

Equation 4. Equation for the calculation of variance inflation factors (VIF).

$$VIF = \frac{1}{1 - R^2}$$

where  $R^2$  represents the proportion of variability explained by the other predictors in the equation for each independent variable (Robinson & Schumacker, 2009). The VIF of a group of variables is calculated by performing an auxiliary regression where that variable becomes the dependent variable, and the remaining variables act as independent variables (Singh, 2024). For the calculation of the VIF in this study, the SHDI was adopted as the dependent variable (see results in Table 6. Variance inflation factors (VIF) for predictor variables. The SHDI was treated as the dependent variable for the calculation.).

Table 6. Variance inflation factors (VIF) for predictor variables. The SHDI was treated as the dependent variable for the calculation.

Variable	VIF
AREA_MN	2.389
DEM	1.007
CONTAG	2.383

The evaluation of the Variance Inflation Factors (VIF) indicates acceptable levels of col-linearity among the variables used. According to Akinwande et al. (2015), values close to 1, such as the case for DEM, suggest no multicollinearity between the regressors (DEM - SHDI). In contrast, values greater than 1 indicate moderate correlation among the regressors, as observed for SHDI with AREA\_MN and CONTAG. Nonetheless, this analysis must be carefully considered during the fitting of the SSF model to ensure appropriate inclusion of variables and maintain the accuracy of the results.

### 2.3.6 Implementation step-selection function (SSF) model

This section describes the process of application and fitting step-selection functions (SSF) models in R. For this analysis, it mainly used specialized libraries in the study of animal movement, such as amt, move, and terra, described in the chapter on Data and Tools. An individualized approach was adopted, performing an independent analysis for each of the 13 individuals studied. The methodology proposed in this study has been developed following the guidelines set out in the work of *A 'How to' guide for interpreting parameters in habitat-selection analyses* (Fieberg et al., 2021), which provides a robust framework for the analysis of SSF models.

### 2.3.6.1 Data preparation

At this stage the data from movement tracking, the land use and land cover dataset, landscape metrics and the elevation dataset are loaded and prepared:

- I. **Creation of subsets per individual and year:** In the framework of a study involving multiple individuals, it was considered to carry out a separate analysis for each individual. For this, the positions were filtered and stored individually, taking as reference the unique identifier “*id*” (“*ind\_ident*” in the raw data) assigned to each animal. As long as the condition is met that the positions are selected where the value of the *id* column is equal to the value of the individual variable as seen in the chunk of code of Figure 7. (see the complete code in the 8.6 Appendix. R code developed for SSF model fitting.). As a result, it generated subsets of position data and stored individually in RDS (R Data Serialization) format to secure the outputs of the intermediate process steps and to be called up in subsequent steps:

```
# Filter data for the current individual and store it
cat("Filtering data for individual: ", individuo, "\n")
position_ind <- position %>% filter(id == individuo)
saveRDS(position_ind, file = paste0("position_ind_", individuo, ".rds"))
```

Figure 15. R code snippet of the creation of subset of positions per individual using the “*id*” as criteria to filter the data.

- II. **Generation of tracks and bursts:** Tracks integrate spatial coordinates (x and y) with timestamps, transforming raw data in shapefile format into an organized format suitable for modeling and analysis (Signer et al., 2024). As part of the data transformation, the process automatically generates bursts by grouping consecutive points, breaking the tracking data into smaller and continuous trajectories. It assigns the same *burst\_id* to all points within each group.

The R function *make\_track* from *amt* package generates the tracks and bursts from the individual’s positions. Tracks and bursts are fundamental to calculate movement metrics, such as step length and turning angles, which are essential for generating available steps. This structured approach ensures that the spatial and temporal dynamics of movement are accurately captured and ready for detailed analysis (see the chunk of code in Figure 16 and the complete code in the 8.6 Appendix. R code developed for SSF model fitting.).

```
# From the individual data stored locally (rds file), create the tracks and bursts
track_ind <- readRDS(paste0("position_ind_", individuo, "_", year_filter, ".rds")) %>%
# Creating track (track_xyt format)
make_track(.x = x, .y = y, .t = t, id = id, crs = 3035) %>%
# Resample data according to the config per individual
track_resample(rate = minutes(rate), tolerance = seconds(tolerance))
head(track_ind)
# save the data in local
saveRDS(track_ind, file = paste0("track_ind_", individuo, "_", year_filter, ".rds"))
```

Figure 16. R code snippet of the creation of tracks: A differentiated resampling rate and tolerance was configured for each individual.

Along with track creation, the data were resampled to retain only points at regular intervals. The time interval was determined individually for each individual and each year using the median of time differences between consecutive points. A tolerance of 10% of the median was applied as a reference, ensuring a more consistent and standardized dataset for subsequent analysis. Although the literature consulted on implementing SSF models in R does not provide a specific criterion for what reference values to use to define the rate and tolerance of resampling, the reference statistics have been defined as in the Appendix A of the supporting information from Signer et al. (2021).

III. **Generating available steps:** The generation of available steps allows modelling the animal's movement decisions by comparing observed steps with theoretical alternatives. Initially, the *steps\_by\_burst* function converts the generated tracks and bursts into a set of used steps, defining each step as the trajectory between two consecutive points on the track. Then the function *random\_steps* creates random steps associated with each used step considering its step length and turning angles. This approach ensures that the generated alternatives are consistent with the animal's movement patterns (see Figure 17).

```
#From tracks and bursts, create the random steps. 10 random step by 1 used.
steps_ind <- burst_ind %>%
  steps_by_burst(burst_ = burst_) %>%
  random_steps(n = 10) %>%
  group_by(step_id_) %>%
  mutate(
    log_sl_ = log(sl_),
    cos_ta_ = cos(ta_)
  ) %>%
  ungroup() %>%
  filter(!is.na(x2_) & !is.na(y2_) & !is.na(log_sl_) & !is.na(cos_ta_)) |
saveRDS(steps_ind, file = paste0("steps_ind_", individuo, "_", year_filter, ".rds"))
```

Figure 17. R code snippet for generating 10 random (available) steps from used steps. The input consists of previously created tracks and bursts: The movement characteristics *log\_sl\_* (logarithm of the step length) and *cos\_ta\_* (cosine of the turning angles) are calculate.

According to Fieberg et al. (2021), using 10 available steps per observed step is sufficient to robustly interpret the coefficients of predictor variables (i.e. Elevation, Human population density and land use) in the SSF analysis; coefficient estimates stabilize once this threshold is reached. As the number of available steps increases, the estimates remain constant and variability decreases significantly. For this study, this number of available steps has been taken as a reference (10 available steps), maintaining a balance between accuracy and computational efficiency.

### 2.3.6.2 Fitting SSF models

The designed models examine how landscape complexity influences northern lapwing habitat selection. It incorporates LULC, elevation, and landscape metrics to predict the relative probability of select a step. Now, considering the indications of moderate collinearity between the landscape metrics reviewed in section 2.3.5 *Assessing collinearity in environmental variables* and under the assumption that one of these variables could explain the behavior and influence on the response variable of the other landscape metrics, two more models are proposed as follows (see Figure 18):

```
# Fit a SSF model for each landscape metric
models <- list(
  model_1 = steps_sf %>% fit_issf(case_ ~ landcover + shannon + elevation + log_sl_ + cos_ta_
    + strata(step_id_), model = TRUE),
  model_2 = steps_sf %>% fit_issf(case_ ~ landcover + mean_patch_area + elevation + log_sl_ + cos_ta_
    + strata(step_id_), model = TRUE),
  model_3 = steps_sf %>% fit_issf(case_ ~ landcover + contagio + elevation + log_sl_ + cos_ta_
    + strata(step_id_), model = TRUE)
)
```

Figure 18. Fitting of SSF models using the function in *fit\_issf* in R: For each landscape metric a separate model is fitted.

In addition to the predictor variables it includes *strata(step\_id\_)* which indicates the stratum, that is the grouping that matches each observed step with its corresponding available steps, avoiding treating all steps as independent. Also, properties of the movement such as *log\_sl\_* (logarithm of the step length) and *cos\_ta\_* (cosine of the turning angles) are included in the model (Fieberg et al., 2021).

The *fit\_issf* function fits a conditional logistic regression model that compares steps used with steps available in relation to predictor variables. The *fit\_issf* function estimates the relative probability of selecting a step based on environmental characteristics compared to the alternatives

generated (Signer et al., 2024). The logistic equation has the following form (see Equation 5) (Fieberg et al., 2021):

Equation 5. Logistic regression equation in logit form.

$$\text{logit}(p_i) = \log \left[ \frac{p_i}{1 - p_i} \right] = \beta_0 + \beta_1 X_{i1} \dots \beta_k X_{ik}$$

Where  $\log \left[ \frac{p_i}{1 - p_i} \right]$  is the logit transformation of the probability, where  $p_i$  represents the probability that a resource is used,  $\beta_0, \beta_1, \dots, \beta_k$  are the regression parameters and  $X_{i1} \dots X_{ik}$  are the predictor variables. To calculate the probability between 0 and 1, the exponent of the logit value is calculated Equation 6:

Equation 6. Inverse-logit transformation.

$$p_i = \frac{\exp(\beta_0 + \beta_1 X_{i1} \dots \beta_k X_{ik})}{1 + \exp(\beta_0 + \beta_1 X_{i1} \dots \beta_k X_{ik})}$$

This equation converts the linear combination of predictor variables and their coefficients into interpretable probabilities, which allow the assessment of the animal's habitat preferences in relation to the predictor variables (Fieberg et al., 2021).

### 2.3.7 Model performance

It is important to evaluate the performance of SSF models to ensure their accuracy and applicability in ecological studies. Carrying out an evaluation allows for identifying the need for model adjustments and ensures that the model accurately reflects the observed movements of individuals. Yates et al. (2022) emphasize that cross-validation is key to selecting models in ecology, as it ensures that inferences are reliable and reproducible. In addition, Burnham (2011) highlights the ability of information-theoretic (I-T) approaches, such as Akaike's information criterion (AIC), to effectively handle complex problems and perform formal inference on a model set.

#### 2.3.7.1 K-Fold cross-validation

K-fold cross-validation is a technique that helps to evaluate how robust the predictive capability of a model is. This method randomly divides the dataset into k independent parts. Then, it uses k - 1 parts to fit the model and evaluates its performance on the remaining part (Hirzel et al., 2006).

To measure the performance of the fitted SSF models, a 10-fold cross-validation was implemented integrating the metric Area under the Curve (AUC). The AUC is a global measure of the ability of a test to discriminate whether a specific condition is present or not present (Hoo et al., 2017), in the context of SSF model, whether the model can discriminate between selected and non-selected steps.

In the implementation of the R cross-validation code, the per-individual evaluation approach has been kept using loops to access the set of steps used and available. For the 9 training subsets the 3 models are fitted and the coefficients obtained from these fits are used to predict selection scores in a log-probability scale, as seen in the code snippet in Figure 19. These scores are then exponentiated to convert them into relative probabilities, which indicate the relative likelihood of each step being chosen compared to alternative steps (Michelot et al., 2023).

```
test_data <- test_data %>% mutate(  
  # Generate step-selection predictions using the coefficients of trained models  
  # predict() ---> returns values in log-probability scale.  
  relative_prob = exp(predict(fold_model_clogit, newdata = test_data, type = "lp"))  
  # exp() transforms log-probabilities into relative probabilities  
  # These values indicate the likelihood (relative) of selecting a step compared to others.  
)
```

Figure 19. Code snippet taken from the k-Fold cross validation for SSF models in R: This segment predicts the probability of selection of a step by comparing it with available options.

From the relative probabilities predicted in the previous stage, the code calculates the AUC in the segment shown in Figure 19. The `roc()` function ranks these probabilities from highest to lowest, separating used steps from available steps (Fawcett, 2006). Then the AUC is extracted from the ROC (Receiver Operating Characteristic) object using the `$auc` attribute, representing the proportion of times a used step is ranked higher than an available step, summarizing the model's ability to differentiate between them. See the 8.7 Appendix. *R script to perform K-Fold cross validation of SSF model fitting, using AUC metric.* for details.

### 2.3.7.2 Akaike's information criterion for model selection

The Akaike's information criterion (AIC) is a widely used method for comparing the quality of various models. The objective of model selection with AIC is to estimate the loss of information that occurs when the actual model that generated the data is approximated by the model being evaluated (Wagenmakers & Farrell, 2004). The AIC is made up of two elements: a penalty function, which favors parsimony, and the maximum likelihood function, which measures how well the model fits the data (Hughes & King, 2003). Here, the principle of parsimony is a criterion

used to identify the model that explains the most variation with the fewest variables (Boyce et al., 2002, p. 20). The model with the smallest AIC over the set of models is preferred (Hastie et al., 2009).

To calculate the AIC of the fitted models, R provides a generic function to perform the calculation. The function *AIC()* calculates *Akaike's 'An Information Criterion'* for one or several fitted model objects for which a log-likelihood value can be obtained, according to the formula  $-2 * \log - likelihood + k * npar$ , where *npar* represents the number of parameters in the fitted model, and  $k = 2$  for the usual AIC (R Core Team, 2024). To perform the AIC calculation with the generic function, a script in R was developed to call the outputs of the fitted models for each individual per year and saves the calculation to a local file, see 8.8 *Appendix. R script to perform AIC calculation iteratively in a set of fitted models using the generic function AIC()*.for details.

## 3 Results

### 3.1 Landscape complexity assessment

The complexity of the landscape in the study area was evaluated using the metrics Shannon diversity index, mean patch area index and the contagion index. These metrics allowed us to analyze both the variety and the spatial structure of the resources available in the study area. This analysis was carried out from two perspectives: first, through a global calculation of landscape metrics and, second, from a local perspective. Both approaches form the basis for subsequent phases of the study on lapwing habitat preferences.

#### 3.1.1 Global landscape metrics

The global metrics were calculated considering the entire study area, providing an overview of the landscape structure. Given that the range of values that the Shannon diversity index can take varies between 0 and the natural logarithm of the number of classes (Peet, 1975), that is  $\ln(9) \approx 2.19$ , the value obtained for the global index was 1.3 (see Table 7), indicating that the area is characterized by an intermediate diversity in terms of the land use and land cover classes included in the calculation. This result not only suggests an intermediate richness of classes in the area, but also an equitable relative representation of each of them.

Table 7. Global landscape metrics and mean patch area index for each LULC in the study area for the year 2021 (spatial resolution: 10 meters).

LCLU Class	Landscape Metric		
	SDHI	AREA_MN	CONTAG
<i>Water</i>	1.37	192.0	0.56
<i>Trees</i>		14.1	
<i>Grass</i>		8.50	
<i>Flooded vegetation</i>		3.01	
<i>Crops</i>		142.0	
<i>Shrub &amp; Scrub</i>		3.12	
<i>Built area</i>		14.5	
<i>Bare ground</i>		6.80	
<i>Snow &amp; Ice</i>		2.07	

Furthermore, the mean patch area index shows that the Water and Crop classes have the largest patches, with mean values of 192 m<sup>2</sup> and 142 m<sup>2</sup>, respectively. In the particular case of the *Water class*, this result is to be expected, since a significant part of the study area includes the North

Sea. Part of the North Sea was included because, based on the tracking data of this study, northern lapwings move frequently from the Netherlands to the southeastern coast of England.

Important classes for lapwings during the breeding season, such as *Grass* and *Flooded vegetation classes*, exhibited smaller mean patch sizes. It is possible that the index values are influenced by extreme cases that inflate or reduce the final values. To complement the previous two landscape metrics, the Contagion Index obtained a value of 0.56. On the 0 to 1 interpretation scale, this result indicates moderate connectivity and clustering between landscape classes, suggesting that, in the overall area, there is a balance between clustering and fragmentation of land use and land cover classes. This value also could indicate that the homogeneous landscape mosaic is interspersed with possibly transformed areas, such as crops, pastures and urbanized areas.

### 3.1.2 Local landscape metrics

In addition to calculating metrics to globally assess landscape complexity in the study area, a localized calculation approach of landscape metrics was adopted. This approach provides pixel-by-pixel values of the metrics at a specific scale, with the objective of capturing the perception of the immediate environment of the lapwings. Although the interactions between the species and its environment are the result of complex physiological processes and external factors that are difficult to reduce to a single value, the movement of individuals is dependent on their perception of their immediate surroundings. In this study, the temporal resolution of observations was used as a criterion, and specifically, the mean distance between consecutive steps, which corresponds to a  $7 \times 7$  pixel window ( $70 \text{ m} \times 70 \text{ m}$ ), as a reference.

These local scale landscape metrics allow to obtain the of Shannon diversity, mean patch area, and contagion indices values for each used and available step, considering a surrounding extent of  $4,900 \text{ m}^2$ . These values are extracted and used in the SSF model fitting process. As explained in the *Method* chapter, a block processing approach was implemented to handle the high computing cost of the calculation. In this process, the LULC raster was divided into blocks of  $5,000 \times 5,000$  pixel subsets, with each block processed separately before merging the subsets to reconstruct the raster for each index.

#### 3.1.2.1 Local Shannon diversity index

The assessment of the diversity index in the study area, calculated at the local scale pixel by pixel, is shown in the map in Figure 20. The results indicate that the index reached a maximum value of 1.95 and a minimum of 0, showing the variability in the distribution of diversity. Areas in blue

represent pixels with the highest diversity of classes and resources, while light green indicates areas with low resource variety. The raster generated from this calculation was incorporated as a variable in one of the SSF models, described in section 3.2.2 *The effect of landscape diversity on the habitat use of the northern lapwing.*

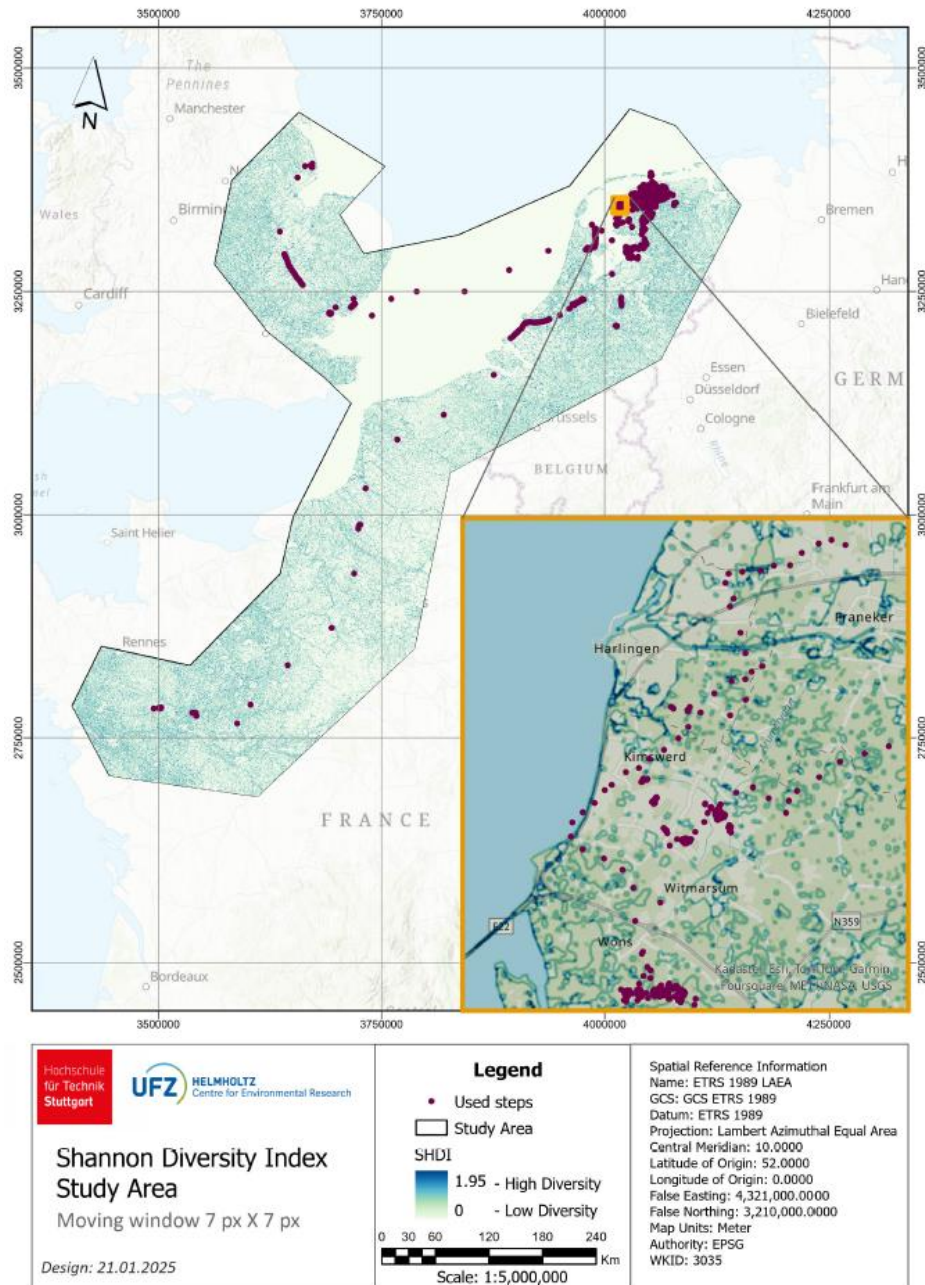


Figure 20. Shannon diversity index - SHDI (local scale) in the study area generated with the moving window approach (7 × 7 pixel) in Google Earth Engine.

The frequency calculation based on intervals, as shown in Table 8, revealed that 85.08% of the pixels have an index between 0 and 0.6, while 14.76% fall within the 0.6 to 1.3 range. These results indicate that, for the most part, LULC classes diversity is low to moderate when the index is calculated over a  $70 \times 70$  meter area. In contrast, only 0.15% of the pixels reflect a high diversity of resources.

Table 8. Distribution table of Shannon diversity index (SHDI) ranges: Absolute frequency (pixel count) and relative frequency (percentage).

Shannon diversity index intervals	Absolute Frequency (number of pixels)	Relative Frequency (%)
$\geq 0$ SHDI $< 0.6$	1,830,427,058	85.08%
$\geq 0.6$ SHDI $< 1.3$	317,595,051	14.76%
$\geq 1.3$ SHDI $\leq 1.95$	3,270,243	0.15%
<b>Total</b>	<b>2,151,292,352</b>	<b>100%</b>

### 3.1.2.2 Local mean patch area index

The evaluation of the average patch size of LULC revealed that the minimum recorded value is  $100 \text{ m}^2$ , equivalent to the size of a single  $10 \times 10$  meter pixel. In contrast, the maximum observed value is  $4,900 \text{ m}^2$ , corresponding to the limit defined by the  $70 \times 70$  meter extent window. These results define the range within which patch sizes vary across the study area. See the Figure 21. Mean patch area index – AREA\_MN (local scale) in the study area generated with the moving window approach ( $7 \times 7$  pixel) in Python Colab +. to visualize the raster generated from the index calculation. The map represents high mean patch area indices in reddish-brown, highlighting areas where patch sizes are larger. In contrast, light reddish-brown or white areas indicate regions with smaller average patch sizes. Further details on its integration into the SSF model fitting are provided in section 3.2.3 *The effect of the patch size on the habitat use of the northern lapwing.*

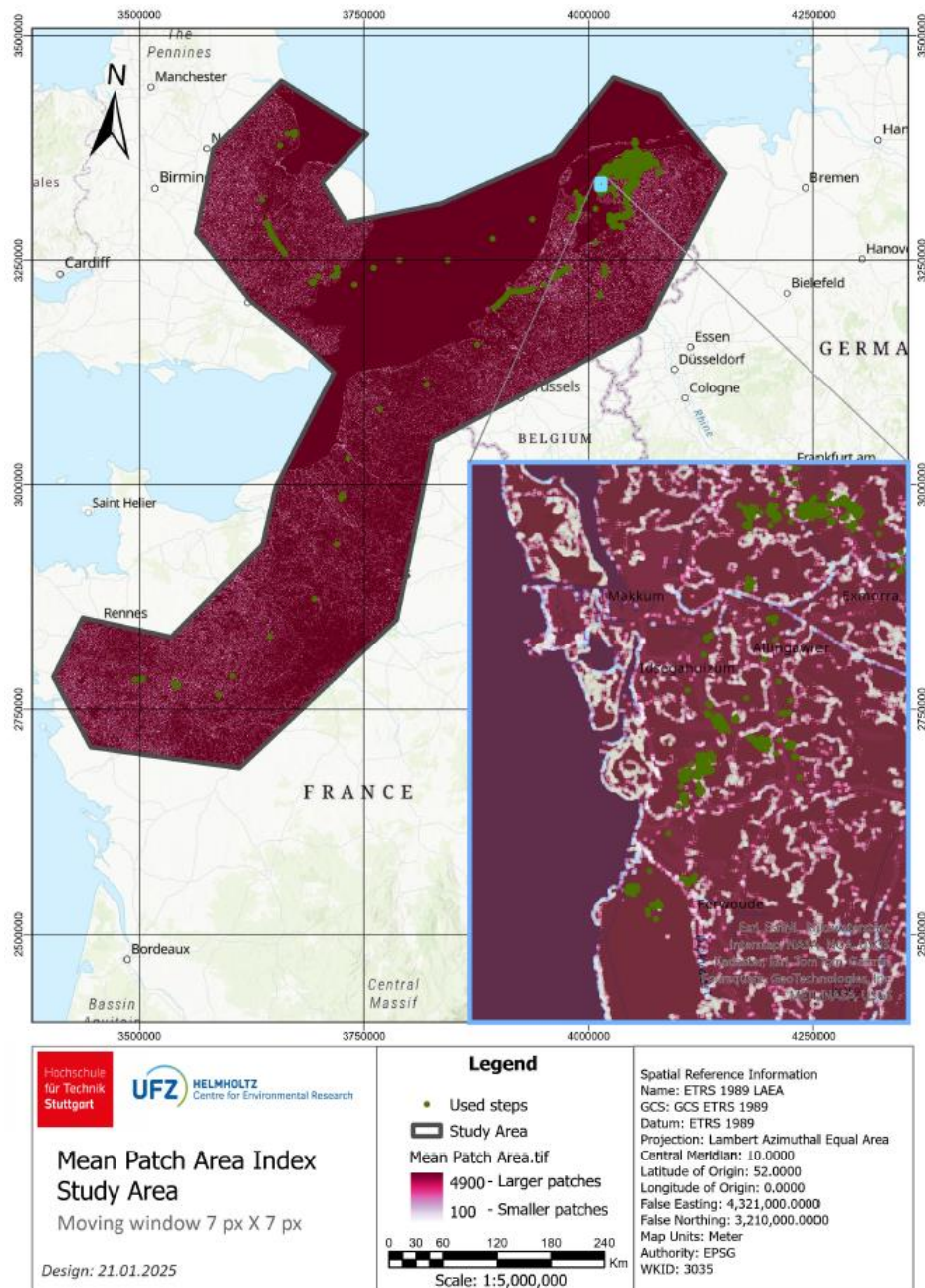


Figure 21. Mean patch area index – AREA\_MN (local scale) in the study area generated with the moving window approach (7 × 7 pixel) in Python Colab +.

The distribution of mean patch area index (AREA\_MN) values shows a clear pattern in patch sizes, where each patch is defined by the continuity of a single LULC class. Only 0.01% of pixels fall within the 1,634 to 2,450 m<sup>2</sup> range, while 12.70% of pixels are in patches that have a mean area between 0 and 1,634 m<sup>2</sup>. In contrast, the majority of pixels (87.29%) are in a range of patches

with a mean area from 2,450 to 4,900 m<sup>2</sup>. These results indicate that, within the extent defined by the 4,900 m<sup>2</sup> moving window, the predominant patches are large at this scale (see Table 9).

Table 9. Distribution table of mean patch index (AREA\_MN) ranges: Absolute frequency (pixel count) and relative frequency (percentage).

Mean patch area index intervals	Absolute Frequency (number of pixels)	Relative Frequency (%)
≥ 0 AREA MN < 1634	273,036,645	12.70%
≥ 1634 AREA MN < 2450	165,175	0.01%
≥ 2450 AREA MN ≤ 4900	1,876,777,812	87.29%
<b>Total</b>	<b>2,149,979,632</b>	<b>100%</b>

### 3.1.2.3 Local contagion index

For the local contagion index (CONTAG), the resulting raster has between 0.2 and 1, this last is the highest value that the index can reach. Figure 22. displays the map with the resulting raster and the range of values present in the study area. The contagion index map represents higher index values in green, indicating areas where class patches are more aggregated. In contrast, lower index values are shown in brown, corresponding to homogeneous landscapes. Further details on how this output is used in model fitting can be found in section 3.2.4 *The effect of the aggregation of patches on the habitat use of the northern lapwing.*

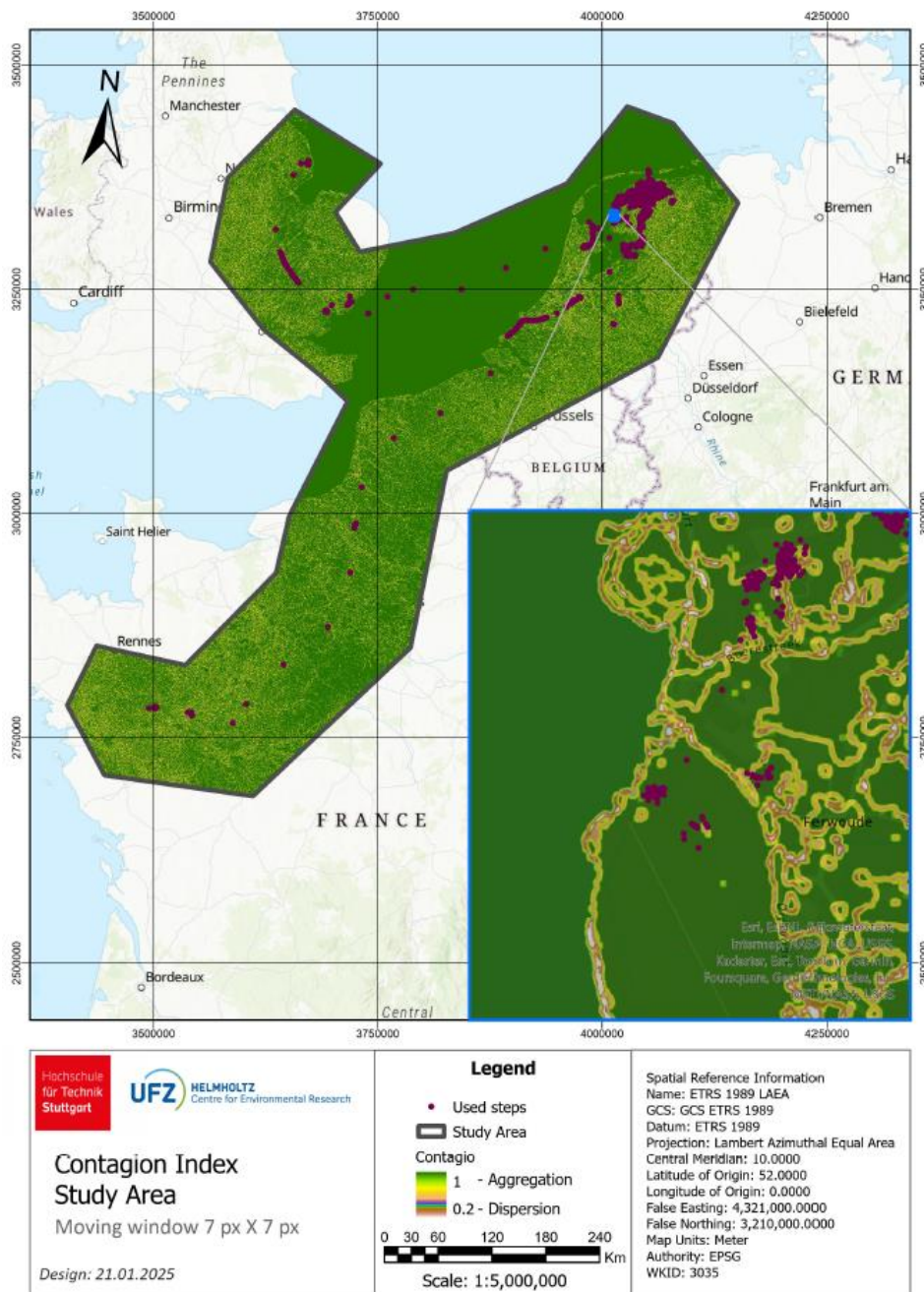


Figure 22. Contagion index - CONTAG (local scale) in the study area generated with the moving window approach (7 × 7 pixel) in Python Colab +.

For the contagion index, the data distribution table (see Table 10) shows that 95.78% of the pixels fall within the 0.7 to 1 range. This indicates that, at the evaluation scale, the study area is predominantly composed of landscapes where LULC classes are more aggregated, forming large patches with minimal fragmentation. In contrast, only 4.22% of the pixels fall below the 0.7 threshold.

Table 10. Distribution table of contagion index (CONTAG) ranges: Absolute frequency (pixel count) and relative frequency (percentage).

Contagion index intervals	Absolute Frequency (number of pixels)	Relative Frequency (%)
$\geq 0$ CONTAG $< 0.3$	3,621	0.0002%
$\geq 0.3$ CONTAG $< 0.7$	90,754,403	4.2213%
$\geq 0.7$ CONTAG $\leq 1$	2,059,167,488	95.7786%
<b>Total</b>	<b>2,149,925,512</b>	<b>100%</b>

### 3.1.3 Used steps distribution in landscape metrics

As a first insight, the steps used per individual for the Shannon diversity index reveals that individuals more frequently utilize sites where landscape diversity tends to be low. It is notable that most individuals are clustered around index values below 0.2 (see Figure 23), including those with the highest number of steps within the group.

Mean Shannon Diversity Index per Individual at Used Steps

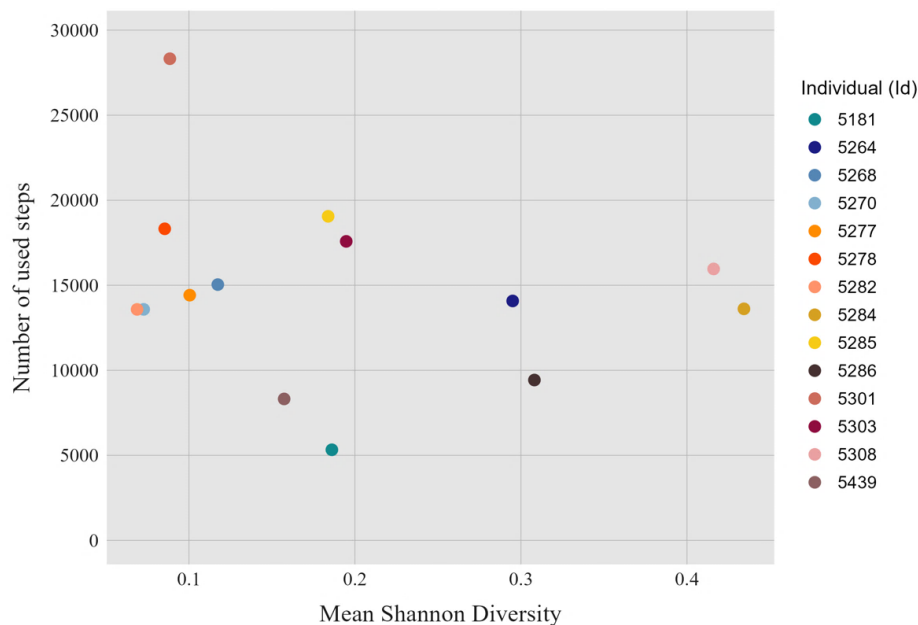


Figure 23. Distribution of mean Shannon diversity index across used steps per individual. Each individual is identified by the “id” number in the legend on the right.

Regarding the mean patch area index, there is a tendency for individuals to select positions where the mean patch size is large. Given that the range of values of this index varies between 100 and 4900 m<sup>2</sup>, corresponding to the size of the moving window, most individuals are concentrated in

values above 3500 m<sup>2</sup> (see Figure 24). This suggests a preference for homogeneous landscapes characterized by large, continuous patches of a single LULC classes.

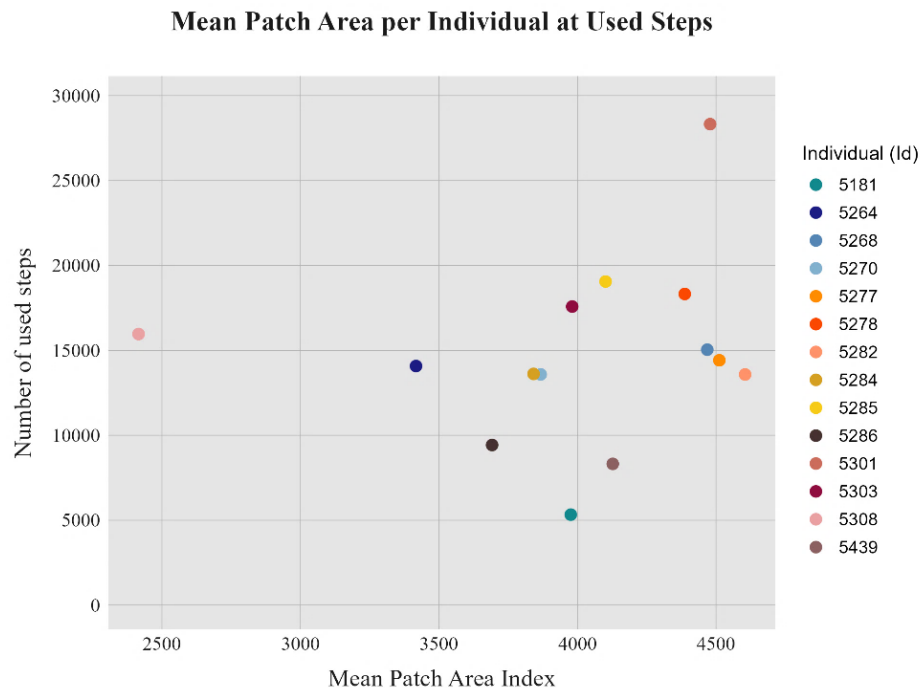


Figure 24. Distribution of mean patch area index across used steps per individual. Each individual is identified by the “id” number in the legend on the right.

The contagion index shows a strong tendency for individuals to select sites with index values close to 1 (see Figure 25). This is the highest value of the index and suggests a clear preference on the part of the group of individuals for highly contiguous or homogeneous landscapes, characterized by compact and contiguous blocks of a single class.

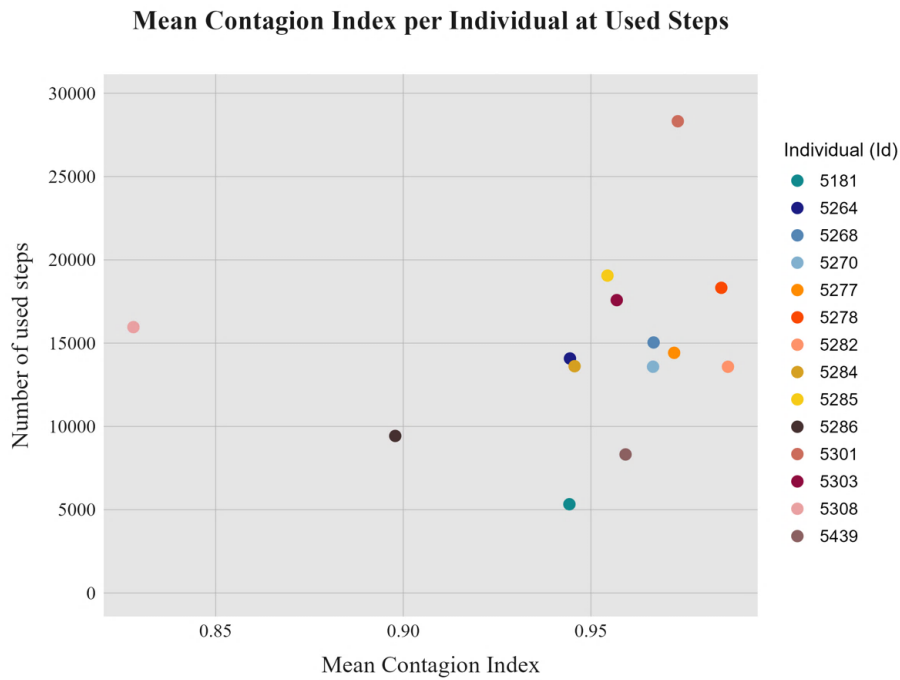


Figure 25. Distribution of Contagion index across used steps per individual. Each individual is identified by the “id” number in the legend on the right.

### 3.2 Estimation parameters of step-selection functions

A separate modeling was conducted for each landscape metric for 2021 and 2022. Although, initially fitting the 3 models was considered with the data available for the individual with id 5301 in the year 2023, the data were not sufficient (steps used = 18) for the generation of the bursts needed in the creation of available steps. A total of 60 models were fitted, from which the standard output provided the coefficients of the covariates and inference statistics.

The estimated parameters of the habitat variables allow the relative selection strength of these resources to be established (Fieberg et al., 2021). This study analyzed the estimated parameter  $exp(coef)$ , which represents the relative selection strength (RSS). More precisely, it measures the effect of a variable change on the selection of a resource or habitat in relation to another (Avgar et al., 2017). The estimated parameters for all fitted models are examined with more detail in the following sections.

### 3.2.1 General findings

There is evidence of an intense relationship between the relative probability of selecting habitats with the presence of the *Grass class*. Although other classes, such as *Flooded Vegetation*, *Shrub & Scrub*, and *Crops classes*, have a smaller effect, they still play a significant role in this probability. At the same time, the *Built Area class* remains as a reference class for the estimation. Furthermore, individuals appear to avoid water bodies and forests. Regarding elevation as a predictor variable, it can be inferred that its influence is not decisive, at least in explaining the relative habitat selection within the reference dataset.

The estimators obtained for the overall landscape metrics reveal, at least in the dataset, a relative preference for homogeneous landscapes, such as open fields where one LCLU class (e.g., *Grass*) predominates. These landscapes are spatially configured in a simple way, characterized by low fragmentation, minimal edge density, and contiguous patches, which could be associated with more predictable and specific resources required during the breeding season for lapwings.

### 3.2.2 The effect of landscape diversity on the habitat use of the northern lapwing

The model 1, which considers the Shannon diversity index of the landscape following the structure (see Equation 7).

Equation 7. Model 1 structure that includes the Shannon diversity index “*shannon*” and the environmental variables: *land cover*, *elevation*, *log\_sl*, *cos\_ta* and *step stratification*.

$$\mathbf{model\ 1} = \mathit{steps} \sim \mathit{landcover} + \mathit{shannon} + \mathit{elevation} + \mathit{log\_sl} + \mathit{cos\_ta} \\ + \mathit{strata}(\mathit{step\_id\_})$$

Where *steps* refers to the set of used and available steps, and *landcover* refers to LULC classes, the results show that, for most variables the relative resource selection preferences of individuals are consistent, with some variations between the years 2021 and 2022. The Figure 26 shows that within the LCLU classes, there is a higher intensity of areas use with *Grass class* presence. In 2021, the SSF for the *Grass class* ranged between 70.68 and 75.95 for 11 individuals, using *Built area class* as the reference class. For two individuals, the RSS values were higher, ranging from 116.17 to 187.78, while one individual had a value of 57.70. In all cases, these RSS values were the most representative within the LULC classes considered (supporting information in 8.1 Appendix. Standard outputs of SSF fitted models.).

In 2022, tracking data was available for only six individuals. Among them, four showed the highest RSS values for the *Grass class*, ranging from 17.40 to 14.26. One individual had the highest RSS magnitude in the *Flooded vegetation class*, with a value of 9, while another had the highest RSS magnitude in the *Tree class*, reaching 16.07. In summary, 18 out of the 20 models fitted for 2021 and 2022 using the Shannon diversity index as an environmental variable showed a higher RSS for areas classified as *Grass* in the LULC dataset.

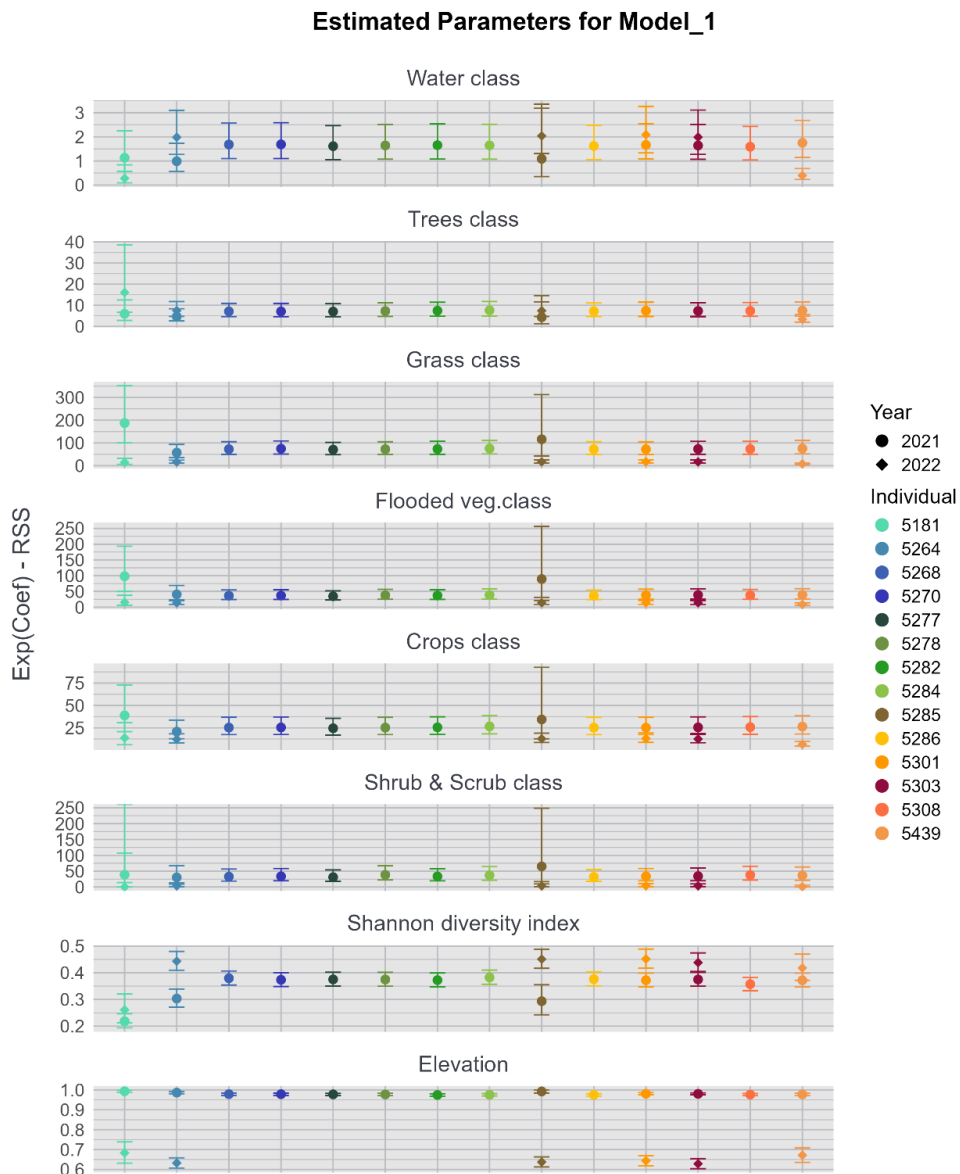


Figure 26. **Estimated parameters in fitted SSF model 1 (landscape diversity) per individual for the years 2021 and 2022.** The x-axis displays the ranges of relative selection strength (RSS) values for all variables considered in Model 1. Each color represents an individual, as indicated

in the legend on the right. Data from 2021 is represented by circles, while data from 2022 is represented by diamond symbols.

Across the 20 fitted models, the *Water class* had the lowest RSS magnitude, indicating the weakest relative selection strength. In general, other classes with a meaningful but lower selection strength than *Grass* included *Flooded Vegetation*, *Crops*, and *Shrub & Scrub*. The inference metric p-value (see supporting information in 8.1 Appendix. Standard outputs of SSF fitted models.) indicated that all LULC classes were statistically significant for the models, using the conventional threshold of  $\alpha = 0.05$ , except for the *Water class*. In three out of the 20 models, the p-value for *Water class* was higher than 0.05, indicating a weaker statistical significance for the models.

In all the models fitted for 2021 and 2022, the Shannon diversity index had the lowest RSS values among all variables, ranging from 0.22 to 0.45. This indicates a weak intensity in the relative selection of habitats by the northern lapwing group. These results suggest a positive relationship between habitat selection and landscapes with lower LULC class diversity, which characterize more homogeneous habitats.

For the elevation variable, the models fitted with 2021 data showed an RSS magnitude of 1, indicating no effect on the relative selection of a habitat. In contrast, the models fitted with 2022 data showed an RSS value of 0.6 for the six individuals, suggesting a weak and negative effect on habitat selection. This suggests that, an increase in the elevation represents a slightly relative risk for the individuals to select a habitat.

### 3.2.3 The effect of the patch size on the habitat use of the northern lapwing

In this model, where the mean patch area index is considered and expressed below in the Equation 8:

Equation 8. Model 2 structure that includes the mean patch area index “*mean\_patch\_area*” and the environmental variables: *land cover*, *elevation*, *log\_sl*, *cos\_ta* and *step stratification*.

$$\mathbf{model\ 2} = \mathit{steps} \sim \mathit{landcover} + \mathit{mean\_patch\_area} + \mathit{elevation} + \mathit{log\_sl} + \mathit{cos\_ta} + \mathit{strata}(\mathit{step\_id\_})$$

Where *steps* refers to the set of used and available steps, and *landcover* refers to LULC classes, the RSS magnitudes obtained for the LULC classes are similar to the results from Model 1 (see

supporting information in 8.1 Appendix. Standard outputs of SSF fitted models.). For the models fitted with 2021 data (see Figure 27.), the highest RSS values across all models corresponded to the *Grass class*. Eleven individuals had values ranging from 72.90 to 78.31, two individuals showed values between 115.85 and 189.86, and one individual had a value of 58.80. These results indicate a higher strength in the relative selection for areas with *Grass class*.

For the models fitted with 2022 data, four out of six individuals had the highest RSS measure in the *Grass class*, with values ranging from 17.41 to 18.52. The last two individuals showed higher RSS value in the *Flooded Vegetation class*, with values between 9.49 and 16.25. Similar to the models in section 3.2.3 *The effect of landscape diversity on the habitat use of the northern lapwing*, these results suggest an increased in the relative probability of individuals selecting habitats mostly containing the class *Grass* and, but with lower preference, habitats with *Flooded Vegetation*.

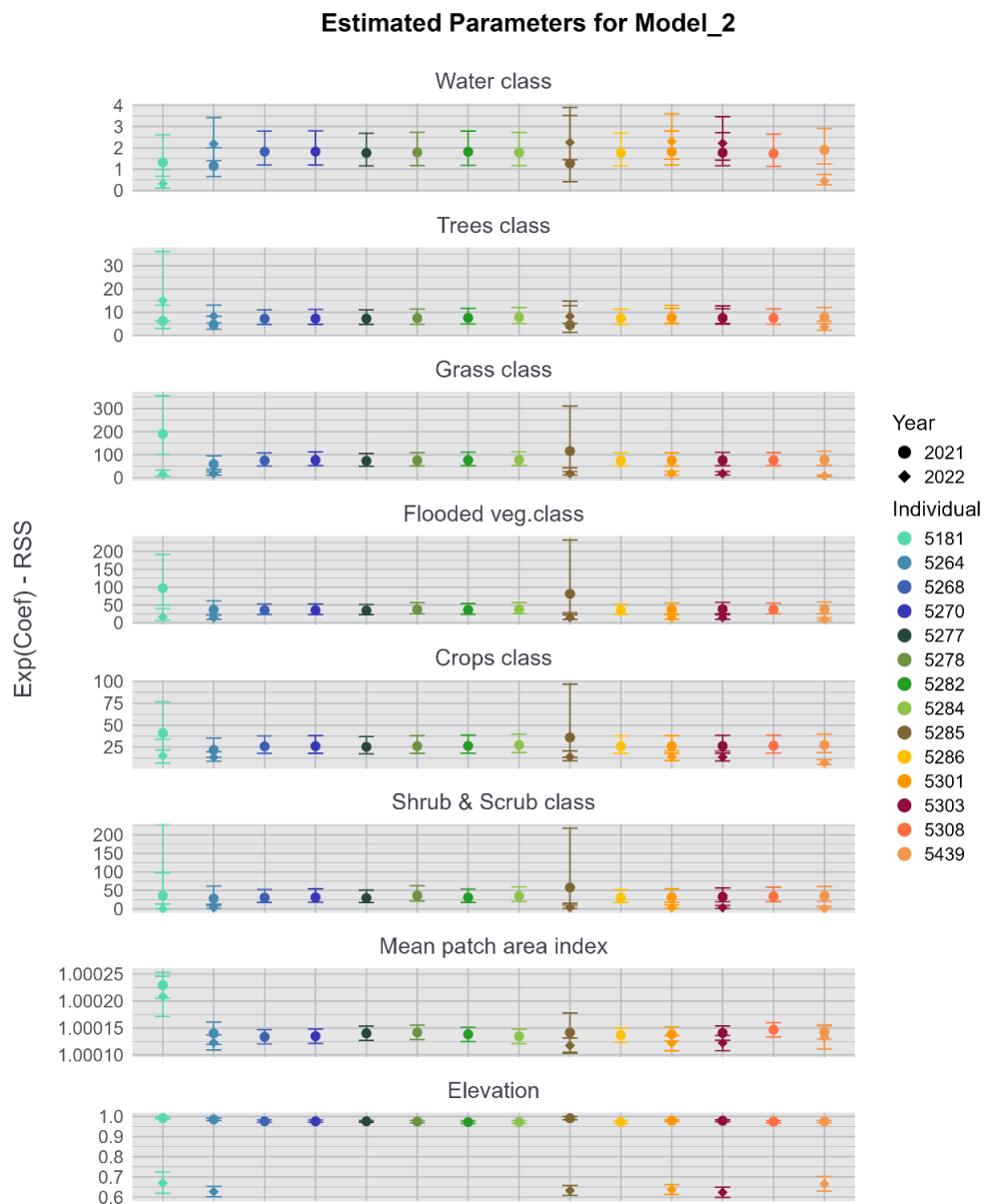


Figure 27. **Estimated parameters in fitted SSF model 2 (size of patches) per individual for the years 2021 and 2022.** The x-axis displays the ranges of relative selection strength (RSS) values for all variables considered in Model 2. Each color represents an individual, as indicated in the legend on the right. Data from 2021 is represented by circles, while data from 2022 is represented by diamond symbols.

For the mean patch area index and elevation variables, the RSS magnitudes in 2021 were 1. This suggests that these predictors had a weak effect on habitat selection by individuals, at least within the dataset used in this study. In contrast, in 2022, elevation had RSS values of 0.6 for the group

of individuals. This indicates a weak negative effect on habitat selection. That is to say, the group of northern lapwings showed a slight preference of use for locations with lower altitude.

For the overall predictor variables, the p-values were  $<0.001$ , pointing statistical significance in the model fit, with the exception of individual “5264” for the year 2021 in the LCLU class “Water” with a p-value = 0.636 and individual “5439” in the LCLU class “Shrub & Scrub” with a p-value = 0.908. The examination of the trends of the confidence intervals for the overall variables (see Figure 27.) showed them to be narrow, indicating minimal uncertainty in the estimation. However, exceptionally, the confidence intervals for individuals ‘5181’ and ‘5285’ in the LCLU classes are notably wider compared to the others, indicating greater uncertainty.

### 3.2.4 The effect of the aggregation of patches on the habitat use of the northern lapwing

For model 3 where the landscape metric considered is Contagion index, following the structure of the Equation 9:

Equation 9. Model 3 structure that includes the Contagion index “contagio” and the environmental variables: *land cover*, *elevation*, *log\_sl*, *cos\_ta* and *step stratification*.

$$\text{model 3} = \text{steps} \sim \text{landcover} + \text{contagio} + \text{elevation} + \text{log\_sl\_} + \text{cos\_ta\_} \\ + \text{strata}(\text{step\_id\_})$$

Where *steps* refers to the set of used and available steps, and *landcover* refers to LULC classes, and in alignment with prior results obtained in the sections 3.2.2 *The effect of landscape diversity on the habitat use of the northern lapwing* and 3.2.3 *The effect of the patch size on the habitat use of the northern lapwing* the RSS measurements for the LCLU classes were similar to that of model 1 and model 2. There is a stronger strength in the relative selection of areas with *Grass* over the other classes (see Figure 28 **Error! Reference source not found.**). The same occurs with the elevation variable, which remains at values close to 1, reiterating that at least for this data set, it could not have an influence on relative habitat preferences (see supporting information in 8.1 *Appendix. Standard outputs of SSF fitted models.*).

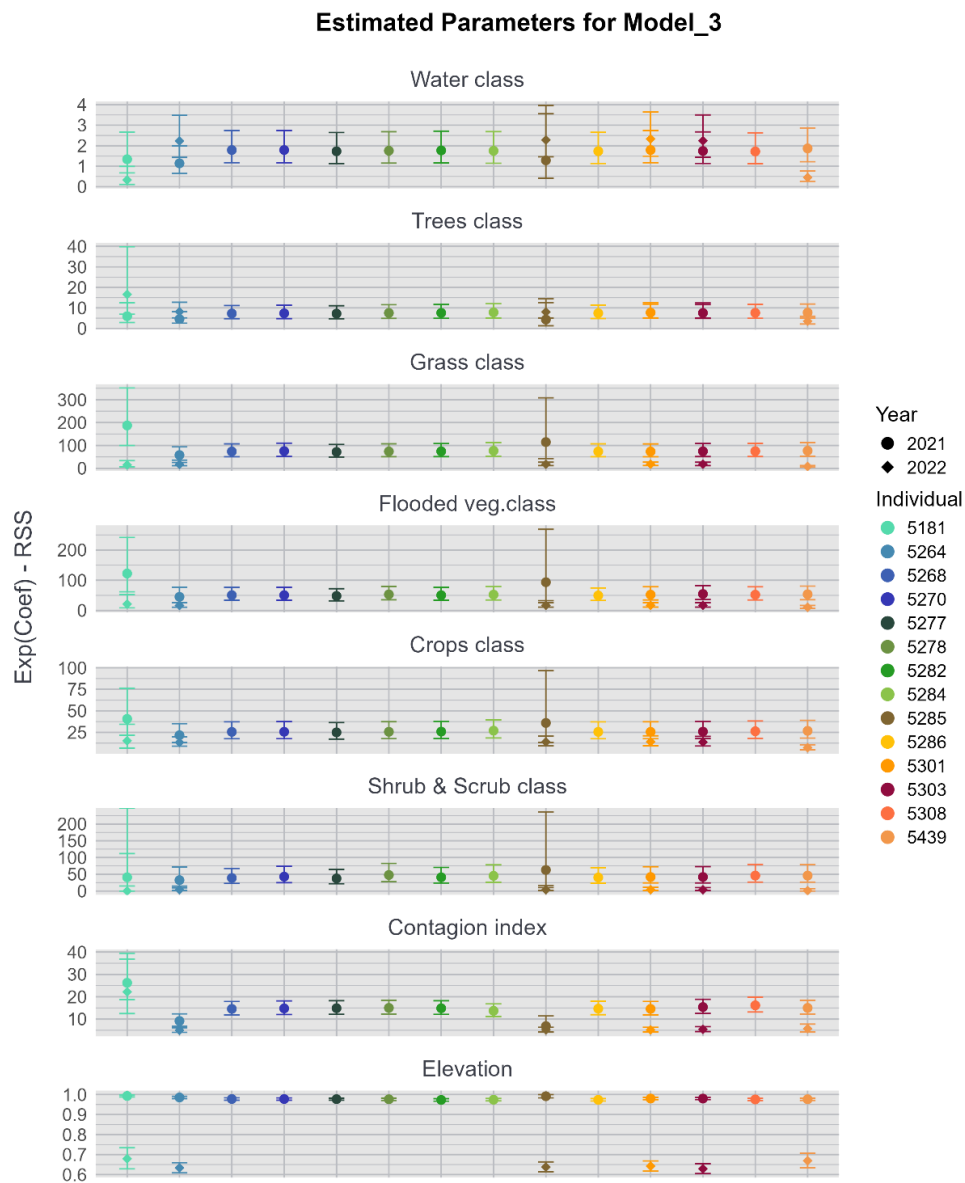


Figure 28. **Estimated parameters in fitted SSF model 3 (aggregation of patches) per individual for the years 2021 and 2022.** The x-axis displays the ranges of relative selection strength (RSS) values for all variables considered in Model 3. Each color represents an individual, as indicated in the legend on the right. Data from 2021 is represented by circles, while data from 2022 is represented by diamond symbols

The contagion index variable showed RSS measures ranging from 5 to 26.3 across the 20 fitted models for 2021 and 2022. These values indicate that the individuals studied exhibited a stronger relative selection for areas where the contagion index was higher. In other words, they showed a greater intensity of use in landscapes with a higher degree of patch clustering, where patches of

similar cover types are more spatially aggregated. This pattern suggests that individuals displayed a relative preference for habitats that provide continuous or cohesive resources, particularly when a specific cover class dominates a large portion of the available habitat.

For both Model 3 discussed in this section and the models examined in sections 3.2.2 *The effect of landscape diversity on the habitat use of the northern lapwing* and 3.2.3 *The effect of the patch size on the habitat use of the northern lapwing*, the confidence intervals remain narrow for most individuals studied. This applies to the variables LULC classes (*land cover*), Shannon diversity index (*shannon*), mean patch area index (*mean\_patch\_area*), and contagion index (*contagio*), indicating reliable estimates. However, individuals identify as “5181” and “5285” present wider confidence intervals, indicating higher uncertainty in the estimated coefficients.

### 3.3 Model performance

#### 3.3.1 Akaike’s Information Criterion for model selection

The AIC value was calculated for each of the three adjusted models to identify the one that best fits the observations, minimizing the loss of information between the data distribution and the model assumptions (Burnham et al., 2011). Based on this criterion, the best model is the one with the lowest AIC. As shown in Table 11, for all individuals across 2021 and 2022, Model 1 that incorporate the Shannon diversity index, held the model with the lowest AIC.

Table 11. Compiled AIC scores for models by individual for the years 2021–2022. Model 1 included the Shannon diversity index, Model 2 incorporated the mean patch area index, and Model 3 used the contagion index.

AKAIKE INFORMATION CRITERION (AIC) EVALUATION						
Id - individual	Data: 2021			Data: 2022		
	<i>Model 1</i>	<i>Model 2</i>	<i>Model 3</i>	<i>Model 1</i>	<i>Model 2</i>	<i>Model 3</i>
<b>5181 – Dokkum</b>	14630.99	14951.36	14927.81	9972.884	10028.33	10029.08
<b>5264 – Joris</b>	22303.52	22617.81	22566.07	70792.71	70911.32	70958.12
<b>5268 – Kollumerpomp</b>	68469.10	68882.70	68533.44	--	--	--
<b>5270 – Kollumeroudzjil</b>	68392.42	68822.13	68475.17	--	--	--
<b>5277 – Buitenpost</b>	68480.70	68870.99	68548.24	--	--	--
<b>5278 – Miedum</b>	68306.60	68686.42	68367.58	--	--	--
<b>5282 – Lekkum d1</b>	68412.14	68834.59	68503.51	--	--	--
<b>5284 – Kollum</b>	68355.30	68750.78	68438.79	--	--	--

<b>5285 – Lekku d2</b>	6305.52	6421.45	6422.93	70858.14	70988.94	70997.15
<b>5286 – Augsburg</b>	68424.78	68831.99	68492.17	--	--	--
<b>5301 – Driezum</b>	68562.15	68981.88	68662.83	70818.99	70926.30	70957.81
<b>5303 – Jorn</b>	68452.76	68844.12	68503.10	70605.68	70745.34	70759.53
<b>5308 – Aagje</b>	68355.61	68789.73	68460.34	--	--	--
<b>5439 – Wyns</b>	68533.00	68929.95	68614.61	28492.10	28565.82	28584.95

Lower AIC value	Intermediate AIC Value	Highest AIC Value
-----------------	------------------------	-------------------

Subsequently, model 2 (mean patch area index model) and model 3 (contagion index) exhibit high and intermediate AIC values. This pattern suggests that, despite their similar structure, these models may offer a less robust statistical explanation for habitat selection preferences.

### 3.3.2 K-Fold cross validation

A k-fold cross-validation was computed to evaluate how well the fitted models generalize to unseen data. Specifically, the area under the curve (AUC) metric was used to quantify the model's ability to predict the relative probability of individuals selecting a specific habitat (Boyce et al., 2002). Overall, most models fitted with 2021 data had a mean AUC > 0.7 (see Table 12), suggesting a good predictive capacity. This indicates that model performance remains relatively stable across different landscape metrics, with minimal variation in fit quality. In other words, the models consistently capture habitat selection patterns, reinforcing their reliability for inference.

Table 12. Results of the 10-Fold Cross-Validation using the AUC metric to evaluate model performance by individuals for the years 2021–2022. Model 1 included the Shannon diversity index, Model 2 incorporated the mean patch area index, and Model 3 used the contagion index.

Cross validated models, K-Folds = 10 Metric: AUC						
Id - individual	Data: 2021			Data: 2022		
	<i>Model 1</i>	<i>Model 2</i>	<i>Model 3</i>	<i>Model 1</i>	<i>Model 2</i>	<i>Model 3</i>
5181 – Dokkum	0.914	0.914	0.914	0.758	0.758	0.758
5264 – Joris	0.783	0.783	0.783	0.686	0.686	0.686
5268 – Kollumerpomp	0.767	0.767	0.767	--	--	--
5270 – Kollumeroudzjil	0.768	0.768	0.768	--	--	--
5277 – Buitenpost	0.768	0.768	0.768	--	--	--
5278 – Miedum	0.769	0.769	0.769	--	--	--

5282 – Lekkum d1	0.768	0.768	0.768	--	--	--
5284 – Kollum	0.768	0.768	0.768	--	--	--
5285 – Lekkum d2	0.843	0.843	0.843	0.686	0.686	0.686
5286 – Augsburg	0.767	0.767	0.767	--	--	--
5301 – Driezum	0.767	0.767	0.767	0.686	0.686	0.686
5303 – Jorn	0.767	0.767	0.767	0.689	0.689	0.689
5308 – Aagje	0.768	0.768	0.768	--	--	--
5439 – Wyns	0.767	0.767	0.767	0.681	0.681	0.681

In contrast to others, the individuals ‘5181’ and ‘5285’ exhibit the highest AUC values ( $> 0.84$ , see Table 12). The validation of models fitted with data from 2022 generally showed low performance, with AUC values around 0.6. This indicates that their predictive capacity requires improvement, which may be partially explained by an insufficient number of observations.

## 4 Discussion

The main goal of this thesis was to determine how landscape complexity influences the habitat selection of the northern lapwing. The calculation of local landscape metrics Shannon diversity index, mean patch area index, and contagion index allowed for addressing the first objective by quantifying landscape complexity in terms of resource diversity and spatial configuration. At the local scale used for these calculations, the study area, defined as the region utilized by northern lapwings during the breeding season was found to be predominantly homogeneous. These landscape metrics served as explanatory variables in the step-selection function (SSF) models, contributing to the analysis of habitat selection.

The application of SSF models made it possible to address the second specific objective: modeling habitat selection in the studied individuals. The estimates derived from these models indicate that habitat selection was stronger in areas where landscape metrics reflected lower complexity. These findings indicate that the likelihood of habitat selection increases in less complex landscapes among the studied group. This indicates that northern lapwings may actively seek out environments where resource distribution is more uniform and predictable, reducing the challenges associated with navigating highly heterogeneous landscapes. Homogeneous areas likely provide clearer visibility, facilitating predator detection, as well as easier access to food resources, which aligns with the species' preference for open habitats with short vegetation.

Moreover, the preference for less complex landscapes may reflect an adaptive strategy linked to their breeding ecology. During the reproductive season, selecting habitats with lower structural diversity could enhance nesting success by minimizing risks associated with habitat fragmentation and predation. In highly diverse landscapes with mixed land cover, contrasting vegetation types may create more complex microhabitats that attract a wider range of predators or increase competition for resources. This could explain why the study group exhibited a stronger preference for landscapes with lower complexity.

Ultimately, these findings answer the central research question by demonstrating that landscape complexity plays a significant role in shaping habitat selection during the breeding season. The observed preference for less complex landscapes suggests that conservation efforts should focus on protecting and restoring large, homogeneous areas with suitable nesting conditions rather than promoting landscape heterogeneity in breeding grounds. An important next step would be to assess how different levels of complexity influence reproductive success and survival rates, refining conservation strategies accordingly.

Although no published studies were found that specifically examine the effect of landscape complexity on the habitat selection process of this species for direct comparison, this thesis provides novel insights based on the findings obtained. However, research focused on other aspects, such as northern lapwing preferences regarding land use and land cover, suggests a preference for large grasslands and wetlands. This aligns with the characterization of homogeneous landscapes, reinforcing the hypothesis that lower landscape complexity is favorable for this species.

As highlighted in the subsection *State of the art for the analysis of habitat selection*, northern lapwings prefer open areas with short vegetation, which enhances their ability to detect predators while tending to their nests and accessing soil-based food resources (Schekkerman et al., 2009). Bertholdt et al. (2017) further observed that when habitat selection considered mixed land cover such as grasslands or agricultural fields adjacent to forests northern lapwings tended to avoid areas where these two covers are near. A proposed conservation measure was the restoration of wet grasslands at distances greater than 500 meters from forested areas. These studies highlight specific habitat preferences but do not report lapwing nests in highly diverse landscapes composed of a mix of different land use and land cover classes.

These findings suggest that both landscape diversity and spatial resource distribution influence habitat selection in northern lapwings, guiding their preference for or avoidance of specific areas. This provides valuable insights into which habitats should be prioritized for protection or restoration to ensure safe breeding grounds, shelter, and food availability. Similar to agri-environmental schemes where delaying crop mowing has been proposed to enhance reproductive success (Bertholdt et al., 2017), conservation efforts could be further refined by targeting protection measures in specific areas where nesting is most likely to occur.

For instance, the estimates derived from SSF models can be used to generate probability maps, serving as a strategic guide for defining priority areas where protective measures such as nest safeguarding and predator deterrence should be implemented. Identifying nests in the field presents a significant challenge, particularly when attempting to identify nests in the field is particularly challenging when assessing nesting sites across northern lapwing populations in Europe. A probability-based habitat selection map would optimize fieldwork efforts, enabling researchers to focus on high-probability nesting areas more efficiently.

Additionally, these targeted conservation strategies could enhance collaboration with farmers by facilitating the adoption of conservation measures. Providing farmers with precise nesting locations would allow delayed mowing strategies to be applied selectively rather than uniformly across entire fields, minimizing economic disruptions while still promoting conservation efforts.

This thesis also emphasizes that the scale at which landscape metrics were calculated using moving windows may influence the results. A more detailed review of this aspect in spatial analysis would provide a stronger basis for selecting the most appropriate scale to represent landscape reality. Similarly, sampling frequency may impact parameter estimates, as discussed in section 3.2 *Estimation parameters of step-selection functions*. Individuals with higher sampling rates exhibited slightly different behaviors compared to the rest of the group. While landscape metrics retained their directional effect, differences in effect size and confidence intervals were observed. Therefore, ensuring frequent and uniform sampling rates for the studied individuals is crucial for robust analyses.

An alternative approach for SSF model fitting involves exploring the use of different landscape metrics that capture the two key components addressed in this study: composition and spatial configuration. Considering alternative metrics, such as those discussed in *FRAGSTATS: Spatial pattern analysis program for categorical maps* (McGarigal et al., 2002), could provide deeper insights into how different aspects of landscape complexity influence habitat selection. Evaluating whether variations in explanatory variables lead to differences in the observed effects of landscape complexity would contribute to a more comprehensive understanding of how spatial patterns and resource distribution shape species movement and site preference. This approach offers a valuable opportunity to enhance habitat selection models and improve their ecological relevance.

## 5 Prospects for future study

### 5.1 Movement simulation approach

The estimated parameters from the SSF model fitting enable simulations to predict potential movement trajectories of northern lapwings during the breeding season, using real recorded positions as a reference. One possible implementation is the approach proposed in “*Estimating utilization distributions from fitted step-selection functions*” by Signer et al., (2017), which provides a set of R functions for simulating space use and includes a complete case study as an example. These simulations help identify movement and habitat selection patterns, enhancing the understanding of northern lapwing behavior. Modeling habitat selection and movement patterns enhances conservation planning by identifying critical areas for protection and management.

### 5.2 Multitemporal analysis

According to report “*IUCN Red List of Threatened Species: Vanellus vanellus, Northern Lapwing*” by BirdLife International (2021), the intensive land use intensification is one of the main reasons for population decline of this species. The effects of land use changes on individual behavior could be analyzed through a multitemporal assessment of land use and land cover changes in recent years, alongside the evaluation of habitat selection during the same period. This approach would help establish patterns and relationships between the two events.

## 6 Conclusions

This thesis aimed to provide insights into the first key question of the subsection *1.3 Objectives and Key Questions*, how landscape characteristics, such as diversity and configuration, influence habitat selection in northern lapwings during the breeding season, when they are most vulnerable. Based on tracking data from 14 northern lapwing deployments, the study area was defined (see *2.3.2 Defining study area*), and performing the calculation of key landscape metrics Shannon diversity index, mean patch area and contagion, applying a local scale approach to capture in more detail the interaction of individuals with the environment.

The calculation of landscape metrics within a  $70 \times 70$  meter extent around each pixel of the input LULC dataset “*Dynamic World*” revealed that the study area is characterized by low diversity of LULC classes. The landscape is dominated by one or a few classes, which are highly aggregated into large patches and continuously connected, characterizing a mainly homogeneous area with a simple spatial configuration.

The second key question of the subsection *1.3 Objectives and Key Questions* is answer by the results of the fitted SSF models, which incorporated landscape metrics, elevation, and LULC classes, indicate that small RSS magnitudes in response to the Shannon diversity index reflect a relatively negative effect on habitat selection by the studied individuals. This suggests that, under the conditions of this study and considering the effect of the other environmental variables included in the model, individuals had a slightly higher selection strength for areas with lower LULC diversity, compared to other available places.

At the study scale considered, RSS values for the mean patch area index between 0.6 and 1, and for elevation at 1, indicate a weak or null effect on the relative selection of a location by the individuals. In other words, the average patch size of LULC classes and elevation values do not have stronger influence in the habitat selection intensity. In contrast, RSS magnitudes ranging from 5 to 26.3 for the contagion index indicate a positive change in the probability of habitat selection. This suggests that, an increase in the contagion index is associated with increased habitat selection by the individuals.

The RSS magnitudes obtained for overall landscape metrics, as discussed in previous paragraphs, suggest a positive effect in the relative probability of northern lapwings selecting homogeneous landscapes. These landscapes are characterized by low class diversity and a simple spatial distribution. This pattern may also be linked to habitat availability in the study area, which predominantly consists of homogeneous landscapes.

Although the environmental variables LULC and elevation may influence the RSS magnitudes of the landscape metrics Shannon diversity index, mean patch area, and contagion index, fitting separate models for each metric while maintaining a consistent variable structure allows for an isolated assessment of their effects. This approach ensures that the influence of each landscape feature on northern lapwing habitat selection can be interpreted independently, providing a clearer understanding of how these three spatial characteristics shape selection patterns.

This master's thesis advocates for the use of moving windows to integrate more detailed spatial analyses into habitat selection modeling. Moving windows were proposed as a method that allows for multi-scale analyses, such as the calculation of landscape metrics. By enabling a more dynamic representation of spatial patterns, these techniques contribute to recreating realistic scenarios in resource selection modeling. This approach highlights an opportunity to foster stronger synergies between geoinformatics and ecology.

The methodology developed for this study is designed to be both applicable and reproducible across different trackable species and study areas. As long as animal movement data can be collected, habitat selection can be analyzed for a wide range of species. The selection of explanatory variables should be guided by the study's hypotheses and objectives to ensure relevance to the research question. Furthermore, this methodology can be automated in environments that integrate multiple programming languages and spatial analysis tools, enhancing its scalability and adaptability for future applications.

## 7 References

- Akinwande, M. O., Dikko, H. G., & Samson, A. (2015). Variance Inflation Factor: As a Condition for the Inclusion of Suppressor Variable(s) in Regression Analysis. *Open Journal of Statistics*, *05*(07), 754–767. <https://doi.org/10.4236/ojs.2015.57075>
- Amani, M., Ghorbanian, A., Ahmadi, S. A., Kakooei, M., Moghimi, A., Mirmazloumi, S. M., Moghaddam, S. H. A., Mahdavi, S., Ghahremanloo, M., Parsian, S., Wu, Q., & Brisco, B. (2020). Google Earth Engine Cloud Computing Platform for Remote Sensing Big Data Applications: A Comprehensive Review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *13*, 5326–5350. <https://doi.org/10.1109/JSTARS.2020.3021052>
- Avgar, T., Lele, S. R., Keim, J. L., & Boyce, M. S. (2017). Relative Selection Strength: Quantifying effect size in habitat- and step-selection inference. *Ecology and Evolution*, *7*(14), 5322–5330. <https://doi.org/10.1002/ece3.3122>
- Baddeley, A., Rubak, E., & Turner, R. (2015). *Spatial Point Patterns: Methodology and Applications with R*. Chapman and Hall/CRC Press. <http://www.crcpress.com/Spatial-Point-Patterns-Methodology-and-Applications-with-R/Baddeley-Rubak-Turner/9781482210200/>
- Barba-Escoto, L., Howison, R. A., Fokkema, R. W., Duriaux-Chavarriá, J.-Y., Stessens, M., Van Der Velde, E., Hooijmeijer, J. C. E. W., Piersma, T., & Tittone, P. A. (2024). Are they even there? How agri-environment schemes investments reach their target species in Dutch dairy-farmland, the case of meadow birds. *Global Ecology and Conservation*, *56*, e03286. <https://doi.org/10.1016/j.gecco.2024.e03286>
- Berg, A., Lindberg, T., & Kallebrink, K. G. (1992). Hatching Success of Lapwings on Farmland: Differences between Habitats and Colonies of Different Sizes. *The Journal of Animal Ecology*, *61*(2), 469. <https://doi.org/10.2307/5337>
- Bertholdt, N. P., Gill, J. A., Laidlaw, R. A., & Smart, J. (2017). Landscape effects on nest site selection and nest success of Northern Lapwing *Vanellus vanellus* in lowland wet grasslands. *Bird Study*, *64*(1), 30–36. <https://doi.org/10.1080/00063657.2016.1262816>

- BirdLife International. (2016). *Species factsheet: Northern Lapwing Vanellus vanellus*. BirdLife DataZone. <https://datazone.birdlife.org/species/factsheet/northern-lapwing-vanellus-vanellus>
- BirdLife International. (2021). *IUCN Red List of Threatened Species: Vanellus vanellus, Northern Lapwing* (E.T22693949A166266204). International Union for Conservation of Nature. <https://doi.org/10.2305/iucn.uk.2021-3.rlts.t22693949a166266204.en>
- Boyce, M. S., Vernier, P. R., Nielsen, S. E., & Schmiegelow, F. K. A. (2002). Evaluating resource selection functions. *Ecological Modelling*, 157(2–3), 281–300. [https://doi.org/10.1016/s0304-3800\(02\)00200-4](https://doi.org/10.1016/s0304-3800(02)00200-4)
- Breeuwer, A., Berendse, F., Willems, F., Foppen, R., Teunissen, W., Schekkerman, H., & Goedhart, P. (2009). Do meadow birds profit from agri-environment schemes in Dutch agricultural landscapes? *Biological Conservation*, 142(12), 2949–2953. <https://doi.org/10.1016/j.biocon.2009.07.020>
- Brown, C. F., Brumby, S. P., Guzder-Williams, B., Birch, T., Hyde, S. B., Mazzariello, J., Czerwinski, W., Pasquarella, V. J., Haertel, R., Ilyushchenko, S., Schwehr, K., Weisse, M., Stolle, F., Hanson, C., Guinan, O., Moore, R., & Tait, A. M. (2022). Dynamic World, Near real-time global 10 m land use land cover mapping. *Scientific Data*, 9(1). <https://doi.org/10.1038/s41597-022-01307-4>
- Buderman, F. E., Helm, P. J., Clark, J. D., Williamson, R. H., Yarkovich, J., & Mullinax, J. M. (2023). A multi-level modeling approach to guide management of female feral hogs in Great Smoky Mountains National Park. *Biological Invasions*, 25(10), 3065–3082. <https://doi.org/10.1007/s10530-023-03086-4>
- Burnham, K. P., Anderson, D. R., & Huyvaert, K. P. (2011). AIC model selection and multi-model inference in behavioral ecology: Some background, observations, and comparisons. *Behavioral Ecology and Sociobiology*, 65(1), 23–35. <https://doi.org/10.1007/s00265-010-1029-6>
- Craft, T. B., Piersma, T., Hooijmeijer, J. C. E. W., Zhu, B., D’Souza, M., O’Reilly, E., Fokkema, R. W., Stessens, M., Belting, H., Marlow, C., Ludwig, J., Melter, J., Alves, J. A., Esteban-Pineda, A., Gutiérrez, J. S., Masero, J. A., Rocha, A. D., Dreef, C., & Howison, R. A. (2025). Remote sensing and GPS tracking reveal temporal shifts in habitat

- use in nonbreeding Black-tailed Godwits. *Journal of Applied Ecology*, 62(1), 119–131.  
<https://doi.org/10.1111/1365-2664.14827>
- Derer, F. (n.d.). *Der Kiebitz ist der Vogel des Jahres 2024—NABU Thüringen*. NABU - Naturschutzbund Deutschland e.V. Retrieved February 6, 2025, from <https://thueringen.nabu.de/news/2023/34019.html>
- Eglington, S. M., Bolton, M., Smart, M. A., Sutherland, W. J., Watkinson, A. R., & Gill, J. A. (2010). Managing water levels on wet grasslands to improve foraging conditions for breeding northern lapwing *Vanellus vanellus*. *Journal of Applied Ecology*, 47(2), 451–458. <https://doi.org/10.1111/j.1365-2664.2010.01783.x>
- European Space Agency & Airbus. (2022). *Copernicus DEM* [Dataset].  
<https://doi.org/10.5270/ESA-c5d3d65>
- Fieberg, J. (2024). *Statistics for Ecologists: A Frequentist and Bayesian Treatment of Modern Regression Models*. University of Minnesota Libraries Publishing.  
<https://doi.org/10.24926/9781959870029>
- Fieberg, J., Signer, J., Smith, B., & Avgar, T. (2021). A 'How to' guide for interpreting parameters in habitat–selection analyses. *Journal of Animal Ecology*, 90(5), 1027–1043.  
<https://doi.org/10.1111/1365-2656.13441>
- Florko, K. R. N., Togunov, R. R., Gryba, R., Sidrow, E., Ferguson, S. H., Yurkowski, D. J., & Auger-Méthé, M. (2024). *A review of statistical models used to characterize species-habitat associations with animal movement data*. arXiv.  
<https://doi.org/10.48550/arXiv.2401.17389>
- GDAL/OGR contributors. (2025). *GDAL/OGR Geospatial Data Abstraction software Library*. Open Source Geospatial Foundation. <https://doi.org/10.5281/zenodo.5884351>
- Gillies, S. & others. (2013). *Rasterio: Geospatial raster I/O for Python programmers* [Computer software]. Mapbox. <https://github.com/rasterio/rasterio>
- Google Developers. (n.d.). *Dynamic World v1 dataset description*. Retrieved January 25, 2025, from [https://developers.google.com/earth-engine/datasets/catalog/GOOGLE\\_DYNAMICWORLD\\_V1#description](https://developers.google.com/earth-engine/datasets/catalog/GOOGLE_DYNAMICWORLD_V1#description)
- Google & the World Resources Institute. (n.d.). *Dynamic World—10m global land cover dataset in Google Earth Engine*. <https://dynamicworld.app/about/>

- Grolemund, G., & Wickham, H. (2011). Dates and Times Made Easy with lubridate. *Journal of Statistical Software*, 40(3), 1–25.
- Grundmann. (n.d.). *Naturschutzgebiet Düffel*. Retrieved February 6, 2025, from <https://www.nabu-naturschutzstation.de/schutzgebiete/naturschutzgebiet-dueffel/>
- Hadd, A., & Rodgers, J. L. (2021). *Understanding correlation matrices*. SAGE.
- Hagen-Zanker, A. (2016). A computational framework for generalized moving windows and its application to landscape pattern analysis. *International Journal of Applied Earth Observation and Geoinformation*, 44, 205–216. <https://doi.org/10.1016/j.jag.2015.09.010>
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., Fernández del Río, J., Wiebe, M., Peterson, P., ... Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585, 357–362. <https://doi.org/10.1038/s41586-020-2649-2>
- Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed). Springer.
- Hesselbarth, M. H. K., Sciaini, M., Nowosad, J., Hanss, S., Graham, L. J., Hollister, J., With, K. A., Privé, F., Nayuki, P., & Strimas-Mackey, M. (2024). *Landscape Metrics for Categorical Map Patterns: Package `landscapemetrics`* (2.1.4). <https://cran.r-project.org/web/packages/landscapemetrics/landscapemetrics.pdf>
- Hijmans, R. J. (2024). *raster: Geographic Data Analysis and Modeling*. <https://rspatial.org/raster>
- Hirzel, A. H., Le Lay, G., Helfer, V., Randin, C., & Guisan, A. (2006). Evaluating the ability of habitat suitability models to predict species presences. *Ecological Modelling*, 199(2), 142–152. <https://doi.org/10.1016/j.ecolmodel.2006.05.017>
- Hoo, Z. H., Candlish, J., & Teare, D. (2017). What is an ROC curve? *Emergency Medicine Journal*, 34(6), 357–359. <https://doi.org/10.1136/emered-2017-206735>
- Horvat, E., & Denac, D. (2019). Population dynamics and habitat use by Northern Lapwing *Vanellus vanellus* in agricultural landscape of Dravsko and Ptujsko polje (NE Slovenia). *Acrocephalus*, 40(182–183), 3–22. <https://doi.org/10.1515/acro-2019-0009>

- Hughes, A. W., & King, M. L. (2003). Model selection using AIC in the presence of one-sided information. *Journal of Statistical Planning and Inference*, 115(2), 397–411.  
[https://doi.org/10.1016/S0378-3758\(02\)00159-3](https://doi.org/10.1016/S0378-3758(02)00159-3)
- Jackson, D. B. (2001). Experimental removal of introduced hedgehogs improves wader nest success in the Western Isles, Scotland. *Journal of Applied Ecology*, 38(4), 802–812.  
<https://doi.org/10.1046/j.1365-2664.2001.00632.x>
- Johansson, O. C., & Blomqvist, D. (1996). Habitat Selection and Diet of Lapwing *Vanellus vanellus* Chicks on Coastal Farmland in S. W. Sweden. *The Journal of Applied Ecology*, 33(5), 1030. <https://doi.org/10.2307/2404683>
- Jost, L., & González-Oreja, J. (2012). Midiendo la diversidad biológica: Más allá del índice de Shannon. *Acta Zoologica Lilloana*, 56, 3–14.
- Juárez, M. G. (2022, February 22). Matriz de correlación [ProbabilidadyEstadistica.net]. *ProbabilidadyEstadistica*. <https://www.probabilidadyestadistica.net/matriz-de-correlacion/>
- Karimi, J. D., Corstanje, R., & Harris, J. A. (2021). Understanding the importance of landscape configuration on ecosystem service bundles at a high resolution in urban landscapes in the UK. *Landscape Ecology*, 36(7), 2007–2024. <https://doi.org/10.1007/s10980-021-01200-2>
- Korner, P., Hohl, S., & Horch, P. (2024). Brood protection is essential but not sufficient for population survival of lapwings *Vanellus vanellus* in central Switzerland. *Wildlife Biology*, 2024(4). <https://doi.org/10.1002/wlb3.01175>
- Lislevand, T., Byrkjedal, I., & Grønstøl, G. (2009). Dispersal and age at first breeding in Norwegian Northern Lapwings (*Vanellus vanellus*). *Ornis Fennica*, 86, 11–17.
- Machado, R., Godinho, S., Pirnat, J., Neves, N., & Santos, P. (2018). Assessment of landscape composition and configuration via spatial metrics combination: Conceptual framework proposal and method improvement. *Landscape Research*, 43(5), 652–664.  
<https://doi.org/10.1080/01426397.2017.1336757>
- McGarigal, K. S., Cushman, S., Neel, M., & Ene, E. (2002). *FRAGSTATS: Spatial pattern analysis program for categorical maps*.

- Michelot, T., Klappstein, N. J., Potts, J. R., & Fieberg, J. (2023). *Understanding step selection analysis through numerical integration* (No. arXiv:2308.15678). arXiv.  
<https://doi.org/10.48550/arXiv.2308.15678>
- Midha, N., & Mathur, P. K. (2010). Assessment of forest fragmentation in the conservation priority Dudhwa landscape, India using FRAGSTATS computed class level metrics. *Journal of the Indian Society of Remote Sensing*, 38(3), 487–500.  
<https://doi.org/10.1007/s12524-010-0034-6>
- Moreno, C. E., Barragán, F., Pineda, E., & Pavón, N. P. (2011). Reanálisis de la diversidad alfa: Alternativas para interpretar y comparar información sobre comunidades ecológicas. *Revista Mexicana de Biodiversidad*, 82(4).  
<https://doi.org/10.22201/ib.20078706e.2011.4.745>
- Morera, C., Pintó, J., & Romero, M. (2008). Procesos de Fragmentación Corredores Biológicos: Una introducción. *Journal of Latin American Geography*, 7(2), 163–166.
- Morrissey, M. B., & Ruxton, G. D. (2018). Multiple Regression Is Not Multiple Regressions: The Meaning of Multiple Regression and the Non-Problem of Collinearity. *Philosophy, Theory, and Practice in Biology*, 10(20220112).  
<https://doi.org/10.3998/ptpbio.16039257.0010.003>
- Müller, K., & Wickham, H. (2024). *tibble: Simple Data Frames*. <https://tibble.tidyverse.org/>
- NABU. (n.d.). *Willkommen beim NABU Kellinghusen*. nabu-kellinghusens Webseite! Retrieved February 6, 2025, from <http://www.nabu-kellinghusen.de/>
- NobleProg. (n.d.). *Advanced ArcGIS Pro for Spatial Analysis Training Course*. Retrieved January 30, 2025, from <https://www.nobleprog.de/en/cc/arcgisprospatial>
- Parish, D. M. B., & Coulson, J. C. (1998). Parental investment, reproductive success and polygyny in the lapwing, *Vanellus vanellus*. *Animal Behaviour*, 56(5), 1161–1167.  
<https://doi.org/10.1006/anbe.1998.0856>
- Pebesma, E. (2018). Simple Features for R: Standardized Support for Spatial Vector Data. *The R Journal*, 10(1), 439. <https://doi.org/10.32614/RJ-2018-009>
- Pebesma, E., & Bivand, R. (2005). *sp: Classes and Methods for Spatial Data* (p. 2.1-4) [Dataset]. <https://doi.org/10.32614/CRAN.package.sp>

- Peet, R. K. (1975). Relative Diversity Indices. *Ecology*, 56(2), 496–498.  
<https://doi.org/10.2307/1934984>
- R Core Team. (2024). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. <https://www.R-project.org/>
- Riitters, K. H., O’Neill, R. V., Wickham, J. D., & Jones, K. B. (1996). A note on contagion indices for landscape analysis. *Landscape Ecology*, 11(4), 197–202.  
<https://doi.org/10.1007/BF02071810>
- Robinson, C., & Schumacker, R. E. (2009). Interaction effects: Centering, variance inflation factor, and interpretation issues. *Multiple Linear Regression Viewpoints*, 35(1), 6–11.
- Royal Society for the Protection of Birds (RSPB). (2017). *Land Management for Wildlife: Lapwing (Vanellus vanellus)*. Royal Society for the Protection of Birds (RSPB).  
<https://www.rspb.org.uk>
- Schekkerman, H., Teunissen, W., & Oosterveld, E. (2009). Mortality of Black-tailed Godwit *Limosa limosa* and Northern Lapwing *Vanellus vanellus* chicks in wet grasslands: Influence of predation and agriculture. *Journal of Ornithology*, 150(1), 133–145.  
<https://doi.org/10.1007/s10336-008-0328-4>
- Sheldon, R. D., Kamp, J., Koshkin, M. A., Urazaliev, R. S., Iskakov, T. K., Field, R. H., Salemgareev, A. R., Khrokov, V. V., Zhuly, V. A., Sklyarenko, S. L., & Donald, P. F. (2013). Breeding ecology of the globally threatened Sociable Lapwing *Vanellus gregarius* and the demographic drivers of recent declines. *Journal of Ornithology*, 154(2), 501–516.  
<https://doi.org/10.1007/s10336-012-0921-4>
- Signer, J. (2024, January 4). *Getting started with amt*. [https://cran.r-project.org/web/packages/amt/vignettes/p1\\_getting\\_started.html](https://cran.r-project.org/web/packages/amt/vignettes/p1_getting_started.html)
- Signer, J., Fieberg, J., & Avgar, T. (2017). Estimating utilization distributions from fitted step-selection functions. *Ecosphere*, 8(4), e01771. <https://doi.org/10.1002/ecs2.1771>
- Signer, J., Smith, B., Reineking, B., Schlaegel, U., Fieberg, J., O’Brien, J., Niebuhr, B., Robitaille, A., Tal, A., & LaPoint, S. (2024). *Animal Movement Tools: Package ‘amt’ (0.2.2.0)*. <https://cran.r-project.org/web/packages/amt/amt.pdf>

- Singh, V. (2024, November 18). *Factor de Inflación de la Varianza: Cómo detectar la multicolinealidad*. [https://www.datacamp.com/tutorial/variance-inflation-factor?utm\\_source=chatgpt.com](https://www.datacamp.com/tutorial/variance-inflation-factor?utm_source=chatgpt.com)
- Spatial Data Science Web Site. (n.d.). *The terra package—R Spatial*. <https://rspatial.org/pkg/1-introduction.html>
- Spellerberg, I. F., & Fedor, P. J. (2003). A tribute to Claude Shannon (1916–2001) and a plea for more rigorous use of species richness, species diversity and the ‘Shannon–Wiener’ Index. *Global Ecology and Biogeography*, 12(3), 177–179. <https://doi.org/10.1046/j.1466-822X.2003.00015.x>
- Subirós, J., Linde, D., Pascual, A., & Ribas Palom, A. (2006). Conceptos y métodos fundamentales en ecología del paisaje (landscape ecology). Una interpretación desde la geografía. *Documents d’Anàlisi Geogràfica; Núm.: 48, 48*.
- Swiss Ornithological Institute. (2023). *Vogelarten: Northern lapwing*. <https://www.vogelwarte.ch/en/birds-of-switzerland/northern-lapwing/>
- Tamiminia, H., Salehi, B., Mahdianpari, M., Quackenbush, L., Adeli, S., & Brisco, B. (2020). Google Earth Engine for geo-big data applications: A meta-analysis and systematic review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 164, 152–170. <https://doi.org/10.1016/j.isprsjprs.2020.04.001>
- Therneau, T. M. (2020). *A Package for Survival Analysis in R*. <https://CRAN.R-project.org/package=survival>
- Thurfjell, H., Ciuti, S., & Boyce, M. S. (2014). Applications of step-selection functions in ecology and conservation. *Movement Ecology*, 2(1), 4. <https://doi.org/10.1186/2051-3933-2-4>.
- United States Department of Agriculture. (2019). *NPI Rescaler User Guide*.
- van den Berg, R. (2024). *The breeding ecology of the northern lapwing (Vanellus vanellus) in France: Investigating the decline of a widely-distributed wader* (Issue 2024STRAJ017) [Theses, Université de Strasbourg]. <https://theses.hal.science/tel-04690418>
- Van Rossum, G. & Python development team. (2018, February 9). *Python Tutorial Release 3.7.0*. Python Software Foundation.

- Wagenmakers, E.-J., & Farrell, S. (2004). AIC model selection using Akaike weights. *Psychonomic Bulletin & Review*, *11*, 192–196.
- Watson, M., Wilson, J. M., Koshkin, M., Sherbakov, B., Karpov, F., Gavrilov, A., Schielzeth, H., Brombacher, M., Collar, N. J., & Cresswell, W. (2006). Nest survival and productivity of the critically endangered Sociable Lapwing *Vanellus gregarius*. *Ibis*, *148*(3), 489–502. <https://doi.org/10.1111/j.1474-919X.2006.00555.x>
- Wickham, H., Chang, W., Lionel, H., Thomas, L. P., Kohske, T., Claus, W., Kara, W., Hiroaki, Y., Dewey, D., & Van den Brand, T. (n.d.). ggplot2: Introduction to ggplot2. *Ggplot2 3.5.1*. <https://ggplot2.tidyverse.org/articles/ggplot2.html>
- Wickham, H., François, R., Henry, L., & Müller, K. (2022). *dplyr: A Grammar of Data Manipulation*.
- Yates, L., Aandahl, Z., Richards, S. A., & Brook, B. W. (2022). *Cross validation for model selection: A primer with examples from ecology* (No. arXiv:2203.04552). arXiv. <https://doi.org/10.48550/arXiv.2203.04552>
- Zeller, K. A., McGarigal, K., Cushman, S. A., Beier, P., Vickers, T. W., & Boyce, W. M. (2016). Using step and path selection functions for estimating resistance to movement: Pumas as a case study. *Landscape Ecology*, *31*(6), 1319–1335. <https://doi.org/10.1007/s10980-015-0301-6>

## 8 Appendix

### 8.1 Appendix. Google Earth Engine code developed in JavaScript, for generating, visualizing, and exporting Dynamic World 2021 data across the interest area.

```
/* *****  
 * Define the area of interest.  
 * *****  
 * Creation of a polygon using a list of geographic coordinates  
 * (longitude, latitude). This polygon represents the boundary  
 * of our area of interest, which in this case is the extent covering  
 * the positions recorded during the breeding season.  
 * *****/  
var europe = ee.Geometry.Polygon([  
  [  
    [6.107946, 54.7522],  
    [8.385332, 52.58993],  
    [-3.823165, 38.03111],  
    [-5.775337, 35.62247],  
    [-8.734014, 36.97163],  
    [-3.291528, 46.41424],  
    [-6.162365, 47.4557],  
    [-6.324172, 49.8537],  
    [-1.775206, 50.59337],  
    [-3.84628, 52.66445],  
    [-3.513924, 54.71713],  
    [6.107946, 54.7522]  
  ]  
]);  
  
/* *****  
 * Load the Dynamic World dataset  
 * *****  
 * Google's Dynamic World (V1) is a global land cover dataset  
 * derived from Sentinel-2 imagery. It provides near real-time  
 * data for 9 land use and land cover classes.  
 * *****/  
var dynamicWorld = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1');  
  
/* *****  
 * Filter images for the year 2021  
 * *****  
 * 1. Restrict the dataset to images between Jan 1, 2021,  
 * and Dec 31, 2021. To download the dataset for 2022 y 2023  
 * just replace the year in the whole code.  
 * 2. Limit the area to interest area polygon.  
 * 3. Compute all filtered images to
```

```
* create a composite.
*****/
var dw2021 = dynamicWorld
  .filterDate('2021-01-01', '2021-12-31')
  .filterBounds(europe);
var dw2021_median = dw2021.median().clip(europe);

/*****
* Visualize the median composite
*****
* 1. Center the map on the area of interest.
* 2. Add the composite to the map, displaying the
*   'label' band, which indicates land cover classes.
* 3. Define a specific palette (color scheme)
*   and value range (0–8) for the classification.
*****/
Map.centerObject(europe, 4);
Map.addLayer(
  dw2021_median.select('label'),
  {
    min: 0,
    max: 8,
    palette: [
      '419BDF', // Water
      '397D49', // Trees
      '88B053', // Grass
      '7A87C6', // Flooded Vegetation
      'E49635', // Crops
      'DFC35A', // Shrub & Scrub
      'C4281B', // Built Area
      'A59B8F', // Bare Ground
      'B39FE1', // Snow & Ice
    ]
  },
  'Dynamic World - 2021'
);

/*****
* Export the processed image
*****
* Export the land cover 'label' band to Google Drive:
* - Using EPSG:4326 (WGS84) as the coordinate reference system.
* - Setting the output spatial resolution of 10 meters .
* - Limiting the export region to the polygon.
* - Allowing a high maximum number of pixels (maxPixels).
*****/
Export.image.toDrive({
  image: dw2021_median.select('label')
  .reproject({
    crs: 'EPSG:4326', // Coordinate reference system
    scale: 10 // Spatial resolution in meters
  }),
```

```
description: 'Dynamic_World_Europe_2021',  
folder: 'EarthEngine_Exports', //Change for name of the user folder  
fileNamePrefix: 'dynamic_world_europe_2021',  
region: europe,  
scale: 10, // Spatial resolution in meters  
crs: 'EPSG:4326', // Use WGS84  
maxPixels: 1e13 // Allow large exports  
});
```

```
/*  
* Create and add a legend to the map  
*  
* Build a small panel that lists each land cover class  
* alongside a color box. This helps to interpret  
* the map's classification.  
*/
```

```
var legend = ui.Panel({  
  style: {  
    position: 'bottom-left',  
    padding: '8px 15px'  
  }  
});
```

```
// Legend title  
var legendTitle = ui.Label({  
  value: 'Land Cover Classes (Dynamic World)',  
  style: {  
    fontWeight: 'bold',  
    fontSize: '14px',  
    margin: '0 0 4px 0',  
    padding: '0'  
  }  
});  
legend.add(legendTitle);
```

```
// Define color palette and class names  
var palette = [  
  '419BDF', '397D49', '88B053', '7A87C6', 'E49635',  
  'DFC35A', 'C4281B', 'A59B8F', 'B39FE1'  
];  
var names = [  
  'Water',  
  'Trees',  
  'Grass',  
  'Flooded Vegetation',  
  'Crops',  
  'Shrub & Scrub',  
  'Built Area',  
  'Bare Ground',  
  'Snow & Ice'  
];
```

```
// Loop through each class, creating legend entries
for (var i = 0; i < palette.length; i++) {
  // Color box
  var colorBox = ui.Label({
    style: {
      backgroundColor: '#' + palette[i],
      padding: '8px',
      margin: '0 0 4px 0'
    }
  });

  // Class name text
  var description = ui.Label({
    value: names[i],
    style: { margin: '0 0 4px 6px' }
  });

  // Combine color box and text into a horizontal panel
  var legendItem = ui.Panel({
    widgets: [colorBox, description],
    layout: ui.Panel.Layout.Flow('horizontal')
  });

  legend.add(legendItem);
}

// Add the legend to the map
Map.add(legend);
```

## 8.2 Appendix. Code developed in python to extract tracking data of breeding season.

```
#Import libraries to perform operations with attributes from shapefile.
import geopandas as gpd
import pandas as pd
# Load original shapefile of tracking data.
positions = "/content/drive/MyDrive/Lapwing_NFW_Vanellus_Vanellus/points_3035.shp"
gdf = gpd.read_file(positions)
# Name of the field containing date and time of capture of the individual's position in the original shapefile.
timestamp = 'timestamp'
# Convert the date and time field from 'text' to 'datetime' format.
gdf[timestamp] = pd.to_datetime(gdf[timestamp], format="%Y-%m-%d %H:%M:%S")
# Select data collected during the breeding season per year (2021, 2022 and 2023).
filtered_gdf = gdf[
  ((gdf[timestamp] >= '2021-03-15') & (gdf[timestamp] <= '2021-06-30')) |
  ((gdf[timestamp] >= '2022-03-15') & (gdf[timestamp] <= '2022-06-30')) |
  ((gdf[timestamp] >= '2023-03-15') & (gdf[timestamp] <= '2023-06-30'))
]
```

```
# Saving the filtered data in a new shapefile.  
filtered_gdf.to_file("/content/drive/MyDrive/Lapwing_NFW_Vanellus_Vanellus/Breeding_poi  
nts.shp")
```

### 8.3 Appendix. JavaScript code for Shannon diversity index calculation in environment GEE.

```
// Define the geometry (in geographic coordinates) base in the polygon define to the study area  
where the analysis will be performed.  
var areaOfInterest = ee.Geometry.Polygon([  
  [-1.529085, 46.87], [-2.229717, 47.50492], [-1.938698, 48.14066], [-0.579187, 48.08423],  
  [0.59795, 49.12168], [0.689647, 49.70694], [1.380388, 50.91567], [0.914895, 51.13522],  
  [-0.090363, 51.5187], [-1.108916, 52.11399], [-1.037776, 52.96926], [-0.091826, 53.74214],  
  [1.45859, 53.31731], [0.812231, 52.78129], [1.288547, 52.41761], [2.802825, 52.7073],  
  [4.593831, 53.29824], [5.518836, 54.10992], [7.850975, 53.67639], [6.264834, 51.61802],  
  [3.048556, 50.33066], [2.750967, 48.52854], [0.666143, 46.87172], [-1.529085, 46.87]  
]);  
// Load the Dynamic World dataset for the year 2022. This dataset provides global land cover  
//classifications  
var dataset = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1')  
  .filterDate('2022-01-01', '2022-12-31') // Selects images from 2022  
  .filterBounds(areaOfInterest); // Filters only images within the study area  
// Extract the land cover classification band and compute the most common class over time  
var landCover = dataset.select('label').mode().clip(areaOfInterest);  
// Apply a mask to remove pixels with no data (values of 0 indicate missing data)  
var landCoverMasked = landCover.updateMask(landCover.gt(0)).unmask(0);  
// Fills no-data pixels with 0  
// Define a kernel (neighborhood window) for spatial analysis  
var kernelSize = 7; // size of the moving window  
var kernel = ee.Kernel.square(kernelSize / 2 + 3, 'pixels', false); // Adds a 3-pixel buffer to the  
window  
// Function to calculate the Shannon diversity index in a moving window  
var calculateShannonOptimized = function(image) {  
  // Compute the total number of pixels in the moving window  
  var totalPixels = ee.Image(1).reduceNeighborhood({  
    reducer: ee.Reducer.sum(),  
    kernel: kernel  
  });  
  // Compute the proportion of each land cover class within the moving window  
  // This generates a set of images where each band represents the proportion of a land cover class  
  var proportions = ee.Image.cat([  
    image.eq(0).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(to-  
totalPixels).rename('class_0'),  
    image.eq(1).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(to-  
totalPixels).rename('class_1'),  
    image.eq(2).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(to-  
totalPixels).rename('class_2'),
```

```
    image.eq(3).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(to-
totalPixels).rename('class_3'),
    image.eq(4).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(to-
totalPixels).rename('class_4'),
    image.eq(5).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(to-
totalPixels).rename('class_5'),
    image.eq(6).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(to-
totalPixels).rename('class_6'),
    image.eq(7).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(to-
totalPixels).rename('class_7'),
    image.eq(8).reduceNeighborhood({reducer: ee.Reducer.sum(), kernel: kernel}).divide(to-
totalPixels).rename('class_8')
  ]);
  // Compute the Shannon diversity index using the equation.
  var shannonIndex = proportions.expression(
    '-1 * (class_0 * log(class_0 + 1e-6) + ' +
    'class_1 * log(class_1 + 1e-6) + ' +
    'class_2 * log(class_2 + 1e-6) + ' +
    'class_3 * log(class_3 + 1e-6) + ' +
    'class_4 * log(class_4 + 1e-6) + ' +
    'class_5 * log(class_5 + 1e-6) + ' +
    'class_6 * log(class_6 + 1e-6) + ' +
    'class_7 * log(class_7 + 1e-6) + ' +
    'class_8 * log(class_8 + 1e-6)', {
    'class_0': proportions.select('class_0'),
    'class_1': proportions.select('class_1'),
    'class_2': proportions.select('class_2'),
    'class_3': proportions.select('class_3'),
    'class_4': proportions.select('class_4'),
    'class_5': proportions.select('class_5'),
    'class_6': proportions.select('class_6'),
    'class_7': proportions.select('class_7'),
    'class_8': proportions.select('class_8')
  });
  // Masking to ensure only valid areas are considered
  return shannonIndex.updateMask(totalPixels.gt(0)).max(0);
};
// Generate a grid over the study area to divide it into smaller analysis units
var scale = 50000; // Defines the size of each block (50 km)
var projection = ee.Projection('EPSG:3035'); // Uses the LAEA projection system
var grid = areaOfInterest.coveringGrid({
  proj: projection,
  scale: scale
});
// Convert the grid into a FeatureCollection
var gridFeatures = ee.FeatureCollection(
  grid.map(function(geometry) {
    return ee.Feature(geometry);
  })
);
// Display the grid on the map
Map.centerObject(areaOfInterest, 10);
```

```
Map.addLayer(gridFeatures, {}, 'Grid');
// Process each block separately and export results
gridFeatures.toList(gridFeatures.size()).evaluate(function(tiles) {
  tiles.forEach(function(tile, index) {
    var block = ee.Feature(tile).geometry();
    // Add a buffer to the block to minimize edge effects
    var bufferedBlock = block.buffer(scale / 10); // 10% buffer
    // Clip the land cover map to the block area
    var landCoverMaskedBlock = landCoverMasked.clip(bufferedBlock);

    // Compute the Shannon index for the block
    var shannonResultBlock = calculateShannonOptimized(landCoverMaskedBlock);
    // Clip the result to match the original block area
    var croppedResultBlock = shannonResultBlock.clip(block);
    // Export the computed Shannon index as a GeoTIFF file
    Export.image.toDrive({
      image: croppedResultBlock,
      description: 'Shannon_Index_Block_' + index,
      folder: 'EarthEngine_Exports',
      scale: 10, // Uses 10-meter resolution
      region: block,
      crs: 'EPSG:3035',
      maxPixels: 1e12
    });
  });
});
```

## 8.4 Appendix. Python (Colab) code for Mean Patch Area index calculation.

```
import numpy as np
import rasterio
from rasterio import features
from scipy.ndimage import label
from osgeo import gdal
import os
import multiprocessing
from joblib import Parallel, delayed

def mpa_by_pixel(i, j, land_cover_block, kernel_radius):
    """
    Computes the Mean Patch Area (MPA) for a given pixel by analyzing a 7x7 neighborhood.
    Parameters:
    - i, j: Coordinates of the pixel within the block.
    - land_cover_block: 2D NumPy array representing the land cover raster block.
    - kernel_radius: Radius of the neighborhood kernel (3 pixels for a 7x7 window).
    Returns:
    - Mean patch area (m2) of the identified patches within the 7x7 window.
```

```
- NaN if the window contains only nodata values.
"""
# Extract the 7x7 neighborhood around the current pixel
kernel = land_cover_block[i - kernel_radius:i + kernel_radius + 1,
                          j - kernel_radius:j + kernel_radius + 1]
# Skip computation if the entire kernel contains only nodata values (-1)
if np.all(kernel == -1):
    return np.nan
# Identify unique land cover classes within the neighborhood (excluding nodata)
unique_classes = np.unique(kernel[kernel >= 0])
patch_areas = [] # Store the area of patches found in the kernel
for cls in unique_classes:
    # Create a binary mask for the specific land cover class
    binary_kernel = (kernel == cls).astype(int)
    # Label connected patches of the same class
    labeled_patches, num_features = label(binary_kernel)
    # Compute the area of each patch (assuming 10x10 m per pixel)
    for patch_id in range(1, num_features + 1):
        patch_size = np.sum(labeled_patches == patch_id)
        patch_areas.append(patch_size * 100) # Convert to square meters
# Return the mean patch area for the pixel, or 0 if no patches were found
return np.mean(patch_areas) if patch_areas else 0

def mpa_by_block(input_raster_path, output_folder_path, block_size=5000, start_block=0):
    """
    Computes the Mean Patch Area (MPA) for an input raster by processing it in blocks.
    Parameters:
    - input_raster_path: Path to the input raster file.
    - output_folder_path: Directory where processed blocks will be saved.
    - block_size: Size of each processing block in pixels (default: 5000x5000).
    - start_block: Index of the first block to process (useful for resuming interrupted runs).
    Output:
    - Saves each processed block as a separate raster file.
    """
    # Create the output directory if it does not exist
    if not os.path.exists(output_folder_path):
        os.makedirs(output_folder_path)
    # Open the input raster file
    with rasterio.open(input_raster_path) as src:
        profile = src.profile # Store raster metadata
        nodata_value = src.nodata # Retrieve the nodata value of the raster
        block_id = 0 # Initialize block counter
        # Loop through raster blocks row-wise
        for block_index in range(0, src.height, block_size):
            # Loop through raster blocks column-wise
            for block_col_index in range(0, src.width, block_size):
                # Skip blocks that have already been processed
                if block_id < start_block:
                    block_id += 1
                    continue
                # Define the actual block size (accounting for raster edges)
                block_height = min(block_size + 6, src.height - block_index)
```

```
block_width = min(block_size + 6, src.width - block_col_index)
# Read the current block from the raster (including a 3-pixel buffer)
window = rasterio.windows.Window(
    max(0, block_col_index - 3),
    max(0, block_index - 3),
    block_width,
    block_height
)
land_cover_block = src.read(1, window=window)
# Convert nodata values to -1 for easier processing
land_cover_block = np.where(land_cover_block == nodata_value, -1,
land_cover_block)
# Define the 7x7 neighborhood moving window size/kernel size
kernel_size = 7
kernel_radius = kernel_size // 2 # 3 pixels (centered window)
# Initialize an array to store the computed MPA values for the block
block_result = np.full((block_height - 6, block_width - 6), fill_value=np.nan,
dtype=float)
# Set up parallel processing to compute MPA for each pixel
num_cores = multiprocessing.cpu_count()
num_cores = 16 # number of cores for optimal performance

results = Parallel(n_jobs=num_cores)(
    delayed(mpa_by_pixel)(i, j, land_cover_block, kernel_radius)
    for i in range(kernel_radius, block_height - kernel_radius)
    for j in range(kernel_radius, block_width - kernel_radius)
)
# Store results in the corresponding block result array
index = 0
for i in range(kernel_radius, block_height - kernel_radius):
    for j in range(kernel_radius, block_width - kernel_radius):
        block_result[i - kernel_radius, j - kernel_radius] = results[index]
        index += 1
# Update raster metadata for the output block
block_profile = profile.copy()
block_profile.update(
    height=block_height - 6,
    width=block_width - 6,
    transform=rasterio.windows.transform(window, src.transform),
    dtype=rasterio.float32,
    nodata=-9999 # Define a new nodata value for output
)
# Define the output path for the processed block
block_output_path = f"{output_folder_path}/MPA_block_{block_id}.tif"
# Write the computed MPA values to a new raster file
with rasterio.open(block_output_path, 'w', **block_profile) as dst:
    dst.write(block_result, 1)
print(f"Block {block_id} calculated and saved to: {block_output_path}")
block_id += 1

# Define input raster path and output folder
input_raster = "/content/drive/My Drive/dw_2022/dw_2022_mosaic_bB.tif"
```

```
output_folder = "/content/drive/My Drive/dw_2022/blocks_mpa4"  
# Start processing from block = number  
start_block = 1 #This is especially useful when suddenly the calculation stop.  
# Run the block processing function  
mpa_by_block(input_raster, output_folder, start_block=start_block)
```

## 8.5 Appendix. Python (Colab) code for Contagion index calculation.

```
import numpy as np  
import rasterio  
from rasterio.windows import Window  
import os  
import multiprocessing  
from joblib import Parallel, delayed  
  
# Function to calculate the Contagion Index for a given 7x7 window  
def calculate_contagion(values, n_classes=9):  
    """  
    Computes the Contagion Index for a given window of land cover values.  
    Parameters:  
    - values: 1D NumPy array representing a flattened 7x7 pixel neighborhood.  
    - n_classes: Number of land cover classes (default is 9).  
    Returns:  
    - Contagion Index value (float) if valid, otherwise NaN.  
    """  
    # If the entire window is NaN (no data), return NaN  
    if np.all(np.isnan(values)):  
        return np.nan  
    # Reshape the flattened array into a 7x7 matrix  
    size = int(np.sqrt(len(values)))  
    matrix = values.reshape((size, size))  
    # Initialize adjacency matrix to count transitions between classes  
    adjacency_matrix = np.zeros((n_classes, n_classes))  
    # Iterate through each pixel in the 7x7 window  
    for i in range(size):  
        for j in range(size):  
            current_class = matrix[i, j]  
            # Ensure valid class values (non-NaN and within the class range)  
            if not np.isnan(current_class) and 0 <= current_class < n_classes and current_class.is_in-  
teger():  
                neighbors = []  
                # Check 8-directional adjacency (including diagonals)  
                if i > 0: neighbors.append(matrix[i - 1, j]) # Up  
                if i < size - 1: neighbors.append(matrix[i + 1, j]) # Down  
                if j > 0: neighbors.append(matrix[i, j - 1]) # Left  
                if j < size - 1: neighbors.append(matrix[i, j + 1]) # Right  
                if i > 0 and j > 0: neighbors.append(matrix[i - 1, j - 1]) # Up-Left
```

```
if i > 0 and j < size - 1: neighbors.append(matrix[i - 1, j + 1]) # Up-Right
if i < size - 1 and j > 0: neighbors.append(matrix[i + 1, j - 1]) # Down-Left
if i < size - 1 and j < size - 1: neighbors.append(matrix[i + 1, j + 1]) # Down-Right

# Update adjacency counts for neighboring land cover classes
for neighbor in neighbors:
    if not np.isnan(neighbor) and 0 <= neighbor < n_classes and neighbor.is_integer():
        adjacency_matrix[int(current_class), int(neighbor)] += 1
# Compute the total number of adjacency transitions
total_adjacencies = np.sum(adjacency_matrix)
# If there are no valid adjacencies, return NaN
if total_adjacencies == 0:
    return np.nan
# Calculate the probability of each adjacency transition
p_ij = adjacency_matrix / total_adjacencies
# Compute the Contagion Index using entropy formula
p_ij_log = np.where(p_ij > 0, p_ij * np.log(p_ij), 0)
contagion_raw = np.sum(p_ij_log)
contagion_index = 1 + (contagion_raw / (2 * np.log(n_classes)))
return contagion_index
# Function to process a single pixel in a raster block
def process_pixel(i, j, block, buffer_size, n_classes=9):
    """
    Extracts a 7x7 window around a pixel and computes its Contagion Index.
    Parameters:
    - i, j: Pixel coordinates within the raster block.
    - block: NumPy array representing the land cover raster block.
    - buffer_size: Half-size of the moving window (7x7 → 3-pixel buffer).
    - n_classes: Number of land cover classes.
    Returns:
    - Tuple (i, j, contagion_value) where contagion_value is the computed Contagion Index.
    """

    # Extract 7x7 neighborhood window
    window_values = block[i - buffer_size:i + buffer_size + 1, j - buffer_size:j + buffer_size + 1]
    # Compute Contagion Index for the extracted window
    return i, j, calculate_contagion(window_values.flatten(), n_classes=n_classes)

# Block processing parameters
window_size = 7 # size moving window
half_window = window_size // 2 # 3-pixel buffer to account for edge effects
buffer_size = half_window
block_size = 5000 # Process raster in blocks of 5000x5000 pixels

# Specify the starting block (in terms of row and column)
start_row = 0 # Row index to start processing from
start_col = 0 # Column index to start processing from

# Input raster file and output directory
input_file = "/content/drive/My Drive/dw_2022/dw_2022_mosaicContagB.tif"
output_dir = "/content/drive/My Drive/dw_2022/blocks_contagionD"
os.makedirs(output_dir, exist_ok=True) # Create output directory if not exists
```

```
# Open the input raster file
with rasterio.open(input_file) as src:
    profile = src.profile # Copy raster metadata
    profile.update(dtype=rasterio.float32, count=1, nodata=np.nan) # Update to float32 format

# Iterate through raster in blocks (starting from the specified row/column)
for row_start in range(start_row, src.height, block_size - 2 * buffer_size):
    for col_start in range(start_col, src.width, block_size - 2 * buffer_size):
        # Define the block size, considering the image boundaries
        row_end = min(row_start + block_size, src.height)
        col_end = min(col_start + block_size, src.width)

        # Define the block window (including buffer)
        window = Window(col_start, row_start, col_end - col_start, row_end - row_start)
        block = src.read(1, window=window)
        # Skip empty blocks (all NaN)
        if np.all(np.isnan(block)):
            print(f"Skipping empty block at rows {row_start}-{row_end} and cols {col_start}-
{col_end}")
            continue

        # Create an output array for storing Contagion index results
        output_block = np.full_like(block, np.nan, dtype=np.float32)
        # Get block dimensions
        block_height, block_width = block.shape
        # Set up parallel processing (use up to 16 CPU cores)
        num_cores = min(16, multiprocessing.cpu_count())
        results = Parallel(n_jobs=num_cores)(
            delayed(process_pixel)(i, j, block, buffer_size, n_classes=9)
            for i in range(buffer_size, block_height - buffer_size)
            for j in range(buffer_size, block_width - buffer_size)
        )

        # Assign the computed Contagion Index values to the output block
        for i, j, contagion_value in results:
            output_block[i, j] = contagion_value
        # Remove the buffer before saving the block
        core_output_block = output_block[
            buffer_size:block.shape[0] - buffer_size,
            buffer_size:block.shape[1] - buffer_size
        ]

        # Skip saving if there is no valid data
        if np.all(np.isnan(core_output_block)):
            print(f"No valid data in core block at rows {row_start}-{row_end} and cols {col_start}-
{col_end}")
            continue

        # Define output transform for the cropped block
        core_transform = rasterio.windows.transform(
            Window(
```

```
        col_start + buffer_size,  
        row_start + buffer_size,  
        core_output_block.shape[1],  
        core_output_block.shape[0]  
    ),  
    src.transform  
)  
  
# Update raster metadata for output block  
block_profile = profile.copy()  
block_profile.update(  
    height=core_output_block.shape[0],  
    width=core_output_block.shape[1],  
    transform=core_transform  
)  
  
# Define the output file path for the block  
output_path = os.path.join(output_dir, f"block_{row_start}_{col_start}.tif")  
# Save the processed block as a new raster file  
with rasterio.open(output_path, "w", **block_profile) as dst:  
    dst.write(core_output_block, 1)  
    print(f"Saved block: {output_path}")
```

## 8.6 Appendix. R code developed for SSF model fitting.

```
library(sf)      # Spatial vector data handling  
library(amt)    # Animal movement analysis  
library(dplyr)  # Data manipulation and transformation  
library(raster) # Raster data processing  
library(spatstat) # Spatial point pattern analysis  
library(sp)     # Spatial data classes and methods  
library(terra)  # Raster and vector processing  
library(tibble) # Improved data frames handling  
library(ggplot2) # Data visualization and plotting  
library(survival) # Survival analysis modeling  
library(lubridate) # Date and time manipulation  
  
#Define work directory  
setwd("Y:/Home/esguerrl/Temp")  
# Load northern lapwing positions  
position <- st_read("Y:/Home/esguerrl/locations_breeding/locations_points.shp")  
# Load raster layers: Land Use /Land cover (LULC), Diversity Shannon Index, Mean Patch Area  
# Index, and Contagion Index  
landcover <- raster("Y:/Home/esguerrl/dw_2022_mosaic/LandcoverDW2022.tif")  
shannon <- raster("Y:/Home/esguerrl/Indices_Final/Shannon_Total.tif")  
mean_patch_area <- raster("Y:/Home/esguerrl/Indices_Final/MPA_Resampled.tif")  
contagio <- raster("Y:/Home/esguerrl/Indices_Final/Contagion_Resampled.tif")  
elevation <- raster("Y:/Home/esguerrl/Indices_Final/dem30m_merged.tif")
```

```
# Convert the required data types of the fields x and y coordinates, timestamp (t) and individual
id (id)
position$x <- as.numeric(position$x)
position$y <- as.numeric(position$y)
position$t <- as.POSIXct(position$t, format="%Y-%m-%d %H:%M:%S")
position$id <- as.factor(position$id)
position$geometry <- NULL # Delete the field geometry to avoid errors during track creation

# Filter positions for year. It's necessary to set the year of analysis
position <- position %>% filter(year(t) == 2021)
year_filter <- unique(year(position$t))
cat("Filtered positions for the year... ", year_filter, "\n")
# Validate the number of positions remaining after filtering
cat("Number of positions for the year ", year_filter, ": ", nrow(position), "\n")
# Unique individual IDs
individuos <- unique(position$id)
head(individuos)

# Uncomment the rates and tolerances for the year required for the analysis.
##### Define rates, tolerance per individual year 2021#####
config <- data.frame(
  individuo = c("5181", "5264", "5268", "5270", "5277", "5278", "5282", "5284", "5285", "5286",
"5301", "5303", "5308", "5439"),
  resample_rate = c(7, 10, 2, 2, 2, 2, 2, 30, 2, 2, 2, 2, 2), # Resample time in minutes
  resample_tolerance = c(42, 60, 12, 12, 12, 12, 12, 12, 180, 12, 12, 12, 12, 12) # Resample
tolerance in seconds
)

##### Define rates, tolerance per individual year 2022#####
#config <- data.frame(
# individuo = c("5181", "5264", "5285", "5301", "5303", "5439" ),
# resample_rate = c(60, 2, 2, 2, 2, 10), # Resample time in minutes
# resample_tolerance = c(360, 10, 10, 10, 10, 60) # Resample tolerance in seconds
#)
# Loop to create subsets, tracks, bursts, and steps for each individual
for (individuo in individuos) {
  cat("\nProcessing individual: ", individuo, "\n")

  # Set the individual-specific resampling parameters according to the data frame "config".
  rate <- config$resample_rate[config$individuo == individuo]
  tolerance <- config$resample_tolerance[config$individuo == individuo]
  cat("Rate (resample): ", rate, "minutes\n")
  cat("Tolerance (resample): ", tolerance, "seconds\n")

  # Filter data for the current individual and store it locally in position_ind_xxxx.rds
  cat("Filtering data for individual: ", individuo, "\n")
  position_ind <- position %>%
    filter(id == individuo) %>%
    filter(!is.na(t)) # Remove row with NA in the timestamps column `t`
  saveRDS(position_ind, file = paste0("position_ind_", individuo, "_", year_filter, ".rds"))
  cat("Number of points for individual ", individuo, ": ", nrow(position_ind), "\n")
  rm(position_ind)
```

```
gc()

# From the individual data stored locally (rds file), create the tracks and bursts
track_ind <- readRDS(paste0("position_ind_", individuo, "_", year_filter, ".rds")) %>%
  # Creating track (track_xyt format)
  make_track(.x = x, .y = y, .t = t, id = id, crs = 3035) %>%
  # Resample data according to the config per individual
  track_resample(rate = minutes(rate), tolerance = seconds(tolerance))
head(track_ind)
# Save the data in local directory
saveRDS(track_ind, file = paste0("track_ind_", individuo, "_", year_filter, ".rds"))
rm(track_ind)
gc()

# Adjust overlapped consecutive positions (step length=0), adding 0.01 m to the (x,y) coordi-
nates.
# Stored as bursts "burst_ind_xxxx.rds".
burst_ind <- readRDS(paste0("track_ind_", individuo, "_", year_filter, ".rds")) %>%
  mutate(
    x_ = ifelse(lead(x_) == x_ & lead(y_) == y_, x_ + 0.01, x_),
    y_ = ifelse(lead(x_) == x_ & lead(y_) == y_, y_ + 0.01, y_)
  )
print(head(burst_ind)) # Debugging: Print first rows
saveRDS(burst_ind, file = paste0("burst_ind_", individuo, "_", year_filter, ".rds"))
rm(burst_ind)
gc()

# Create used and available steps and store them.
cat("Creating used and available steps for individual: ", individuo, "\n")
burst_ind <- readRDS(paste0("burst_ind_", individuo, "_", year_filter, ".rds"))
# Validar si hay suficientes bursts para procesar
if (n_distinct(burst_ind$burst_) < 2) {
  cat("Skipping individual", individuo, "as there is only one burst.\n")
  next # Saltar al siguiente individuo
}
# Delete bursts with less than 3 positions, otherwise it will show an error.
burst_ind <- burst_ind %>%
  group_by(burst_) %>%
  filter(n() >= 3) %>%
  ungroup()

# Debugging: Print the structure and unique values of burst_
print("Structure of burst_:")
print(str(burst_ind$burst_))
print("Unique values of burst_:")
print(unique(burst_ind$burst_))
burst_ind$burst_ <- as.integer(burst_ind$burst_) # Ensure burst_ is treated as an integer

#From tracks and bursts, create the random steps. 10 random step by 1 used.
steps_ind <- burst_ind %>%
  steps_by_burst(burst_ = burst_) %>%
```

```
random_steps(n = 10) %>%
group_by(step_id_) %>%
mutate(
  log_sl_ = log(sl_),
  cos_ta_ = cos(ta_)
) %>%
ungroup() %>%
filter(!is.na(x2_) & !is.na(y2_) & !is.na(log_sl_) & !is.na(cos_ta_))
saveRDS(steps_ind, file = paste0("steps_ind_", individuo, "_", year_filter, ".rds"))
rm(steps_ind)
gc()

# Convert steps to sf (simple feature) for raster extraction and store it
cat("Converting steps to sf for individual: ", individuo, "\n")
steps_sf <- readRDS(paste0("steps_ind_", individuo, "_", year_filter, ".rds")) %>%
  filter(!is.na(x2_) & !is.na(y2_)) %>% # Remove rows with "na" missing coordinates before
conversion
  st_as_sf(coords = c("x2_", "y2_"), crs = 3035, remove = FALSE)
# remove = FALSE to keep the x2 and y2 columns in the original dataframe

# Extract raster values for each step
cat("Extracting raster values for individual: ", individuo, "\n")
steps_sf$landcover <- raster::extract(landcover, steps_sf)
steps_sf$shannon <- raster::extract(shannon, steps_sf)
steps_sf$mean_patch_area <- raster::extract(mean_patch_area, steps_sf)
steps_sf$contagio <- raster::extract(contagio, steps_sf)
steps_sf$elevation <- raster::extract(elevation, steps_sf)

  cat("Reclassifying landcover and removing NAs for individual: ", individuo, "\n")
# Specify the landcover classes of interest as numerical codes, where 0: water; 1: trees; 2: grass;
3: flooded vegetation; 4: crops, 5: Shrub & Scrub, 6: Built area. The class 6 is used as a reference
level for the other classes
selected_classes <- c(6, 0, 1, 2, 3, 4, 5)
# Filter and clean the data for the selected landcover classes
steps_sf <- steps_sf %>%
  filter(landcover %in% selected_classes) %>% # Retain only rows where landcover is in the
selected classes
  mutate(
    # Reclassify landcover into a factor with meaningful labels: "Class_0", "Class_1", etc.
    landcover = factor(landcover, levels = selected_classes, labels = paste0("Class_", se-
lected_classes)) # convert landcover as a categorical variable type factor
  ) %>%
  na.omit() # Remove rows with missing values in any column
# Save the processed data for the individual to an rds file steps_sf_xxxx.rds
saveRDS(steps_sf, file = paste0("steps_sf_", individuo, "_", year_filter, ".rds"))
rm(steps_sf)
gc()
}

# Loop to fit multiple SSF models for each individual
for(individuo in individuos) {
  # Get the steps for the current individual
```

```
steps_sf <- readRDS(paste0("steps_sf_", individuo, "_", year_filter, ".rds"))
print(head(steps_sf))

# Fit a SSF model for each landscape metric
models <- list(
  model_1 = steps_sf %>% fit_issf(case_ ~ landcover + shannon + elevation + log_sl_ + cos_ta_
+ strata(step_id_), model = TRUE),
  model_2 = steps_sf %>% fit_issf(case_ ~ landcover + mean_patch_area + elevation + log_sl_
+ cos_ta_ + strata(step_id_), model = TRUE),
  model_3 = steps_sf %>% fit_issf(case_ ~ landcover + contagio + elevation + log_sl_ + cos_ta_
+ strata(step_id_), model = TRUE)
)

# Store each fitted model
for (model_name in names(models)) {
  saveRDS(models[[model_name]], file = paste0(model_name, "_", individuo, "_", year_filter,
".rds"))
}
rm(models)
gc()
}

# Summarize the models for each individual
for (individuo in individuos) {
  for (model_name in c("model_1", "model_2", "model_3")) {
    cat("\nSummary for Individual: ", individuo, " - Model: ", model_name, "\n")
    model_ind <- readRDS(paste0(model_name, "_", individuo, "_", year_filter, ".rds"))
    print(summary(model_ind))
    rm(model_ind)
    gc()
  }
}
```

## 8.7 Appendix. R script to perform K-Fold cross validation of SSF model fitting, using AUC metric.

```
#####
#I strongly recommend to run the validation in the same R space than the SSF models fitting, this
make easier rebuilt the formulas and the input data.
#####
library(dplyr)
library(pROC)

# Set the year of the fitted models outputs
year_filter <- 2021
# Loop to fit multiple SSF models for each individual
validation_results <- data.frame() # DataFrame to store validation results
# Iterate over each individual
for (individuo in individuos) {
  cat("Processing individual:", individuo, "\n")
```

```
# Load the dataset for the current individual
steps_sf <- readRDS(paste0("steps_sf_", individuo, "_", year_filter, ".rds"))
print(head(steps_sf)) # Print the first rows for debugging
# Fit multiple SSF models using different predictor combinations
models <- list(
  model_1 = steps_sf %>% fit_issf(case_ ~ landcover + shannon + elevation + log_sl_ + cos_ta_
+ strata(step_id_), model = TRUE),
  model_2 = steps_sf %>% fit_issf(case_ ~ landcover + mean_patch_area + elevation + log_sl_
+ cos_ta_ + strata(step_id_), model = TRUE),
  model_3 = steps_sf %>% fit_issf(case_ ~ landcover + contagio + elevation + log_sl_ + cos_ta_
+ strata(step_id_), model = TRUE)
)

# Save each fitted model to disk for later use
for (model_name in names(models)) {
  saveRDS(models[[model_name]], file = paste0(model_name, "_", individuo, "_", year_filter,
".rds"))
}
# Cross-validation process for each model
for (model_name in names(models)) {
  cat("\nValidating model:", model_name, "for Individual:", individuo, "\n")
  model <- models[[model_name]]
  # Create folds for cross-validation by randomly assigning fold IDs
  steps_sf <- steps_sf %>%
  group_by(step_id_) %>% # Ensure each step_id remains within a single fold
  mutate(fold_id = sample(rep(1:10, length.out = n())) %>%
  ungroup()

  auc_values <- c() # List to store AUC values for each fold

  for (fold in 1:10) {
    # Split the dataset into training and testing subsets
    train_data <- steps_sf %>% filter(fold_id != fold) # Training set (9/10 of the data)
    test_data <- steps_sf %>% filter(fold_id == fold) # Testing set (1/10 of the data)
    if (nrow(train_data) > 0 && nrow(test_data) > 0) {
      formula_actual <- switch(model_name,
        "model_1" = case_ ~ landcover + shannon + elevation + log_sl_ + cos_ta_ +
strata(step_id_),
        "model_2" = case_ ~ landcover + mean_patch_area + elevation + log_sl_ +
cos_ta_ + strata(step_id_),
        "model_3" = case_ ~ landcover + contagio + elevation + cos_ta_ +
strata(step_id_)
      )
      # Fit the model using the training data
      fold_model <- fit_issf(formula_actual, data = train_data, model = TRUE)
      fold_model_clogit <- fold_model$model # Extract the internal clogit model

      # Generate predictions for the test dataset
      test_data <- test_data %>% mutate(
        relative_prob = exp(predict(fold_model_clogit, newdata = test_data, type = "lp"))
      )
    }
  }
}
```

```
# Compute AUC only if the test dataset contains both used (case_ = 1)
# and available (case_ = 0) steps. If the test data lacks one of these classes,
# AUC cannot be computed, so it is set to NA.
if (length(unique(test_data$case_)) > 1) {
  auc <- tryCatch({
    # Compute the AUC (Area Under the Curve) to evaluate how well the model
    # distinguishes between used and available steps.
    pROC::roc(test_data$case_, test_data$relative_prob)$auc
  }, error = function(e) {
    NA # If an error occurs (e.g., due to insufficient data), assign NA to AUC.
  })
} else {
  auc <- NA
}

auc_values <- c(auc_values, auc) # Store AUC values from each fold
}
}

# Compute the average AUC across all folds as a measure of model performance
mean_auc <- mean(auc_values, na.rm = TRUE)
cat("Mean AUC for", model_name, "on individual", individuo, ":", mean_auc, "\n")

# Store validation results in the DataFrame
validation_results <- bind_rows(validation_results, data.frame(
  Individual = individuo,
  Model = model_name,
  Mean_AUC = mean_auc
))
}
rm(models) # Remove stored models from memory
gc() # Perform garbage collection to free up memory
}
# Save the cross-validation results as a CSV file
write.csv(validation_results, "Y:/Home/esguerrl/Temp_prueba/CV_seguridad.csv", row.names =
FALSE)
```

## 8.8 Appendix. R script to perform AIC calculation iteratively in a set of fitted models using the generic function *AIC()*.

```
# Load required libraries
library(dplyr)
library(purrr)

# Directory where the files are located
path <- "Y:/Home/esguerrl/temp_prueba" # Change this to your folder path
# List files with a specific pattern
```

```
files <- list.files(path, pattern = "model_\\d+_\\d+_\\d{4}\\rds$", full.names = TRUE)
# Create a list to store models and their AIC values
model_list <- list()
aic_values <- list()

# Loop to load files and calculate AIC
for (file in files) {
  # Extract information from the file name
  base_name <- gsub("model_(\\d+)_(\\d+)_(\\d{4})\\rds", "see\\2_\\3m\\1", basename(file))
  # Read the .rds file
  model <- readRDS(file)
  # Store the model with the specific name
  model_list[[base_name]] <- model
  # Calculate AIC and store it
  aic_values[[base_name]] <- AIC(model)
}

# Convert the AIC list into a data frame for easier visualization
aic_df <- data.frame(Model = names(aic_values), AIC = unlist(aic_values))
# Save the AIC results to a CSV file
write.csv(aic_df, file = "Y:/Home/esguerrl/temp_prueba/AIC_results.csv", row.names = FALSE)
# Display the result
print(aic_df)
```

## 8.1 Appendix. Standard outputs of SSF fitted models.

Model 1: landcover + shannon + elevation + log_sl_ + cos_ta_ + strata(step_id_). Data: 2021								
<i>Individual 5181 (n = 68306, number of events = 6623)</i>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
<b>Water class</b>	0.124	1.132	0.351	0.725	0.569	2.252		
<b>Trees class</b>	1.789	5.981	0.377	<0.001	2.856	12.523		
<b>Grass class</b>	5.235	187.779	0.32	<0.001	100.2	351.907		
<b>Flooded vegetation class</b>	4.586	98.148	0.346	<0.001	49.81	193.395		
<b>Crops class</b>	3.659	38.84	0.321	<0.001	20.711	72.838	0.907 (se =	16249.29,
<b>Shrub &amp; Scrub class</b>	3.662	38.939	0.517	<0.001	14.143	107.204	0.002)	p=<0.001
<b>shannon</b>	-1.521	0.218	0.061	<0.001	0.194	0.246		
<b>elevation</b>	-0.007	0.993	0.003	0.007	0.988	0.998		
<b>log_sl_</b>	0.247	1.28	0.009	<0.001	1.257	1.303		
<b>cos_ta_</b>	-1.735	0.176	0.027	<0.001	0.167	0.186		
<i>Individual 5264 (n = 63701, number of events = 6010)</i>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
<b>Water class</b>	-0.013	0.987	0.287	0.965	0.563	1.732		
<b>Trees class</b>	1.538	4.656	0.299	<0.001	2.591	8.368		
<b>Grass class</b>	4.055	57.695	0.246	<0.001	35.644	93.386	0.777 (se =	6066.05,
<b>Flooded vegetation class</b>	3.705	40.658	0.27	<0.001	23.948	69.03	0.004)	p=<0.001
<b>Crops class</b>	3.035	20.792	0.246	<0.001	12.837	33.677		

<b>Shrub &amp; Scrub class</b>	3.415	30.418	0.406	<0.001	13.729	67.392		
<b>shannon</b>	-1.193	0.303	0.057	<0.001	0.271	0.339		
<b>elevation</b>	-0.014	0.986	0.003	<0.001	0.981	0.992		
<b>log_sl_</b>	0.101	1.106	0.004	<0.001	1.097	1.115		
<b>cos_ta_</b>	-0.829	0.436	0.021	<0.001	0.419	0.455		
<b>Individual 5268 (n = 191256, number of events = 17681)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.519	1.681	0.216	0.016	1.1	2.569		
<b>Trees class</b>	1.962	7.116	0.219	<0.001	4.636	10.922		
<b>Grass class</b>	4.288	72.812	0.191	<0.001	50.092	105.835		
<b>Flooded vegetation class</b>	3.599	36.551	0.206	<0.001	24.387	54.782		
<b>Crops class</b>	3.23	25.277	0.191	<0.001	17.378	36.767	0.765 (se =	15718.26,
<b>Shrub &amp; Scrub class</b>	3.489	32.761	0.28	<0.001	18.927	56.708	0.002)	p=<0.001
<b>shannon</b>	-0.969	0.379	0.035	<0.001	0.354	0.407		
<b>elevation</b>	-0.022	0.979	0.003	<0.001	0.973	0.985		
<b>log_sl_</b>	0.05	1.051	0.002	<0.001	1.047	1.055		
<b>cos_ta_</b>	-0.918	0.399	0.012	<0.001	0.39	0.409		
<b>Individual 5270 (n = 191282, number of events = 17681)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.523	1.686	0.218	0.017	1.1	2.586		
<b>Trees class</b>	1.959	7.094	0.22	<0.001	4.607	10.922		
<b>Grass class</b>	4.316	74.855	0.193	<0.001	51.304	109.218		
<b>Flooded vegetation class</b>	3.611	37.01	0.208	<0.001	24.604	55.672		
<b>Crops class</b>	3.236	25.432	0.193	<0.001	17.419	37.133	0.765 (se =	15801.05,
<b>Shrub &amp; Scrub class</b>	3.522	33.864	0.279	<0.001	19.597	58.518	0.002)	p=<0.001
<b>shannon</b>	-0.985	0.374	0.035	<0.001	0.348	0.4		
<b>elevation</b>	-0.022	0.979	0.003	<0.001	0.973	0.984		
<b>log_sl_</b>	0.052	1.053	0.002	<0.001	1.05	1.057		
<b>cos_ta_</b>	-0.915	0.401	0.012	<0.001	0.391	0.41		
<b>Individual 5277 (n = 191165, number of events = 17681)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.477	1.612	0.216	0.027	1.055	2.464		
<b>Trees class</b>	1.951	7.037	0.219	<0.001	4.583	10.806		
<b>Grass class</b>	4.258	70.676	0.19	<0.001	48.661	102.649		
<b>Flooded vegetation class</b>	3.544	34.614	0.206	<0.001	23.103	51.86		
<b>Crops class</b>	3.199	24.52	0.191	<0.001	16.869	35.641	0.765 (se =	15689.76,
<b>Shrub &amp; Scrub class</b>	3.439	31.144	0.281	<0.001	17.968	53.983	0.002)	p=<0.001
<b>shannon</b>	-0.981	0.375	0.035	<0.001	0.35	0.402		
<b>elevation</b>	-0.023	0.978	0.003	<0.001	0.973	0.983		
<b>log_sl_</b>	0.048	1.049	0.002	<0.001	1.045	1.052		
<b>cos_ta_</b>	-0.917	0.4	0.012	<0.001	0.391	0.409		
<b>Individual 5278 (n = 191145, number of events = 17681)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.495	1.641	0.216	0.022	1.074	2.508		
<b>Trees class</b>	1.985	7.276	0.219	<0.001	4.741	11.166		
<b>Grass class</b>	4.286	72.69	0.19	<0.001	50.045	105.581	0.767 (se =	15859.75,
<b>Flooded vegetation class</b>	3.634	37.851	0.206	<0.001	25.27	56.695	0.002)	p=<0.001

<b>Crops class</b>	3.23	25.273	0.191	<0.001	17.388	36.735		
<b>Shrub &amp; Scrub class</b>	3.656	38.721	0.279	<0.001	22.399	66.936		
<b>shannon</b>	-0.982	0.375	0.035	<0.001	0.35	0.402		
<b>elevation</b>	-0.023	0.977	0.003	<0.001	0.971	0.983		
<b>log_sl_</b>	0.049	1.051	0.002	<0.001	1.047	1.054		
<b>cos_ta_</b>	-0.93	0.395	0.012	<0.001	0.385	0.404		
<b>Individual 5282 (n = 191295, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
<b>Water class</b>	0.503	1.654	0.218	0.021	1.078	2.536		
<b>Trees class</b>	2.007	7.443	0.219	<0.001	4.843	11.438		
<b>Grass class</b>	4.299	73.657	0.192	<0.001	50.514	107.404		
<b>Flooded vegetation class</b>	3.613	37.09	0.208	<0.001	24.664	55.777		
<b>Crops class</b>	3.241	25.558	0.193	<0.001	17.516	37.294	0.766 (se =	15784.13,
<b>Shrub &amp; Scrub class</b>	3.515	33.615	0.279	<0.001	19.459	58.068	0.002)	p=<0.001
<b>shannon</b>	-0.988	0.372	0.035	<0.001	0.347	0.399		
<b>elevation</b>	-0.026	0.975	0.003	<0.001	0.968	0.981		
<b>log_sl_</b>	0.051	1.052	0.002	<0.001	1.049	1.056		
<b>cos_ta_</b>	-0.921	0.398	0.012	<0.001	0.389	0.407		
<b>Individual 5284 (n = 191142, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
<b>Water class</b>	0.497	1.644	0.218	0.023	1.072	2.521		
<b>Trees class</b>	2.033	7.636	0.221	<0.001	4.948	11.783		
<b>Grass class</b>	4.327	75.754	0.194	<0.001	51.827	110.727		
<b>Flooded vegetation class</b>	3.653	38.602	0.209	<0.001	25.623	58.156		
<b>Crops class</b>	3.281	26.605	0.194	<0.001	18.19	38.914	0.766 (se =	15810.50,
<b>Shrub &amp; Scrub class</b>	3.615	37.155	0.279	<0.001	21.485	64.252	0.002)	p=<0.001
<b>shannon</b>	-0.961	0.383	0.035	<0.001	0.357	0.41		
<b>elevation</b>	-0.025	0.976	0.003	<0.001	0.97	0.982		
<b>log_sl_</b>	0.051	1.053	0.002	<0.001	1.049	1.056		
<b>cos_ta_</b>	-0.92	0.398	0.012	<0.001	0.389	0.408		
<b>Individual 5285 (n = 21070, number of events = 2033)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
<b>Water class</b>	0.084	1.087	0.575	0.884	0.352	3.358		
<b>Trees class</b>	1.47	4.347	0.618	0.017	1.294	14.609		
<b>Grass class</b>	4.755	116.17	0.505	<0.001	43.161	312.675		
<b>Flooded vegetation class</b>	4.494	89.463	0.536	<0.001	31.272	255.938		
<b>Crops class</b>	3.536	34.346	0.506	<0.001	12.747	92.541	0.841 (se =	3208.03,
<b>Shrub &amp; Scrub class</b>	4.177	65.15	0.683	<0.001	17.07	248.65	0.006)	p=<0.001
<b>shannon</b>	-1.226	0.293	0.097	<0.001	0.242	0.355		
<b>elevation</b>	-0.008	0.992	0.004	0.055	0.985	1		
<b>log_sl_</b>	0.153	1.165	0.01	<0.001	1.143	1.189		
<b>cos_ta_</b>	-1.227	0.293	0.039	<0.001	0.271	0.317		
<b>Individual 5286 (n = 191185, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
<b>Water class</b>	0.482	1.619	0.217	0.026	1.059	2.475		
<b>Trees class</b>	1.979	7.232	0.219	<0.001	4.708	11.11	0.765 (se =	15750.03,
<b>Grass class</b>	4.288	72.788	0.191	<0.001	50.091	105.769	0.002)	p=<0.001

<b>Flooded vegetation class</b>	3.593	36.331	0.206	<0.001	24.25	54.43		
<b>Crops class</b>	3.229	25.248	0.191	<0.001	17.364	36.712		
<b>Shrub &amp; Scrub class</b>	3.459	31.784	0.282	<0.001	18.274	55.28		
<b>shannon</b>	-0.978	0.376	0.035	<0.001	0.351	0.403		
<b>elevation</b>	-0.025	0.975	0.003	<0.001	0.969	0.981		
<b>log_sl_</b>	0.05	1.052	0.002	<0.001	1.048	1.055		
<b>cos_ta_</b>	-0.916	0.4	0.012	<0.001	0.391	0.41		
<b>Individual 5301 (n = 191157, number of events = 17681)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.508	1.663	0.217	0.019	1.087	2.542		
<b>Trees class</b>	2.004	7.419	0.219	<0.001	4.833	11.389		
<b>Grass class</b>	4.282	72.418	0.191	<0.001	49.829	105.249		
<b>Flooded vegetation class</b>	3.64	38.092	0.207	<0.001	25.412	57.1		
<b>Crops class</b>	3.23	25.271	0.191	<0.001	17.376	36.755	0.764 (se =	15606.58,
<b>Shrub &amp; Scrub class</b>	3.52	33.77	0.281	<0.001	19.485	58.527	0.002)	p=<0.001
<b>shannon</b>	-0.989	0.372	0.035	<0.001	0.347	0.398		
<b>elevation</b>	-0.019	0.981	0.003	<0.001	0.976	0.986		
<b>log_sl_</b>	0.049	1.05	0.002	<0.001	1.047	1.054		
<b>cos_ta_</b>	-0.914	0.401	0.012	<0.001	0.391	0.41		
<b>Individual 5303 (n = 191044, number of events = 17681)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.494	1.639	0.218	0.023	1.069	2.513		
<b>Trees class</b>	1.99	7.316	0.22	<0.001	4.752	11.264		
<b>Grass class</b>	4.297	73.492	0.192	<0.001	50.464	107.027		
<b>Flooded vegetation class</b>	3.66	38.859	0.207	<0.001	25.877	58.353		
<b>Crops class</b>	3.24	25.532	0.192	<0.001	17.518	37.213	0.765 (se =	15694.06,
<b>Shrub &amp; Scrub class</b>	3.536	34.34	0.283	<0.001	19.704	59.847	0.002)	p=<0.001
<b>shannon</b>	-0.982	0.375	0.035	<0.001	0.35	0.402		
<b>elevation</b>	-0.02	0.98	0.003	<0.001	0.976	0.985		
<b>log_sl_</b>	0.049	1.05	0.002	<0.001	1.046	1.053		
<b>cos_ta_</b>	-0.917	0.4	0.012	<0.001	0.39	0.409		
<b>Individual 5308 (n = 190976, number of events = 17681)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.463	1.589	0.217	0.033	1.039	2.43		
<b>Trees class</b>	1.999	7.378	0.219	<0.001	4.8	11.341		
<b>Grass class</b>	4.301	73.76	0.191	<0.001	50.739	107.226		
<b>Flooded vegetation class</b>	3.63	37.705	0.207	<0.001	25.153	56.522		
<b>Crops class</b>	3.25	25.799	0.191	<0.001	17.734	37.532	0.765 (se =	15777.56,
<b>Shrub &amp; Scrub class</b>	3.636	37.924	0.279	<0.001	21.942	65.546	0.002)	p=<0.001
<b>shannon</b>	-1.03	0.357	0.035	<0.001	0.333	0.383		
<b>elevation</b>	-0.024	0.976	0.003	<0.001	0.971	0.982		
<b>log_sl_</b>	0.052	1.053	0.002	<0.001	1.05	1.057		
<b>cos_ta_</b>	-0.909	0.403	0.012	<0.001	0.393	0.412		
<b>Individual 5439 (n = 191150, number of events = 17681)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.561	1.752	0.218	0.01	1.144	2.683	0.766 (se =	15634.01,
<b>Trees class</b>	2.011	7.472	0.22	<0.001	4.852	11.507	0.002)	p=<0.001

Grass class	4.33	75.953	0.192	<0.001	52.111	110.703
Flooded vegetation class	3.664	39.015	0.208	<0.001	25.965	58.625
Crops class	3.273	26.377	0.193	<0.001	18.084	38.475
Shrub & Scrub class	3.601	36.644	0.279	<0.001	21.208	63.314
shannon	-0.99	0.372	0.035	<0.001	0.347	0.398
elevation	-0.023	0.978	0.003	<0.001	0.972	0.983
log_sl_	0.049	1.051	0.002	<0.001	1.047	1.054
cos_ta_	-0.915	0.401	0.012	<0.001	0.391	0.41

**Model 1: landcover + shannon + elevation + log\_sl\_ + cos\_ta\_ + strata(step\_id\_). Data: 2022**  
**Individual 5181 (n = 27690, number of events = 2527)**

Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	-1.277	0.279	0.561	0.023	0.093	0.838		
Trees class	2.777	16.066	0.447	<0.001	6.684	38.617		
Grass class	2.658	14.262	0.417	<0.001	6.293	32.323		
Flooded vegetation class	2.731	15.342	0.457	<0.001	6.26	37.603		
Crops class	2.612	13.633	0.415	<0.001	6.045	30.745	0.759 (se =	2145.60,
Shrub & Scrub class	12.007	0	896.668	0.989	0	Inf	0.006)	p=<0.001
shannon	-1.343	0.261	0.104	<0.001	0.213	0.32		
elevation	-0.381	0.683	0.04	<0.001	0.632	0.739		
log_sl_	0.056	1.057	0.006	<0.001	1.045	1.07		
cos_ta_	-1.015	0.362	0.032	<0.001	0.34	0.386		

**Individual 5264 (n = 177942, number of events = 16186)**

Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.684	1.982	0.228	0.003	1.268	3.098		
Trees class	2.012	7.48	0.231	<0.001	4.758	11.759		
Grass class	2.792	16.309	0.204	<0.001	10.943	24.305		
Flooded vegetation class	2.615	13.664	0.216	<0.001	8.95	20.863		
Crops class	2.506	12.253	0.203	<0.001	8.233	18.236	0.686 (se =	6818.73,
Shrub & Scrub class	1.242	3.463	0.502	0.013	1.293	9.27	0.003)	p=<0.001
shannon	-0.814	0.443	0.041	<0.001	0.409	0.48		
elevation	-0.458	0.632	0.021	<0.001	0.607	0.659		
log_sl_	0.018	1.018	0.002	<0.001	1.015	1.021		
cos_ta_	-0.718	0.488	0.012	<0.001	0.476	0.499		

**Individual 5285 (n = 177962, number of events = 16186)**

Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.712	2.037	0.228	0.002	1.304	3.183		
Trees class	2.002	7.405	0.23	<0.001	4.715	11.631		
Grass class	2.838	17.09	0.204	<0.001	11.469	25.467		
Flooded vegetation class	2.64	14.018	0.216	<0.001	9.183	21.398		
Crops class	2.549	12.791	0.203	<0.001	8.593	19.039	0.687 (se =	6756.77,
Shrub & Scrub class	1.256	3.511	0.526	0.017	1.253	9.841	0.003)	p=<0.001
shannon	-0.796	0.451	0.04	<0.001	0.417	0.488		
elevation	-0.45	0.637	0.02	<0.001	0.613	0.663		
log_sl_	0.017	1.017	0.002	<0.001	1.014	1.02		
cos_ta_	-0.707	0.493	0.012	<0.001	0.482	0.505		

<b>Individual 5301 (n = 177949, number of events = 16186)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.735	2.086	0.228	0.001	1.334	3.263		
Trees class	1.995	7.349	0.231	<0.001	4.677	11.546		
Grass class	2.856	17.398	0.204	<0.001	11.661	25.957		
Flooded vegetation class	2.644	14.072	0.216	<0.001	9.208	21.505		
Crops class	2.561	12.948	0.203	<0.001	8.69	19.294	0.687 (se =	6795.40,
Shrub & Scrub class	1.376	3.957	0.508	0.007	1.461	10.721	0.003)	p=<0.001
shannon	-0.795	0.451	0.04	<0.001	0.417	0.489		
elevation	-0.44	0.644	0.02	<0.001	0.619	0.67		
log_sl_	0.018	1.018	0.002	<0.001	1.014	1.021		
cos_ta_	-0.71	0.491	0.012	<0.001	0.48	0.503		

<b>Individual 5303 (n = 177932, number of events = 16186)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.686	1.986	0.228	0.003	1.269	3.108		
Trees class	1.971	7.18	0.232	<0.001	4.559	11.309		
Grass class	2.832	16.979	0.204	<0.001	11.378	25.336		
Flooded vegetation class	2.632	13.898	0.216	<0.001	9.093	21.24		
Crops class	2.525	12.495	0.204	<0.001	8.383	18.624	0.689 (se =	7005.76,
Shrub & Scrub class	1.304	3.684	0.504	0.01	1.372	9.887	0.003)	p=<0.001
shannon	-0.825	0.438	0.04	<0.001	0.405	0.474		
elevation	-0.464	0.629	0.02	<0.001	0.604	0.654		
log_sl_	0.017	1.017	0.002	<0.001	1.014	1.02		
cos_ta_	-0.723	0.485	0.012	<0.001	0.474	0.497		

<b>Individual 5439 (n = 71400, number of events = 6498)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	-0.925	0.396	0.276	<0.001	0.231	0.681		
Trees class	1.226	3.409	0.262	<0.001	2.041	5.695		
Grass class	2.058	7.832	0.2	<0.001	5.296	11.581		
Flooded vegetation class	2.197	9.001	0.221	<0.001	5.84	13.874		
Crops class	1.885	6.589	0.198	<0.001	4.47	9.712	0.681 (se =	2670.85,
Shrub & Scrub class	-0.143	0.867	1.025	0.889	0.116	6.462	0.004)	p=<0.001
shannon	-0.873	0.418	0.06	<0.001	0.372	0.47		
elevation	-0.398	0.672	0.028	<0.001	0.636	0.709		
log_sl_	0.03	1.031	0.003	<0.001	1.025	1.037		
cos_ta_	-0.586	0.556	0.019	<0.001	0.536	0.577		

**Model 2: landcover + mena\_patch\_area + elevation + log\_sl\_ + cos\_ta\_ + strata(step\_id\_). Data: 2021**

<b>Individual 5181 (n = 68306, number of events = 6623)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.273	1.313	0.351	0.437	0.66	2.613		
Trees class	1.818	6.159	0.376	<0.001	2.946	12.873	0.905 (se =	15928.92,
Grass class	5.246	189.86	0.32	<0.001	101.372	355.595	0.002)	p=<0.001

Flooded vegetation class	4.575	97.01	0.346	<0.001	49.244	191.107		
Crops class	3.715	41.048	0.321	<0.001	21.901	76.936		
Shrub & Scrub class	3.585	36.066	0.51	<0.001	13.284	97.915		
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.008	0.992	0.003	0.002	0.987	0.997		
log_sl_	0.237	1.268	0.009	<0.001	1.245	1.29		
cos_ta_	-1.745	0.175	0.026	<0.001	0.166	0.184		
<b>Individual 5264 (n = 63701, number of events = 6010)</b>								
Covariate	coef	Exp(Coeff)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.136	1.145	0.286	0.636	0.653	2.008		
Trees class	1.522	4.58	0.299	<0.001	2.551	8.223		
Grass class	4.074	58.804	0.245	<0.001	36.366	95.086		
Flooded vegetation class	3.598	36.518	0.27	<0.001	21.526	61.952		
Crops class	3.086	21.879	0.246	<0.001	13.521	35.403	0.772 (se = 0.004)	5751.76, p=<0.001
Shrub & Scrub class	3.331	27.966	0.403	<0.001	12.697	61.599		
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.015	0.985	0.003	<0.001	0.979	0.991		
log_sl_	0.093	1.097	0.004	<0.001	1.088	1.106		
cos_ta_	-0.832	0.435	0.021	<0.001	0.418	0.453		
<b>Individual 5268 (n = 191256, number of events = 17681)</b>								
Covariate	coef	Exp(Coeff)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.599	1.821	0.216	0.006	1.192	2.782		
Trees class	1.972	7.183	0.218	<0.001	4.681	11.021		
Grass class	4.307	74.244	0.191	<0.001	51.088	107.896		
Flooded vegetation class	3.564	35.318	0.207	<0.001	23.561	52.941		
Crops class	3.254	25.882	0.191	<0.001	17.797	37.641	0.761 (se = 0.002)	15304.66, p=<0.001
Shrub & Scrub class	3.423	30.649	0.278	<0.001	17.779	52.835		
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.024	0.976	0.003	<0.001	0.97	0.982		
log_sl_	0.046	1.047	0.002	<0.001	1.044	1.051		
cos_ta_	-0.92	0.399	0.012	<0.001	0.389	0.408		
<b>Individual 5270 (n = 191282, number of events = 17681)</b>								
Covariate	coef	Exp(Coeff)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.602	1.825	0.218	0.006	1.19	2.799		
Trees class	1.977	7.221	0.22	<0.001	4.691	11.117		
Grass class	4.338	76.526	0.193	<0.001	52.45	111.653		
Flooded vegetation class	3.562	35.23	0.208	<0.001	23.417	53.003		
Crops class	3.263	26.118	0.193	<0.001	17.888	38.133	0.761 (se = 0.002)	15371.34, p=<0.001
Shrub & Scrub class	3.45	31.496	0.277	<0.001	18.297	54.217		
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.024	0.976	0.003	<0.001	0.97	0.982		
log_sl_	0.048	1.05	0.002	<0.001	1.046	1.053		
cos_ta_	-0.918	0.399	0.012	<0.001	0.39	0.409		
<b>Individual 5277 (n = 191165, number of events = 17681)</b>								

Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.564	1.758	0.217	0.009	1.15	2.688		
Trees class	1.972	7.183	0.219	<0.001	4.678	11.029		
Grass class	4.29	72.965	0.19	<0.001	50.235	105.98		
Flooded vegetation class	3.532	34.199	0.206	<0.001	22.821	51.249		
Crops class	3.235	25.407	0.191	<0.001	17.478	36.932		
Shrub & Scrub class	3.383	29.457	0.277	<0.001	17.101	50.74	0.761 (se = 0.002)	15299.48, p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.024	0.976	0.003	<0.001	0.971	0.981		
log_sl	0.044	1.045	0.002	<0.001	1.042	1.049		
cos_tan	-0.92	0.398	0.012	<0.001	0.389	0.408		
<b>Individual 5278 (n = 191145, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.581	1.787	0.216	0.007	1.17	2.73		
Trees class	2.001	7.394	0.219	<0.001	4.818	11.347		
Grass class	4.316	74.895	0.19	<0.001	51.569	108.772		
Flooded vegetation class	3.625	37.534	0.206	<0.001	25.052	56.234		
Crops class	3.263	26.135	0.191	<0.001	17.982	37.983		
Shrub & Scrub class	3.591	36.26	0.277	<0.001	21.061	62.428	0.763 (se = 0.002)	15479.93, p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.025	0.975	0.003	<0.001	0.969	0.981		
log_sl	0.046	1.047	0.002	<0.001	1.044	1.051		
cos_tan	-0.933	0.393	0.012	<0.001	0.384	0.403		
<b>Individual 5282 (n = 191295, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.595	1.814	0.218	0.006	1.183	2.782		
Trees class	2.016	7.51	0.219	<0.001	4.886	11.546		
Grass class	4.33	75.941	0.192	<0.001	52.076	110.745		
Flooded vegetation class	3.586	36.075	0.208	<0.001	23.98	54.268		
Crops class	3.276	26.471	0.193	<0.001	18.14	38.629		
Shrub & Scrub class	3.435	31.028	0.277	<0.001	18.026	53.406	0.762 (se = 0.002)	15361.68, p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.029	0.972	0.003	<0.001	0.966	0.978		
log_sl	0.048	1.049	0.002	<0.001	1.045	1.052		
cos_tan	-0.923	0.397	0.012	<0.001	0.388	0.407		
<b>Individual 5284 (n = 191142, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.576	1.779	0.218	0.008	1.161	2.726		
Trees class	2.049	7.76	0.221	<0.001	5.035	11.961		
Grass class	4.353	77.741	0.193	<0.001	53.25	113.495		
Flooded vegetation class	3.626	37.579	0.209	<0.001	24.966	56.566		
Crops class	3.31	27.382	0.193	<0.001	18.743	40.003		
Shrub & Scrub class	3.543	34.57	0.277	<0.001	20.087	59.497	0.763 (se = 0.002)	15415.03, p=<0.001

mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.028	0.972	0.003	<0.001	0.966	0.979		
log_sl	0.048	1.049	0.002	<0.001	1.046	1.053		
cos_ta	-0.924	0.397	0.012	<0.001	0.388	0.406		
<b>Individual 5285 (n = 21070, number of events = 2033)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.233	1.262	0.575	0.685	0.409	3.895		
Trees class	1.481	4.396	0.618	0.017	1.31	14.746		
Grass class	4.752	115.84	0.504	<0.001	43.122	311.221		
Flooded vegetation class	4.398	81.27	0.536	<0.001	28.442	232.215		
Crops class	3.585	36.048	0.505	<0.001	13.404	96.947	0.835 (se =	3092.09,
Shrub & Scrub class	4.058	57.838	0.676	<0.001	15.375	217.572	0.006)	p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.009	0.991	0.004	0.036	0.983	0.999		
log_sl	0.143	1.154	0.01	<0.001	1.132	1.176		
cos_ta	-1.233	0.292	0.039	<0.001	0.27	0.315		
<b>Individual 5286 (n = 191185, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.566	1.761	0.216	0.009	1.152	2.692		
Trees class	1.989	7.307	0.219	<0.001	4.758	11.223		
Grass class	4.311	74.536	0.191	<0.001	51.306	108.283		
Flooded vegetation class	3.558	35.092	0.206	<0.001	23.424	52.572		
Crops class	3.255	25.93	0.191	<0.001	17.837	37.694	0.761 (se =	15342.81,
Shrub & Scrub class	3.407	30.173	0.279	<0.001	17.477	52.093	0.002)	p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.028	0.972	0.003	<0.001	0.966	0.979		
log_sl	0.047	1.048	0.002	<0.001	1.045	1.052		
cos_ta	-0.918	0.399	0.012	<0.001	0.39	0.409		
<b>Individual 5301 (n = 191157, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.598	1.818	0.217	0.006	1.189	2.779		
Trees class	2.022	7.552	0.219	<0.001	4.92	11.591		
Grass class	4.313	74.664	0.191	<0.001	51.375	108.51		
Flooded vegetation class	3.614	37.114	0.207	<0.001	24.754	55.647		
Crops class	3.261	26.087	0.191	<0.001	17.937	37.94	0.760 (se =	15186.84,
Shrub & Scrub class	3.444	31.321	0.278	<0.001	18.153	54.04	0.002)	p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.021	0.979	0.003	<0.001	0.974	0.985		
log_sl	0.046	1.047	0.002	<0.001	1.043	1.05		
cos_ta	-0.917	0.4	0.012	<0.001	0.39	0.409		
<b>Individual 5303 (n = 191044, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.572	1.772	0.218	0.009	1.155	2.718	0.761 (se =	15302.69,
Trees class	2.007	7.44	0.22	<0.001	4.832	11.456	0.002)	p=<0.001

Grass class	4.329	75.837	0.192	<0.001	52.061	110.469		
Flooded vegetation class	3.645	38.292	0.208	<0.001	25.485	57.535		
Crops class	3.272	26.368	0.192	<0.001	18.087	38.44		
Shrub & Scrub class	3.489	32.764	0.281	<0.001	18.881	56.856		
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.022	0.979	0.003	<0.001	0.974	0.983		
log_sl_	0.045	1.046	0.002	<0.001	1.043	1.05		
cos_ta_	-0.919	0.399	0.012	<0.001	0.39	0.408		
<b>Individual 5308 (n = 190976, number of events = 17681)</b>								
Covariate	coef	Exp(Coeff)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.549	1.732	0.217	0.011	1.133	2.647		
Trees class	2.005	7.425	0.219	<0.001	4.832	11.41		
Grass class	4.324	75.475	0.191	<0.001	51.943	109.668		
Flooded vegetation class	3.61	36.948	0.206	<0.001	24.654	55.372		
Crops class	3.278	26.526	0.191	<0.001	18.242	38.572		
Shrub & Scrub class	3.532	34.178	0.277	<0.001	19.852	58.841	0.762 (se = 0.002)	15343.44, p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.026	0.974	0.003	<0.001	0.969	0.98		
log_sl_	0.048	1.05	0.002	<0.001	1.046	1.053		
cos_ta_	-0.911	0.402	0.012	<0.001	0.393	0.412		
<b>Individual 5439 (n = 191150, number of events = 17681)</b>								
Covariate	coef	Exp(Coeff)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.643	1.902	0.218	0.003	1.241	2.915		
Trees class	2.044	7.723	0.221	<0.001	5.01	11.905		
Grass class	4.361	78.31	0.193	<0.001	53.687	114.225		
Flooded vegetation class	3.651	38.514	0.208	<0.001	25.607	57.927		
Crops class	3.308	27.336	0.193	<0.001	18.726	39.905		
Shrub & Scrub class	3.556	35.015	0.277	<0.001	20.333	60.299	0.761 (se = 0.002)	15237.07, p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.025	0.975	0.003	<0.001	0.969	0.981		
log_sl_	0.046	1.047	0.002	<0.001	1.043	1.05		
cos_ta_	-0.917	0.4	0.012	<0.001	0.39	0.409		
<b>Model 2: landcover + mena_patch_area + elevation + log_sl_ + cos_ta_ + strata(step_id_). Data: 2022</b>								
<b>Individual 5181 (n = 27690, number of events = 2527)</b>								
Covariate	coef	Exp(Coeff)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	-1.135	0.321	0.561	0.043	0.107	0.965		
Trees class	2.706	14.972	0.448	<0.001	6.228	35.994		
Grass class	2.689	14.713	0.417	<0.001	6.502	33.291		
Flooded vegetation class	2.788	16.247	0.457	<0.001	6.634	39.787	0.755 (se = 0.006)	2090.16, p=<0.001
Crops class	2.714	15.089	0.414	<0.001	6.701	33.975		
Shrub & Scrub class	12.027	0	912.489	0.989	0	Inf		

mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.401	0.67	0.04	<0.001	0.619	0.724		
log_sl_	0.053	1.054	0.006	<0.001	1.041	1.067		
cos_ta_	-1.016	0.362	0.032	<0.001	0.34	0.385		
<b>Individual 5264 (n = 177942, number of events = 16186)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
Water class	0.783	2.188	0.228	<0.001	1.401	3.417		
Trees class	2.116	8.301	0.23	<0.001	5.288	13.03		
Grass class	2.857	17.413	0.203	<0.001	11.691	25.934		
Flooded vegetation class	2.684	14.64	0.216	<0.001	9.59	22.35		
Crops class	2.592	13.36	0.203	<0.001	8.982	19.872	0.683 (se =	6700.12,
Shrub & Scrub class	1.242	3.461	0.502	0.013	1.293	9.263	0.003)	p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.467	0.627	0.02	<0.001	0.602	0.653		
log_sl_	0.017	1.017	0.002	<0.001	1.014	1.02		
cos_ta_	-0.718	0.488	0.012	<0.001	0.476	0.499		
<b>Individual 5285 (n = 177962, number of events = 16186)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
Water class	0.812	2.253	0.227	<0.001	1.443	3.518		
Trees class	2.098	8.146	0.23	<0.001	5.195	12.775		
Grass class	2.903	18.22	0.203	<0.001	12.235	27.134		
Flooded vegetation class	2.686	14.674	0.216	<0.001	9.615	22.397		
Crops class	2.633	13.922	0.203	<0.001	9.359	20.709	0.684 (se =	6625.96,
Shrub & Scrub class	1.265	3.541	0.523	0.016	1.269	9.88	0.003)	p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.458	0.633	0.02	<0.001	0.608	0.658		
log_sl_	0.016	1.016	0.002	<0.001	1.013	1.019		
cos_ta_	-0.707	0.493	0.012	<0.001	0.482	0.505		
<b>Individual 5301 (n = 177949, number of events = 16186)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
Water class	0.833	2.3	0.228	<0.001	1.472	3.594		
Trees class	2.104	8.201	0.23	<0.001	5.228	12.863		
Grass class	2.919	18.519	0.204	<0.001	12.421	27.611		
Flooded vegetation class	2.712	15.064	0.216	<0.001	9.859	23.018		
Crops class	2.644	14.064	0.203	<0.001	9.445	20.941	0.685 (se =	6688.10,
Shrub & Scrub class	1.327	3.768	0.512	0.01	1.381	10.278	0.003)	p=<0.001
mean_patch_area	0	1	0	<0.001	1	1		
elevation	-0.451	0.637	0.02	<0.001	0.612	0.663		
log_sl_	0.017	1.017	0.002	<0.001	1.014	1.02		
cos_ta_	-0.711	0.491	0.012	<0.001	0.48	0.503		
<b>Individual 5303 (n = 177932, number of events = 16186)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
Water class	0.795	2.214	0.228	<0.001	1.416	3.46		
Trees class	2.085	8.044	0.23	<0.001	5.12	12.637		
Grass class	2.899	18.154	0.204	<0.001	12.179	27.061	0.687 (se =	6866.09,
Flooded vegetation class	2.697	14.833	0.216	<0.001	9.71	22.659	0.003)	p=<0.001
Crops class	2.614	13.658	0.203	<0.001	9.174	20.336		

<b>Shrub &amp; Scrub class</b>	1.271	3.564	0.503	0.011	1.33	9.545		
<b>mean_patch_area</b>	0	1	0	<0.001	1	1		
<b>elevation</b>	-0.472	0.624	0.02	<0.001	0.599	0.649		
<b>log_sl_</b>	0.016	1.016	0.002	<0.001	1.013	1.019		
<b>cos_ta_</b>	-0.723	0.485	0.012	<0.001	0.474	0.497		
<b>Individual 5439 (n = 71400, number of events = 6498)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	-0.814	0.443	0.276	0.003	0.258	0.761		
<b>Trees class</b>	1.289	3.629	0.261	<0.001	2.176	6.053		
<b>Grass class</b>	2.112	8.267	0.199	<0.001	5.597	12.211		
<b>Flooded vegetation class</b>	2.25	9.484	0.221	<0.001	6.153	14.619		
<b>Crops class</b>	1.971	7.176	0.197	<0.001	4.874	10.565	0.677 (se =	2597.12,
<b>Shrub &amp; Scrub class</b>	-0.118	0.889	1.025	0.908	0.119	6.625	0.004)	p=<0.001
<b>mean_patch_area</b>	0	1	0	<0.001	1	1		
<b>elevation</b>	-0.407	0.666	0.028	<0.001	0.631	0.703		
<b>log_sl_</b>	0.029	1.029	0.003	<0.001	1.023	1.035		
<b>cos_ta_</b>	-0.585	0.557	0.019	<0.001	0.537	0.578		
<b>Model 3: landcover + contagio + elevation + log_sl_ + cos_ta_ + strata(step_id_). Data: 2021</b>								
<b>Individual 5181 (n = 68306, number of events = 6623)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.292	1.339	0.351	0.405	0.673	2.664		
<b>Trees class</b>	1.783	5.95	0.377	<0.001	2.845	12.446		
<b>Grass class</b>	5.234	187.497	0.32	<0.001	100.095	351.216		
<b>Flooded vegetation class</b>	4.806	122.229	0.347	<0.001	61.879	241.436		
<b>Crops class</b>	3.705	40.664	0.321	<0.001	21.691	76.232	0.905 (se =	15952.47,
<b>Shrub &amp; Scrub class</b>	3.713	40.984	0.512	<0.001	15.029	111.76	0.002)	p=<0.001
<b>contagio</b>	3.267	26.233	0.173	<0.001	18.676	36.848		
<b>elevation</b>	-0.007	0.993	0.003	0.003	0.988	0.997		
<b>log_sl_</b>	0.236	1.266	0.009	<0.001	1.243	1.289		
<b>cos_ta_</b>	-1.74	0.176	0.026	<0.001	0.167	0.185		
<b>Individual 5264 (n = 63701, number of events = 6010)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.131	1.14	0.286	0.648	0.65	1.999		
<b>Trees class</b>	1.51	4.526	0.299	<0.001	2.52	8.127		
<b>Grass class</b>	4.061	58.009	0.245	<0.001	35.862	93.834		
<b>Flooded vegetation class</b>	3.809	45.102	0.271	<0.001	26.501	76.76		
<b>Crops class</b>	3.077	21.703	0.246	<0.001	13.407	35.132	0.772 (se =	5803.50,
<b>Shrub &amp; Scrub class</b>	3.481	32.486	0.404	<0.001	14.708	71.751	0.004)	p=<0.001
<b>contagio</b>	2.214	9.151	0.152	<0.001	6.788	12.336		
<b>elevation</b>	-0.015	0.985	0.003	<0.001	0.979	0.991		
<b>log_sl_</b>	0.094	1.098	0.004	<0.001	1.089	1.107		
<b>cos_ta_</b>	0.831	0.435	0.021	<0.001	0.418	0.454		
<b>Individual 5268 (n = 191256, number of events = 17681)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.582	1.789	0.216	0.007	1.171	2.734		

Trees class	1.983	7.266	0.219	<0.001	4.734	11.151	0.764 (se = 0.002)	15653.91, p=<0.001
Grass class	4.3	73.675	0.191	<0.001	50.69	107.081		
Flooded vegetation class	3.926	50.688	0.208	<0.001	33.734	76.161		
Crops class	3.242	25.595	0.191	<0.001	17.597	37.227		
Shrub & Scrub class	3.663	38.986	0.279	<0.001	22.566	67.354		
contagio	2.678	14.558	0.104	<0.001	11.871	17.853		
elevation	-0.023	0.977	0.003	<0.001	0.971	0.983		
log_sl_	0.048	1.05	0.002	<0.001	1.046	1.053		
cos_ta_	-0.92	0.399	0.012	<0.001	0.389	0.408		
<b>Individual 5270 (n = 191282, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.582	1.789	0.218	0.008	1.168	2.742		
Trees class	1.992	7.333	0.22	<0.001	4.765	11.286		
Grass class	4.325	75.595	0.192	<0.001	51.842	110.231		
Flooded vegetation class	3.925	50.642	0.209	<0.001	33.605	76.317		
Crops class	3.247	25.721	0.193	<0.001	17.627	37.533	0.765 (se = 0.002)	15718.30, p=<0.001
Shrub & Scrub class	3.759	42.908	0.278	<0.001	24.885	73.984		
contagio	2.693	14.781	0.104	<0.001	12.047	18.135		
elevation	-0.023	0.977	0.003	<0.001	0.971	0.983		
log_sl_	0.05	1.052	0.002	<0.001	1.048	1.055		
cos_ta_	-0.916	0.4	0.012	<0.001	0.391	0.409		
<b>Individual 5277 (n = 191165, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.548	1.729	0.216	0.011	1.131	2.643		
Trees class	1.974	7.2	0.219	<0.001	4.688	11.057		
Grass class	4.273	71.769	0.19	<0.001	49.412	104.24		
Flooded vegetation class	3.871	47.984	0.207	<0.001	31.95	72.064		
Crops class	3.22	25.022	0.191	<0.001	17.214	36.371	0.765 (se = 0.002)	15622.22, p=<0.001
Shrub & Scrub class	3.626	37.569	0.278	<0.001	21.806	64.726		
contagio	2.699	14.867	0.104	<0.001	12.129	18.223		
elevation	-0.024	0.977	0.003	<0.001	0.972	0.982		
log_sl_	0.046	1.047	0.002	<0.001	1.044	1.051		
cos_ta_	-0.92	0.398	0.012	<0.001	0.389	0.408		
<b>Individual 5278 (n = 191145, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.563	1.756	0.216	0.009	1.15	2.683		
Trees class	2.017	7.514	0.219	<0.001	4.895	11.534		
Grass class	4.302	73.85	0.19	<0.001	50.843	107.268		
Flooded vegetation class	3.973	53.118	0.207	<0.001	35.373	79.765		
Crops class	3.246	25.7	0.191	<0.001	17.681	37.356	0.766 (se = 0.002)	15798.76, p=<0.001
Shrub & Scrub class	3.867	47.805	0.278	<0.001	27.732	82.406		
contagio	2.71	15.026	0.104	<0.001	12.256	18.423		
elevation	-0.024	0.976	0.003	<0.001	0.97	0.982		
log_sl_	0.048	1.049	0.002	<0.001	1.046	1.053		
cos_ta_	-0.932	0.394	0.012	<0.001	0.385	0.403		
<b>Individual 5282 (n = 191295, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test

Water class	0.571	1.77	0.218	0.009	1.155	2.712		
Trees class	2.024	7.565	0.219	<0.001	4.925	11.619		
Grass class	4.31	74.435	0.192	<0.001	51.09	108.446		
Flooded vegetation class	3.922	50.51	0.209	<0.001	33.536	76.073		
Crops class	3.255	25.913	0.192	<0.001	17.774	37.779	0.765 (se =	15692.76,
Shrub & Scrub class	3.712	40.956	0.277	<0.001	23.792	70.502	0.002)	p=<0.001
contagio	2.698	14.843	0.104	<0.001	12.099	18.209		
elevation	-0.027	0.973	0.003	<0.001	0.967	0.979		
log_sl_	0.049	1.051	0.002	<0.001	1.047	1.054		
cos_ta_	-0.923	0.397	0.012	<0.001	0.388	0.407		
<b>Individual 5284 (n = 191142, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.561	1.753	0.218	0.01	1.144	2.687		
Trees class	2.055	7.807	0.221	<0.001	5.061	12.044		
Grass class	4.342	76.836	0.193	<0.001	52.588	112.265		
Flooded vegetation class	3.958	52.339	0.21	<0.001	34.675	79.001		
Crops class	3.296	27.014	0.194	<0.001	18.477	39.497	0.766 (se =	15727.01,
Shrub & Scrub class	3.815	45.373	0.278	<0.001	26.327	78.195	0.002)	p=<0.001
contagio	2.619	13.718	0.104	<0.001	11.193	16.814		
elevation	-0.026	0.974	0.003	<0.001	0.968	0.98		
log_sl_	0.05	1.051	0.002	<0.001	1.048	1.055		
cos_ta_	-0.923	0.397	0.012	<0.001	0.388	0.407		
<b>Individual 5285 (n = 21070, number of events = 2033)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.252	1.286	0.575	0.662	0.417	3.968		
Trees class	1.458	4.299	0.618	0.018	1.281	14.421		
Grass class	4.743	114.819	0.504	<0.001	42.747	308.405		
Flooded vegetation class	4.538	93.494	0.538	<0.001	32.567	268.402		
Crops class	3.583	35.965	0.505	<0.001	13.374	96.714	0.835 (se =	3090.61,
Shrub & Scrub class	4.142	62.901	0.675	<0.001	16.756	236.129	0.006)	p=<0.001
contagio	1.925	6.853	0.259	<0.001	4.121	11.396		
elevation	-0.009	0.991	0.004	0.04	0.983	1		
log_sl_	0.143	1.154	0.01	<0.001	1.131	1.176		
cos_ta_	-1.232	0.292	0.039	<0.001	0.27	0.315		
<b>Individual 5286 (n = 191185, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.551	1.735	0.216	0.011	1.135	2.651		
Trees class	2.004	7.417	0.219	<0.001	4.829	11.391		
Grass class	4.301	73.764	0.191	<0.001	50.777	107.157		
Flooded vegetation class	3.905	49.627	0.207	<0.001	33.057	74.504		
Crops class	3.242	25.58	0.191	<0.001	17.597	37.184	0.765 (se =	15682.63,
Shrub & Scrub class	3.7	40.433	0.279	<0.001	23.388	69.9	0.002)	p=<0.001
contagio	2.686	14.674	0.104	<0.001	11.972	17.985		
elevation	-0.027	0.973	0.003	<0.001	0.967	0.98		
log_sl_	0.049	1.05	0.002	<0.001	1.047	1.054		
cos_ta_	-0.918	0.399	0.012	<0.001	0.39	0.409		
<b>Individual 5301 (n = 191157, number of events = 17681)</b>								

Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.582	1.79	0.217	0.007	1.171	2.736		
Trees class	2.04	7.694	0.219	<0.001	5.013	11.808		
Grass class	4.298	73.52	0.191	<0.001	50.59	106.841		
Flooded vegetation class	3.956	52.272	0.208	<0.001	34.79	78.538		
Crops class	3.243	25.62	0.191	<0.001	17.617	37.26	0.763 (se =	15505.90,
Shrub & Scrub class	3.735	41.906	0.279	<0.001	24.239	72.45	0.002)	p=<0.001
contagio	2.679	14.565	0.104	<0.001	11.878	17.86		
elevation	-0.02	0.98	0.003	<0.001	0.974	0.985		
log_sl_	0.048	1.049	0.002	<0.001	1.045	1.052		
cos_ta_	-0.917	0.4	0.012	<0.001	0.391	0.409		
<b>Individual 5303 (n = 191044, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.556	1.743	0.218	0.011	1.137	2.673		
Trees class	2.019	7.529	0.22	<0.001	4.891	11.591		
Grass class	4.314	74.721	0.192	<0.001	51.304	108.826		
Flooded vegetation class	4	54.596	0.209	<0.001	36.262	82.199		
Crops class	3.254	25.899	0.192	<0.001	17.768	37.751	0.764 (se =	15643.71,
Shrub & Scrub class	3.735	41.874	0.282	<0.001	24.091	72.786	0.002)	p=<0.001
contagio	2.733	15.384	0.104	<0.001	12.552	18.856		
elevation	-0.021	0.98	0.003	<0.001	0.975	0.985		
log_sl_	0.047	1.048	0.002	<0.001	1.045	1.052		
cos_ta_	-0.919	0.399	0.012	<0.001	0.39	0.408		
<b>Individual 5308 (n = 190976, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.543	1.721	0.217	0.012	1.126	2.631		
Trees class	2.027	7.593	0.219	<0.001	4.939	11.672		
Grass class	4.318	75.05	0.191	<0.001	51.629	109.096		
Flooded vegetation class	3.955	52.179	0.208	<0.001	34.73	78.394		
Crops class	3.27	26.309	0.191	<0.001	18.085	38.272	0.765 (se =	15672.83,
Shrub & Scrub class	3.83	46.04	0.278	<0.001	26.695	79.402	0.002)	p=<0.001
contagio	2.782	16.145	0.104	<0.001	13.162	19.803		
elevation	-0.025	0.975	0.003	<0.001	0.97	0.981		
log_sl_	0.05	1.052	0.002	<0.001	1.048	1.055		
cos_ta_	-0.911	0.402	0.012	<0.001	0.393	0.412		
<b>Individual 5439 (n = 191150, number of events = 17681)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.625	1.868	0.218	0.004	1.22	2.861		
Trees class	2.042	7.705	0.22	<0.001	5.002	11.867		
Grass class	4.343	76.972	0.192	<0.001	52.815	112.177		
Flooded vegetation class	3.981	53.556	0.209	<0.001	35.563	80.654		
Crops class	3.287	26.759	0.193	<0.001	18.346	39.028	0.764 (se =	15552.40,
Shrub & Scrub class	3.828	45.95	0.278	<0.001	26.648	79.234	0.002)	p=<0.001
contagio	2.71	15.025	0.104	<0.001	12.256	18.419		
elevation	-0.024	0.976	0.003	<0.001	0.97	0.982		
log_sl_	0.048	1.049	0.002	<0.001	1.045	1.053		
cos_ta_	-0.917	0.4	0.012	<0.001	0.39	0.409		

<b>Model 3: landcover + contagio + elevation + log_sl_ + cos_ta_ + strata(step_id_). Data: 2022</b>								
<b>Individual 5181 (n = 27690, number of events = 2527)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	-1.107	0.33	0.561	0.048	0.11	0.992		
Trees class	2.808	16.571	0.447	<0.001	6.895	39.827		
Grass class	2.694	14.795	0.417	<0.001	6.536	33.49		
Flooded vegetation class	3.056	21.249	0.46	<0.001	8.619	52.386		
Crops class	2.727	15.28	0.414	<0.001	6.784	34.417	0.755 (se =	2089.41,
Shrub & Scrub class	11.757	0	889.065	0.989	0	Inf	0.006)	p=<0.001
contagio	3.099	22.175	0.294	<0.001	12.46	39.465		
elevation	-0.387	0.679	0.04	<0.001	0.628	0.734		
log_sl_	0.052	1.054	0.006	<0.001	1.041	1.066		
cos_ta_	-1.013	0.363	0.032	<0.001	0.341	0.387		
<b>Individual 5264 (n = 177942, number of events = 16186)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.803	2.232	0.228	<0.001	1.429	3.486		
Trees class	2.093	8.109	0.23	<0.001	5.162	12.738		
Grass class	2.850	17.293	0.203	<0.001	11.609	25.76		
Flooded vegetation class	2.805	16.522	0.217	<0.001	10.808	25.255		
Crops class	2.598	13.432	0.203	<0.001	9.030	19.981	0.683 (se =	6653.32,
Shrub & Scrub class	1.283	3.608	0.502	0.011	1.347	9.659	0.003)	p=<0.001
contagio	1.608	4.99	0.103	<0.001	4.076	6.11		
elevation	-0.457	0.633	0.02	<0.001	0.609	0.659		
log_sl_	0.017	1.017	0.002	<0.001	1.014	1.02		
cos_ta_	-0.717	0.488	0.012	<0.001	0.477	0.5		
<b>Individual 5285 (n = 177962, number of events = 16186)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.824	2.28	0.227	<0.001	1.46	3.56		
Trees class	2.08	8.006	0.23	<0.001	5.101	12.564		
Grass class	2.891	18.01	0.203	<0.001	12.09	26.829		
Flooded vegetation class	2.826	16.87	0.217	<0.001	11.036	25.789		
Crops class	2.632	13.896	0.203	<0.001	9.339	20.675	0.683 (se =	6617.76,
Shrub & Scrub class	1.293	3.643	0.528	0.014	1.295	10.247	0.003)	p=<0.001
contagio	1.643	5.17	0.103	<0.001	4.224	6.329		
elevation	-0.45	0.638	0.02	<0.001	0.613	0.663		
log_sl_	0.016	1.016	0.002	<0.001	1.013	1.019		
cos_ta_	-0.707	0.493	0.012	<0.001	0.482	0.505		
<b>Individual 5301 (n = 177949, number of events = 16186)</b>								
Covariate	coef	Exp(Coef)	se(coef)	Pr(> z )	lower_95	upper_95	Concordance	Likelihood ratio test
Water class	0.847	2.333	0.228	<0.001	1.492	3.648		
Trees class	2.081	8.014	0.23	<0.001	5.105	12.583		
Grass class	2.909	18.336	0.204	<0.001	12.293	27.349		
Flooded vegetation class	2.841	17.124	0.217	<0.001	11.189	26.208	0.683 (se =	6656.58,
Crops class	2.646	14.1	0.203	<0.001	9.465	21.003	0.003)	p=<0.001
Shrub & Scrub class	1.386	3.999	0.509	0.007	1.473	10.853		

<b>contagio</b>	1.642	5.168	0.103	<0.001	4.22	6.328		
<b>elevation</b>	-0.443	0.642	0.02	<0.001	0.617	0.668		
<b>log_sl_</b>	0.017	1.017	0.002	<0.001	1.014	1.02		
<b>cos_ta_</b>	-0.71	0.492	0.012	<0.001	0.48	0.503		
<b>Individual 5303 (n = 177932, number of events = 16186)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	0.807	2.241	0.228	<0.001	1.433	3.503		
<b>Trees class</b>	2.064	7.881	0.231	<0.001	5.01	12.396		
<b>Grass class</b>	2.886	17.928	0.204	<0.001	12.022	26.737		
<b>Flooded vegetation class</b>	2.835	17.031	0.217	<0.001	11.131	26.059		
<b>Crops class</b>	2.614	13.652	0.203	<0.001	9.166	20.335	0.686 (se =	6851.91,
<b>Shrub &amp; Scrub class</b>	1.343	3.829	0.502	0.007	1.431	10.243	0.003)	p=<0.001
<b>contagio</b>	1.686	5.397	0.103	<0.001	4.411	6.604		
<b>elevation</b>	-0.464	0.629	0.02	<0.001	0.605	0.654		
<b>log_sl_</b>	0.016	1.016	0.002	<0.001	1.013	1.019		
<b>cos_ta_</b>	-0.722	0.486	0.012	<0.001	0.474	0.497		
<b>Individual 5439 (n = 71400, number of events = 6498)</b>								
<b>Covariate</b>	<b>coef</b>	<b>Exp(Coef)</b>	<b>se(coef)</b>	<b>Pr(&gt; z )</b>	<b>lower_95</b>	<b>upper_95</b>	<b>Concordance</b>	<b>Likelihood ratio test</b>
<b>Water class</b>	-0.796	0.451	0.276	0.004	0.263	0.775		
<b>Trees class</b>	1.265	3.544	0.261	<0.001	2.123	5.916		
<b>Grass class</b>	2.105	8.207	0.199	<0.001	5.554	12.126		
<b>Flooded vegetation class</b>	2.373	10.729	0.222	<0.001	6.939	16.588		
<b>Crops class</b>	1.975	7.21	0.198	<0.001	4.895	10.618	0.676 (se =	2578.00,
<b>Shrub &amp; Scrub class</b>	-0.042	0.959	1.025	0.967	0.129	7.148	0.004)	p=<0.001
<b>contagio</b>	1.741	5.703	0.157	<0.001	4.195	7.754		
<b>elevation</b>	-0.402	0.669	0.028	<0.001	0.634	0.706		
<b>log_sl_</b>	0.029	1.029	0.003	<0.001	1.023	1.035		
<b>cos_ta_</b>	-0.585	0.557	0.019	<0.001	0.537	0.578		