Field of studies: Geoinformation

Album ID: 455828

Tomasz Matuszek

Measuring an impact of the Landsat 8 thermal band on the supervised land cover classification results

Ocena wpływu zastosowania kanału termalnego Landsat na wyniki nadzorowanej klasyfikacji pokrycia terenu

Engineer's thesis written

in the Institute of Geoecology and Geoinformation

under the supervision of

dr. hab. Jakub Nowosad

Poznań, 2023

Abstract

Abstrakt

Streszczenie powinno przedstawiać skrótowo główny problem pracy i jego rozwiązanie. Możliwa struktura streszczenia to: (1) 1-3 zdania wstępu do problemu (czym się zajmujemy, dlaczego jest to ważne, jakie są problemy/luki do wypełnienia), (2) 1 zdanie opisujące cel pracy, (3) 1-3 zdania przedstawiające użyte materiały (dane) i metody (techniki, narzędzia), (4) 1-3 zdania obrazujące główne wyniki pracy, (5) 1-2 zdania podsumowujące; możliwe jest też określenie dalszych kroków/planów.

Słowa kluczowe: (4-6 słów/zwrotów opisujących treść pracy, które nie wystąpiły w tytule)

Abstract

The abstract must be consistent with the above text.

Keywords: (as stated before)

Contents

Chapter 1

Introduction

- applications and relevance of land cover maps

- machine learning and supervised classification of satellite images as a tool for creating land cover maps

- pointing out that thermal band if often omitted in land cover classification models, exact impact of thermal factor isn't fully clear

- goal of the thesis is to create land cover map of Poznań metropolitan area and measure the impact of thermal band on the model results

---

Wprowadzenie powinno mieć charakter opisu od ogółu do szczegółu (np. trzy-pięć paragrafów). Pierwszy paragraf powinien być najbardziej ogólny, a kolejne powinny przybliżać czytelnika do problemu. Przedostatni paragraf powinien określić jaki jest problem (są problemy), który praca ma rozwiązać i dlaczego jest to (są one) ważne.

Wprowadzenie powinno być zakończone stwierdzeniem celu pracy. Dodatkowo tutaj może znaleźć się również krótki opis co zostało zrealizowane w pracy.

Pisząc ten rozdział proszę pomyśleć o osobach, które zupełnie nie znają opisywanej tematyki. Należy tutaj krok po kroku wyjaśnić podstawowe koncepcje, istotność problemu,

wyniki poprzednich podobnych badań, itd. Ten rozdział obejmuje tylko kwestie, które już zostały wykonane przez inne osoby - nowe wyniki mają swoje miejsce w rozdziale **?@sec-wyniki**.

Każda kwestia opisana w tym rozdziale powinna być cytowana. Dodatnie cytowania odbywa się poprzez uzupełnienie pliku `thesis.bib` zapisem w formacie BibTeX, a następnie dodanie nazwy referencji poprzedzonej znakiem @. Przykładowo, zacytowanie książki Geocomputation with R odbywa się poprzez (Lovelace et al., 2019).

W przypadku, gdy cytowanie zostało poprawnie wpisane oraz istnieje w pliku `thesis.bib` to bibliografia powinna się automatycznie wygenerować na końcu pracy.

W przypadku, gdy praca dyplomowa opisuje konkretny obszar to można po tym rozdziale stworzyć kolejny rozdział opisujący "obszar badań".

Ten i kolejne rozdziału moją mieć także podrozdziały. Tworzenie podrozdziałów polega na stworzeniu nowej linii rozpoczynającej się od znaków ## a następnie tytułu podrozdziału. Dodatkowo w postaci {#sec-} można dodać skrót nazwy rozdziału/podrozdziału umożliwiający odnoszenie się do niego używając operatora [-@sec].

Chapter 2

Materials and methods

Workflow of the study consisted of several stages: preprocessing of source data (described in Sections 2.1 and 2.2), creating training dataset, model's parameters tuning, land cover map prediction, model quality assessment and evaluating the impact of thermal band on the model's results (Figure 2.5). Each of these steps was performed using R programming language (R Core Team, 2021). Final visualisations were created in QGIS software (QGIS Development Team, 2009). Both programming environment and GIS software used in this process are open-source.

Landsat ARD dataset, provided by GLAD laboratory at the Univeristy of Maryland, was used as a source of multi-spectral satellite imagery. Training points were obtained from LUCAS dataset created by Eurostat (d'Andrimont et al., 2020). Both data sets were downloaded for central-western part of Poland which was chosen as training area (Figure 2.2). This data was pre-processed and then used to train the model and validate its performance.
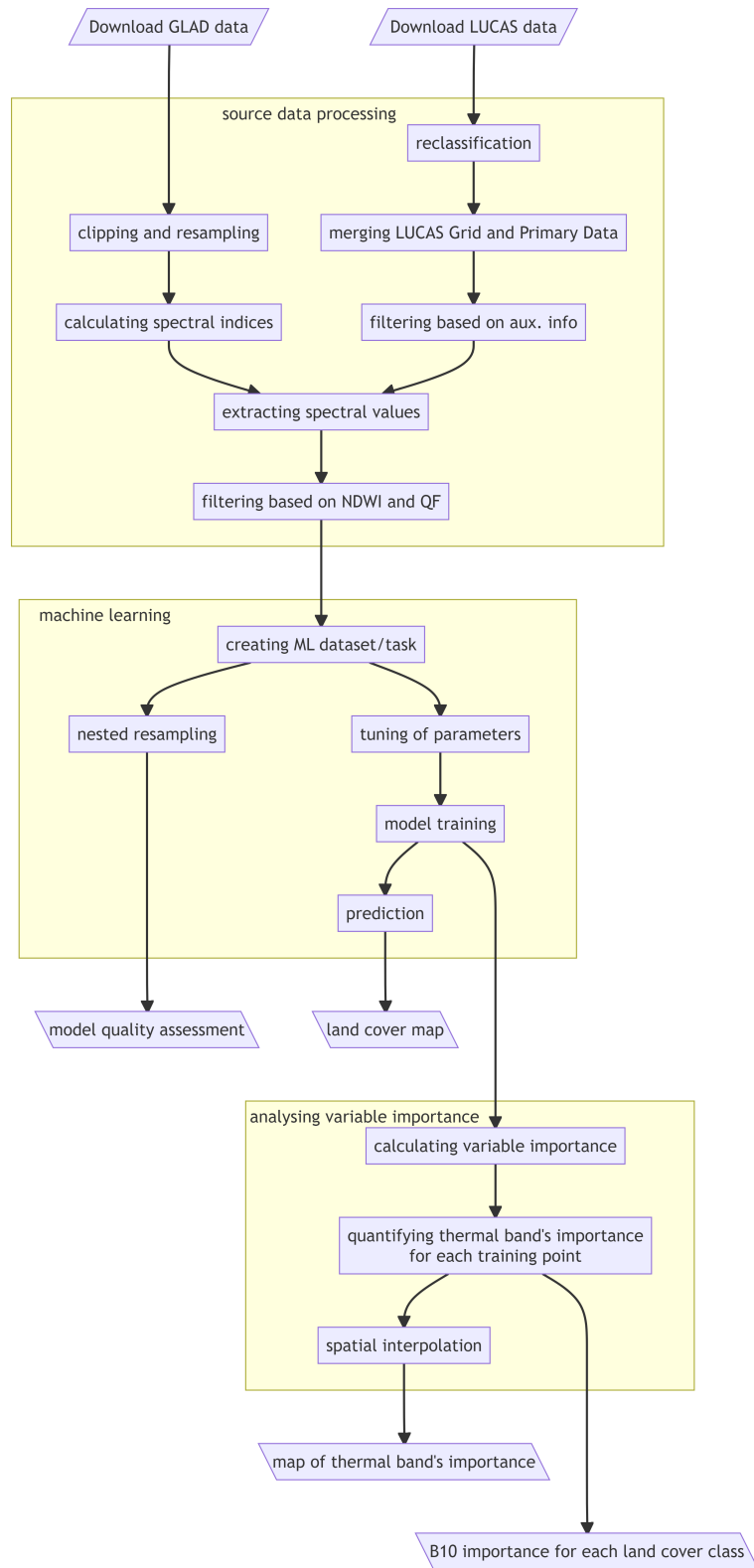
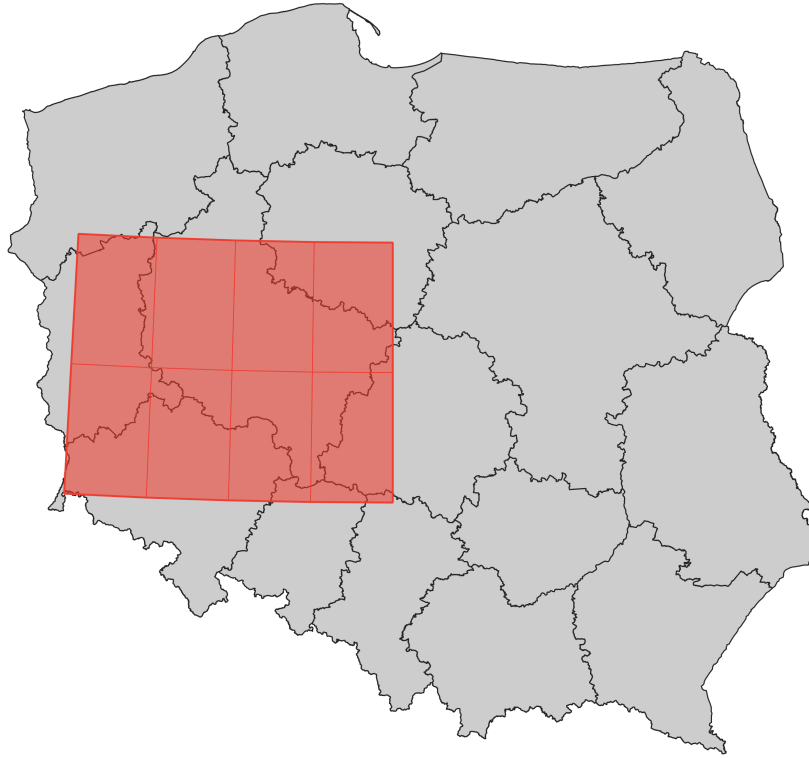**Figure 2.1:** *General workflow of the study*

Figure 2.2: *Training area*

2.1 Satellite imagery

Satellite imagery from GLAD Landsat ARD is available in 16-day interval composites and is divided into 1° x 1° tiles. Processing of original Landsat images performed by GLAD team included converting spectral bands to top-of-atmosphere (TOA) reflectance, converting thermal band to brightness temperature (BT) in Kelvins, scaling the values of all bands as well as adding quality flag for every pixel (Potapov et al., 2020).

Satellite images for eight 1° x 1° tiles, covering the study area (Figure 2.2), were downloaded using GLAD Tools v1.1 and PERL programming language. These images are from 10th interval of the year 2018, so downloaded mosaics consist of images created between 24.05.2018 and 8.06.2018. All downloaded images were merged and reprojected from WGS84 coordinate reference system (EPSG:4326) to UTM zone 33N (EPSG:32633). Every band was also resampled from 0.00025° resolution (corresponding to 27.83 m on equator) to 30 meters resolution.

Table 2.1: *Formulas of spectral indices dervied from Landsat data*

band/index	abbreviation	formula
Blue	B2	-
Green	B3	-
Red	B4	-
Near Infrared	B5 (NIR)	-
Short-wave Infrared 1	B6 (SWIR1)	-
Short-wave Infrared 2	B7 (SWIR2)	-
Thermal	B10 (TIRS1)	-
Normalized Difference Vegetation Index	NDVI	(B5 -B4) / (B4 + B5)
Modified Normalized Difference Water Index	MNDWI	(B3 - B6) / (B3 + B6)
Normalized Difference Moisture Index	NDMI	(B5 - B6) / (B5 + B6)
Modified Bare Surface Index	MBI	(B6 - B7 - B5) / (B6 + B7 + B5) + 0.5

In addition, four spectral indices were derived: Normalized Difference Vegetation Index (NDVI), Modified Normalized Difference Water Index (MNDWI), Normalized Difference Moisture Index (NDMI) and Modified Bare soil Index (MBI). Formulas used to calculate these indices can be found in Table 2.1.

2.2 Land cover data

Data collected during LUCAS survey performed by Eurostat was chosen as land cover training set. At the moment of writing, it is the most accurate and comprehensive dataset containing information about land use and land cover (Pflugmacher et al., 2019) due to the fact, that every point was either manually photo-interpreted or assessed during *in-situ* visit.

LUCAS survey consists of two phases. First phase is based on grid of points with 2km spacing covering whole territory of the European Union (which equals to more than 1 million points). Each point of the grid is visually interpreted using ortho-photos or satellite images, and classified into one of seven major land-cover classes. These classes are: arable land, permanent crops, grassland, wooded areas/shrub land, bare land, artificial land and water. In the second phase, a subsample of grid points is selected

Table 2.2: *Filters applied to reclassified land cover groups. IMP - impervious surface, HRB - herbaceous plants cover, TC - tree cover*

ID	LC class	LUCAS Grid	LUCAS Primary Data	Filters
1	arable land	-	B00 (Cropland)	<30% IMP
2	grasslands	-	E00 (Grassland), D00 (Shrubland)	>50% HRB; <30% IMP
3	forests	-	C00 (Woodland)	>50% TC; <20% IMP
4	bare land	6 (Bare surface)	F00 (Bare land)	-
5	artificial land	7 (Artificial areas)	A00 (Artificial land)	>70% IMP
6	water bodies	8 (Inland water)	G00 (Water areas)	-
7	wetlands	-	H00 (Wetlands)	-

and then visited by Eurostat surveyors. They classify each point according to full LUCAS land cover and land use classification. The survey takes place in the spring and summer in order to observe chosen places in high vegetation season (d'Andrimont et al., 2020).

Surveyor not only assign land cover and land use classes to points, but they also add auxillary information such as plant species present at the site, percentage of land coverage for a chosen class, height of the trees and their maturity, as well as information about water management and irrigation. If there are more than one land cover/land use types at the point, observer can also assign a secondary class for every LUCAS point.

Majority of the training points used for the classification model were points from the second phase of LUCAS survey, also called LUCAS Primary Data. I downloaded a total of 4,153 points for the study area. Pre-processing step included omitting records with missing data, excluding artificial linear land cover classes (e.g. roads or railways) and excluding points that were surveyed more than 500 meters from their theoretical location. In the next step, detailed land cover classes were aggregated into eight main groups of land cover types. Two of them - grassland and shrubland were additionally aggregated into one land cover class. Then, I filtered some of the classes according to the percentage of land coverage or percentage of impervious surface coverage (Table 2.2).

For the least frequent classes in the LUCAS Primary Data dataset - bare land, artificial land and water bodies - I also added points classified during the first phase of LUCAS
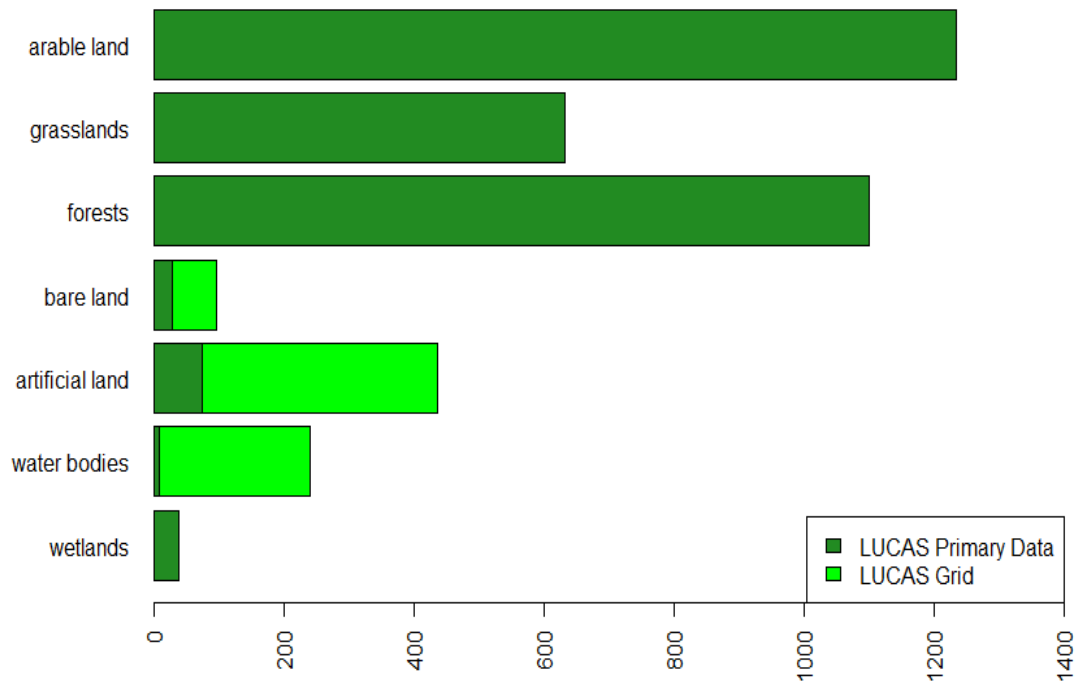
Figure 2.3: *Distribution of points by land cover class after pre-processing*

survey (Figure 2.3). This step was necessary to ensure that every land cover class is represented by enough number of points. It was not possible only for wetlands class, because of lack of such category in the first phase classification. At the end of the pre-processing, dataset was left with 3,778 training points (Buck et al., 2015).

After extracting values from Landsat ARD raster, LUCAS points were also filtered using quality flag provided. Only points with clear-sky quality flag were taken into account during the process of model training. Moreover, water bodies points in which NDWI was lower than 0 were also excluded. These two conditions eliminated over 400 points in total.

Training set obtained after pre-processing can be seen in Figure 2.4. Spatial distribution of data points was fairly even and due to the structure of LUCAS data set, every point was located at least 2 kilometers from another one.
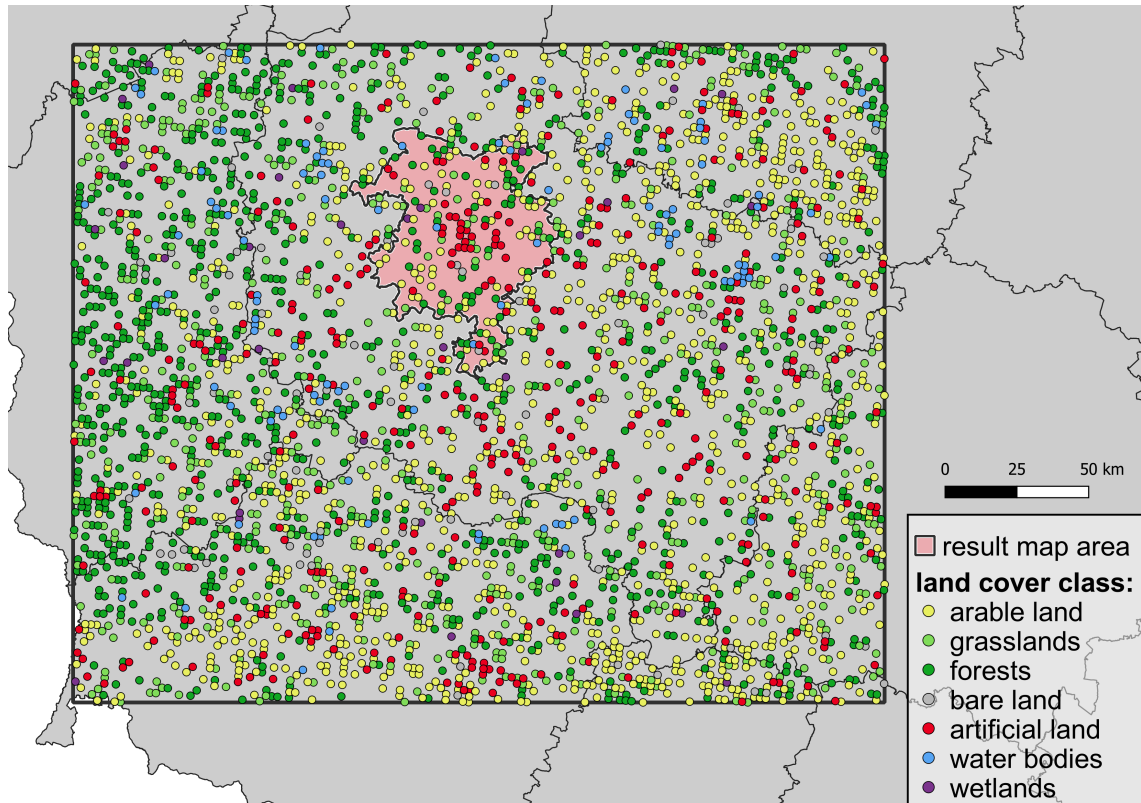
Figure 2.4: *Spatial distribution of LUCAS training points after pre-processing*

2.3 Machine learning

Machine learning is a computation method used to teach machines to learn from datasets automatically, without being specifically programmed (Mahesh (2018); Sarker (2021)). We can divide machine learning methods into two main groups: supervised and unsupervised.

Unsupervised learning analyzes unlabeled datasets without the need for human intervention. This is widely used for extracting generative features, identifying meaningful trends and structures, grouping results and exploratory purposes (Sarker, 2021). This type of machine learning discovers hidden patterns or data groupings (clusters) which is used in exploratory analysis or objects segmentation. The most common unsupervised learning algorithm is clustering. REF NEEDED

Supervised learning uses labeled training data and a collection of training examples, which are used by an algorithm to find relationships between different variables used to describe training data. It is carried out when certain goals are identified to be

accomplished from a certain set of inputs. There are two types of supervised learning tasks: classification (separating data) and regression (fitting data) (Sarker, 2021).

In this study, supervised classification algorithm called Random Forest (RF) was used (Breiman, 2001).

2.3.1 Random forest algorithm

I chose Random Forest as an algorithm used in this study. It is a very popular machine learning tool thanks to its high interpretability and relatively high accuracy (Qi, 2012). Other advantages of this algorithm is its ability to handle missing values, wide spectrum of accepted variable types (continuous, binary, categorical) and ease of modelling high-dimensional data (Qi, 2012). Random Forest consists of a specified number of decision trees, which are based on series of splitting rules.

Decision tree aims to partition the dataset into smaller, more homogeneous groups (Kuhn et al., 2013). This process creates a set of rules by dividing dataset into several categories. Each rule in the decision tree is specified by a feature (variable used to split) and a threshold (value of a feature dividing dataset) (Sekulić et al., 2020). Random forest algorithm is characterized by using many decision trees at the same time and receiving results by applying majority voting system based on outputs of all decision trees (Kuhn et al., 2013). Each tree in the forest has slightly different input data - a subset of data is sampled with replacement to get different result in every tree. This process is known as bagging or bootstrap aggregating (Schonlau et al., 2020). Moreover, algorithm is allowed to use only subset (randomly sampled) of available variables which reduces correlation between trees (Sohil et al., 2022).

2.3.2 Parameter tuning

Random Forest algorithm takes several hyper-parameters as an input in order to specify how much should it fit to training data or how long computation should take. Optimizing these parameters is crucial for tree-based machine learning models (Yang et al., 2020). Model's hyper-parameters can be fine-tuned to find values that give the best model accuracy. I chose three hyper-parameters for tuning: number of trees, maximum depth

Table 2.3: *Tuned parameters of RF model*

Hyper-parameter	Search space	Optimal value
number of trees	50 - 400	272
maximum depth	10 - 40	20
min. node size	1 - 10	1

of the forest and minimal size of each node in decision tree. Then, 10 models with different hyper-parameter values chosen randomly from specified search space were created and trained. I used overall accuracy achieved by each classifier to rank their performance and choose parameters that best train the model. Parameters' search spaces and tuning results can be found in Table 2.3.

2.3.3 Model quality assessment

Accuracy of the model was assessed using four performance measures:

- overall accuracy - ratio of number of correct predictions to the total number of input points

- Kappa coefficient - how well the classification performed as compared to randomly assigning values

- recall (producer's accuracy) - how often are real features on the ground correctly shown on the classified map

- precision (user's accuracy) - how often the class on the map will actually be present on the ground

- F1-score - harmonic mean between precision and recall, measures if classifier both classifies data correctly and does not miss a significant number of points

Values of these indices were estimated with the help of resampling technique called spatial cross-validation (CV).

In *n*-fold cross-validation, every data point is used in both training and testing set. Whole dataset is randomly divided into *n* equal parts (*folds*). Then, machine learning model is

independently trained *n* times and in each run, different part of the dataset is used as validation set while remaining *n - 1* parts are used to fit the model. This way, every data point is used in testing set only once and is used to train the model in the remaining runs (Jiao et al., 2016). Usually, whole cross-validation procedure is repeated several times to get higher number of unique dataset splits and to receive more reliable average values of the overall accuracy (Varga et al., 2021). Such approach is a compromise which enables possibility of using a whole dataset in the training process of the final model without a need of acquiring independent testing set.

Since this study is based on geographic data, Tobler's "first law of geography" and spatial autocorrelation need to be taken into account. Tobler stated that "everything is related to everything else, but near things are more related than distant things" (Tobler, 1970). In order to prevent testing points from being related to training points, I applied spatial cross-validation approach which aims to prevent the model to over-fit to the training data. This method is different than regular cross-validation only in the partitioning step - instead of randomly dividing dataset into groups, location of data points is used together with k-means clustering (Brenning, 2012) in order to create spatially disjoint folds (Lovelace et al., 2019). FIGURE? Thanks to this partitioning method, spatial bias can be significantly reduced which leads to more reliable performance estimation.

With the aim to determine values of model's hyperparameters as accurately as possible, I performed nested spatial cross-validation. This method is an extension of previously described approach, with hyperparameter tuning added to the process. Each fold created in the spatial CV is further divided into next *k* folds which comprise the tuning level of the process. Then, another cross-validation is performed on these folds in order to determine performance of randomly sampled hyperparameter values. The best of 10 (????) hyperparameter combinations is chosen to train the model on outer fold on performance estimation level (Schratz et al., 2019). Whole process is then repeated on every of *n* outer folds which leads to most accurate performance measurement as well as defining the best hyperparameter setting. FIGURE?

2.4 Variable importance and its spatial distribution

Quantifying importance of model's variables is a part of evaluating its results. This method can be used in model simplification and exploration, domain-knowledge-based validation or knowledge generation (Biecek et al., 2021). This study is focused on the latter purpose since my aim was to check if thermal information has an impact on land-cover classification.

Importance of model variables can be measured on two levels: dataset and instance level (Biecek et al., 2021). On the dataset level, we can measure change in model's accuracy depending on presence of one specific variable (Section 2.4.1). This gives basic knowledge about this variable's impact on model predictions. Assessing importance on instance level helps to understand impact of variables for one specific data point (Section 2.4.2). Moreover, instance-level importance can be utilized to interpolate variable importance values into continuous raster data (Section 2.4.3).

2.4.1 Dataset level

Measuring variable importance on dataset level requires evaluating model twice: once with original data and once with permuted variables of analysed variable. The main idea behind this action is to measure difference between models' performance. Breiman (2001) assumes that if variable is important, then it is expected to lower model's performance after permuting its values. In this study, I chose cross entropy as a measure of model's performance thus considering its change as a measure of variable importance.

2.4.2 Instance level

Second method to measure variable importance in machine learning model is instance level evaluation. It helps to find out how much each variable contributed to classification result for particular observation (Biecek et al., 2021). One way of calculating variable impact on observation result is creating break-down plot (Figure 2.4). Its main idea is to estimate contribution of variable by measuring change in model's predictions while
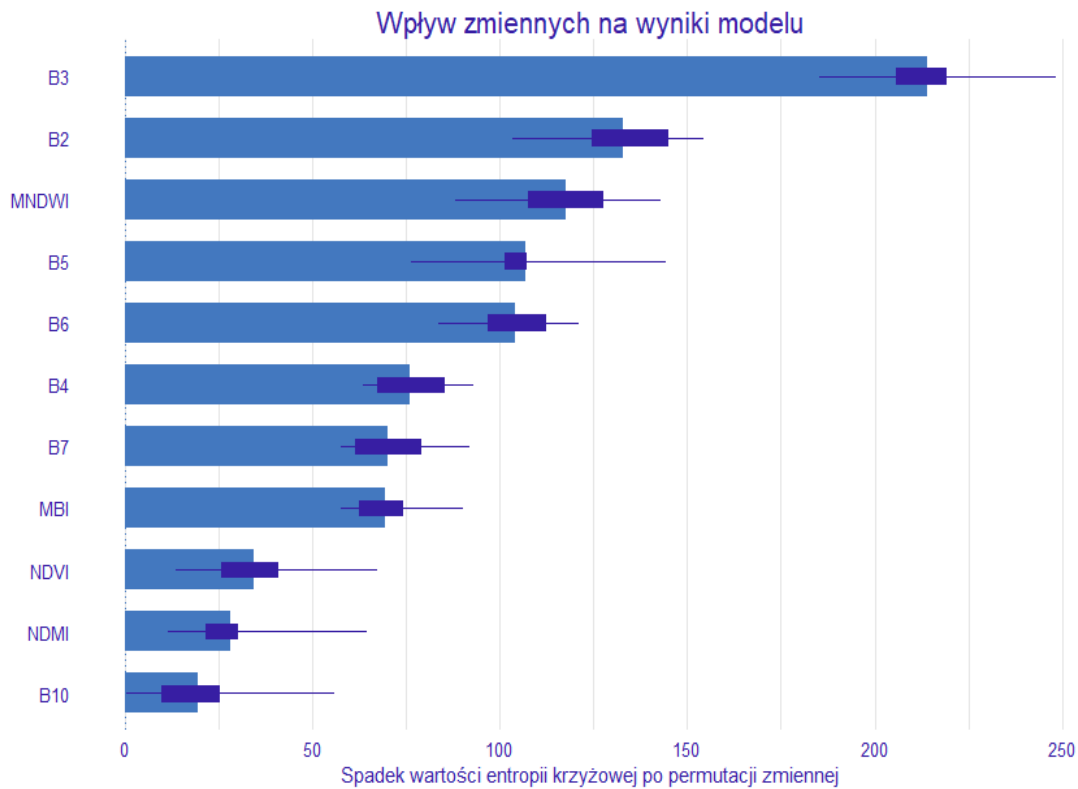
Figure 2.5: *Example of break-down plot that visualises variables' impact on chosen observation*

fixing the values of consecutive variables to values recorded for the chosen observation (Biecek et al., 2021). After fixing the value of variable for whole dataset, change in model's prediction is calculated. This value indicates variable impact on chosen observation.

However, above method is highly dependent on variable ordering and interactions between these variables (Biecek et al., 2021). To address this issue, I applied another approach based on averaging values from multiple break-down plots, each one with different ordering of the variables. This method originates from "Shapley values" (Shapley et al., 1953) and was applied to machine learning by Štrumbelj and Kononenko (2010). Main idea of this approach is to apply 10 different variable orderings, create a break-down plot for each of them and calculate the mean value of contribution for each variable. Thanks to this method, the influence of variable ordering can be mostly removed (Biecek et al., 2021).

Eventually, Shapley values provide possibilty to measure contribution of each variable in every observation in the training set. Such result enables us to add spatial context to the variable importance, which is further described in Section 2.4.3.

2.4.3 Interpolating variable importance

- calculating thermal band importance on aggregated raster

- interpolating thermal band importance from training points

2.5 R language environment

Almost every step of analysis described in previous sections was performed with use of R (R Core Team, 2021) - an open-source programming language designed mainly for statistical computing and visualizing data. I used RStudio (RStudio Team, 2020) as an integrated development environment (IDE) created for R. Apart from base R functionalities, I used a number of packages created by R community. We used *terra* package (Hijmans, 2022) to perform raster data operations and *sf* (Pebesma, 2022) to manipulate and process vector data. To conduct machine learning steps of the analysis, we used whole environment of various machine learning packages called *mlr3* (Lang et al., 2022). Random forest algorithm used by *mlr3* framework is part of *ranger* package (Wright et al., 2021) . I also used *dplyr* (Wickham et al., 2022) and *tidyr* packages (Wickham, 2021) to clean and process tabular data. *DALEX* (Biecek et al., 2022) and *DALEXtra* (Maksymiuk et al., 2022) packages provided various functionalities enabling me to estimate variable importance and visualize these results with the help of *ggplot2* package (Wickham et al., 2021). Package called *gstat* (Pebesma et al., 2021) helped to interpolate variable importance values from points to continuous raster layer. In addition, *future* package (Bengtsson, 2021) was used to enable multi-threading of some computationally intensive tasks.

Chapter 3

Result of the model - land cover map

- land cover map of Poznań metropolitan area

- comparison of land cover map and RGB satellite imagery

- probability map of model results

Część **Wyniki** może składać się z jednego lub więcej rozdziałów. Każdy z tych rozdziałów powinien mieć tytuł adekwatny do swojej treści.

Rozdziały wynikowe powinny korzystać z wiedzy opisanej w poprzednich rozdziałach (Rozdziały **?@sec-lit**, **?@sec-dane**, **?@sec-metody**). W przypadku prac analitycznych, ich treść powinna przedstawiać kolejne etapy eksploracji i analizy danych. W przypadku prac technicznych, treść tych rozdziałów powinna opisywać stworzone narzędzia, a następnie pokazywać ich zastosowanie/a.

W przypadku prac technicznych warto pokazywać fragmenty napisanego rozwiązania lub jego wywołania używając bloków kodu.

```r
moja_funkcja = function(x){

  cat(x, "rządzi!")

}

moja_funkcja("Autor tej pracy")
```

Autor tej pracy rządzi!

Chapter 4

Assessing model quality

Table with quality indices:

- overall accuracy (OA) / classification error (CE)

- producer's and user's accuracy (PA, UA) / precission and recall

- F1-score

- Kappa coefficient

Część **Wyniki** może składać się z jednego lub więcej rozdziałów. Każdy z tych rozdziałów powinien mieć tytuł adekwatny do swojej treści.

Rozdziały wynikowe powinny korzystać z wiedzy opisanej w poprzednich rozdziałach (Rozdziały **?@sec-lit**, **?@sec-dane**, **?@sec-metody**). W przypadku prac analitycznych, ich treść powinna przedstawiać kolejne etapy eksploracji i analizy danych. W przypadku prac technicznych, treść tych rozdziałów powinna opisywać stworzone narzędzia, a następnie pokazywać ich zastosowanie/a.

W przypadku prac technicznych warto pokazywać fragmenty napisanego rozwiązania lub jego wywołania używając bloków kodu.

```r
moja_funkcja = function(x){

  cat(x, "rządzi!")

}

moja_funkcja("Autor tej pracy")
```

Autor tej pracy rządzi!

Chapter 5

Evaluation of thermal band's impact on prediction results

5.1 Overall importance of thermal band

- mean temperature for every predicted land cover class

- variable importance plots, variable profiles

- mean importance of thermal band on each land cover class

5.2 Spatial distribution of thermal band's importance

- thermal band importance map (two methods: raster aggregation and interpolation of importance in LUCAS points)

- difference raster map between prediction with and without thermal band included, transition matrix

————————————————————

Część **Wyniki** może składać się z jednego lub więcej rozdziałów. Każdy z tych rozdziałów powinien mieć tytuł adekwatny do swojej treści.

Rozdziały wynikowe powinny korzystać z wiedzy opisanej w poprzednich rozdziałach (Rozdziały **?@sec-lit**, **?@sec-dane**, **?@sec-metody**). W przypadku prac analitycznych, ich treść powinna przedstawiać kolejne etapy eksploracji i analizy danych. W przypadku prac technicznych, treść tych rozdziałów powinna opisywać stworzone narzędzia, a następnie pokazywać ich zastosowanie/a.

W przypadku prac technicznych warto pokazywać fragmenty napisanego rozwiązania lub jego wywołania używając bloków kodu.

```
moja_funkcja = function(x){
  cat(x, "rządzi!")
}
moja_funkcja("Autor tej pracy")
```

```
Autor tej pracy rządzi!
```

Chapter 6

Conclusion

- land cover map of Poznań metropolitan area was created, impact of thermal band on classification results was measured

- despite thermal band having low overall impact on model results, there is a strong spatial auto-correlation for its importance

- land surface temperature was especially significant for land cover classification of urban areas, it helped in identify built-up areas

- it may mean that thermal band will become increasingly important in studies on urban sprawl and suburbanisation

- better land cover maps will help in better management of metropolitan areas growth and quantifying impact of urbanisation on natural environment more precisely

---

Podsumowanie pracy jest w pewnym sensie znacznie rozbudowanym abstraktem. Należy wyliczyć i opisać osiągnięcia uzyskane w pracy dyplomowej. Tutaj jednak (w przeciwieństwie do np. rozdziału **?@sec-wprowadzenie**) należy przechodzić od szczegółu do ogółu - co zostało stworzone/określone, jak zostało to zrobione, jakie ma to konsekwencje, itd.

Ten rozdział powinien też zawierać opis kwestii, których nie udało się rozwiązać w pracy dyplomowej (i dlaczego się nie udało) oraz pomysły na przyszłe ulepszenie uzyskanych wyników lub dalsze badania.

Bibliography

Bengtsson, H (2021). *future: Unified Parallel and Distributed Processing in R for Everyone*. R package version 1.23.0. https://CRAN.R-project.org/package=future.

Biecek, P and T Burzykowski (2021). *Explanatory Model Analysis*. Chapman and Hall/CRC, New York. https://pbiecek.github.io/ema/.

Biecek, P, S Maksymiuk, and H Baniecki (2022). *DALEX: moDel Agnostic Language for Exploration and eXplanation*. R package version 2.4.0. https://CRAN.R-project.org/package=DALEX.

Breiman, L (2001). Random Forests. *Machine Learning* **45**(1), 5–32.

Brenning, A (2012). Spatial cross-validation and bootstrap for the assessment of prediction rules in remote sensing: The R package sperrorest. en. In: *2012 IEEE International Geoscience and Remote Sensing Symposium*. Munich, Germany: IEEE, pp.5372–5375. http://ieeexplore.ieee.org/document/6352393/ (visited on 01/03/2023).

Buck, O, C Haub, S Woditsch, D Lindemann, L Kleinewillinghöfer, G Hazeu, B Kosztra, S Kleeschulte, S Arnold, and M Hölzl (2015). *Analysis of the LUCAS nomenclature and proposal for adaptation of the nomenclature in view of its use by the Copernicus land monitoring services*. https://land.copernicus.eu/user-corner/technical-library/LUCAS_Copernicus_Report_v22.pdf.

d'Andrimont, R, M Yordanov, L Martinez-Sanchez, B Eiselt, A Palmieri, P Dominici, J Gallego, HI Reuter, C Joebges, G Lemoine, and M van der Velde (2020). Harmonised LUCAS in-situ land cover and use database for field surveys from 2006 to 2018 in the European Union. en. *Scientific Data* **7**(1), 352. (Visited on 11/13/2022).

Hijmans, RJ (2022). *terra: Spatial Data Analysis*. R package version 1.5-21. https://rspatial.org/terra/.

Jiao, Y and P Du (2016). Performance measures in evaluating machine learning based bioinformatics predictors for classifications. en. *Quantitative Biology* **4**(4), 320–330. (Visited on 01/03/2023).

Kuhn, M and K Johnson (2013). *Applied Predictive Modeling*. en. New York, NY: Springer New York. http://link.springer.com/10.1007/978-1-4614-6849-3 (visited on 12/20/2022).

Lang, M, B Bischl, J Richter, P Schratz, and M Binder (2022). *mlr3: Machine Learning in R - Next Generation*. R package version 0.13.3. https://CRAN.R-project.org/package= mlr3.

Lovelace, R, J Nowosad, and J Muenchow (2019). *Geocomputation with R*. CRC Press.

Mahesh, B (2018). Machine Learning Algorithms - A Review. en. **9**(1).

Maksymiuk, S, P Biecek, and H Baniecki (2022). *DALEXtra: Extension for DALEX Package*. R package version 2.2.0. https://CRAN.R-project.org/package=DALEXtra.

Pebesma, E (2022). *sf: Simple Features for R*. R package version 1.0-7. https://CRAN.R-project.org/package=sf.

Pebesma, E and B Graeler (2021). *gstat: Spatial and Spatio-Temporal Geostatistical Modelling, Prediction and Simulation*. R package version 2.0-8. https://github.com/r-spatial/gstat/.

Pflugmacher, D, A Rabe, M Peters, and P Hostert (2019). Mapping pan-European land cover using Landsat spectral-temporal metrics and the European LUCAS survey. en. *Remote Sensing of Environment* **221**, 583–595. (Visited on 04/14/2022).

Potapov, P, MC Hansen, I Kommareddy, A Kommareddy, S Turubanova, A Pickens, B Adusei, A Tyukavina, and Q Ying (2020). Landsat Analysis Ready Data for Global Land Cover and Land Cover Change Mapping. en. *Remote Sensing* **12**(3), 426. (Visited on 04/07/2022).

QGIS Development Team (2009). *QGIS Geographic Information System*. https://www.qgis.org.

Qi, Y (2012). Random Forest for Bioinformatics. en.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/.

RStudio Team (2020). *RStudio: Integrated Development Environment for R*. Boston, MA. http://www.rstudio.com/.

Sarker, IH (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. en. *SN Computer Science* **2**(3), 160. (Visited on 12/15/2022).

Schonlau, M and RY Zou (2020). The random forest algorithm for statistical learning. en. *The Stata Journal: Promoting communications on statistics and Stata* **20**(1), 3–29. (Visited on 12/20/2022).

Schratz, P, J Muenchow, E Iturritxa, J Richter, and A Brenning (2019). Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data. en. *Ecological Modelling* **406**, 109–120. (Visited on 01/03/2023).

Sekulić, A, M Kilibarda, GB Heuvelink, M Nikolić, and B Bajat (2020). Random Forest Spatial Interpolation. en. *Remote Sensing* **12**(10), 1687. (Visited on 12/20/2022).

Shapley, LS, KJ ARROW, EW BARANKIN, D BLACKWELL, R BOTT, N DALKEY, M DRESHER, D GALE, DB GILLIES, I GLICKSBERG, O GROSS, S KARLIN, HW KUHN, JP MAYBERRY, JW MILNOR, TS MOTZKIN, J VON NEUMANN, H RAIFFA, LS SHAPLEY, M SHIFFMAN, FM STEWART, GL THOMPSON, and RM THRALL (1953). "A VALUE FOR n-PERSON GAMES". In: *Contributions to the Theory of Games (AM-28), Volume II*. Ed. by HW Kuhn and AW Tucker. Princeton University Press, pp.307–318. http://www.jstor.org/stable/j.ctt1b9x1zv.24 (visited on 01/09/2023).

Sohil, F, MU Sohali, and J Shabbir (2022). An introduction to statistical learning with applications in R: by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, New York, Springer Science and Business Media, 2013, $41.98, eISBN: 978-1-4614-7137-7. en. *Statistical Theory and Related Fields* **6**(1), 87–87. (Visited on 01/03/2023).

Strumbelj, E and I Kononenko (2010). An Efficient Explanation of Individual Classifications using Game Theory. en.

Tobler, WR (1970). A Computer Movie Simulating Urban Growth in the Detroit Region. en. *Economic Geography* **46**, 234. (Visited on 01/04/2023).

Varga, OG, Z Kovács, L Bekő, P Burai, Z Csatáriné Szabó, I Holb, S Ninsawat, and S Szabó (2021). Validation of Visually Interpreted Corine Land Cover Classes with Spectral Values of Satellite Images and Machine Learning. en. *Remote Sensing* **13**(5), 857. (Visited on 04/07/2022).

Wickham, H (2021). *tidyr: Tidy Messy Data*. R package version 1.1.4. `https://CRAN.R-project.org/package=tidyr`.

Wickham, H, W Chang, L Henry, TL Pedersen, K Takahashi, C Wilke, K Woo, H Yutani, and D Dunnington (2021). *ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics*. R package version 3.3.5. `https://CRAN.R-project.org/package=ggplot2`.

Wickham, H, R François, L Henry, and K Müller (2022). *dplyr: A Grammar of Data Manipulation*. R package version 1.0.10. `https://CRAN.R-project.org/package=dplyr`.

Wright, MN, S Wager, and P Probst (2021). *ranger: A Fast Implementation of Random Forests*. R package version 0.13.1. `https://github.com/imbs-hl/ranger`.

Yang, L and A Shami (2020). On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice. en. *Neurocomputing* **415**. arXiv:2007.15745 [cs, stat], 295–316. (Visited on 01/02/2023).