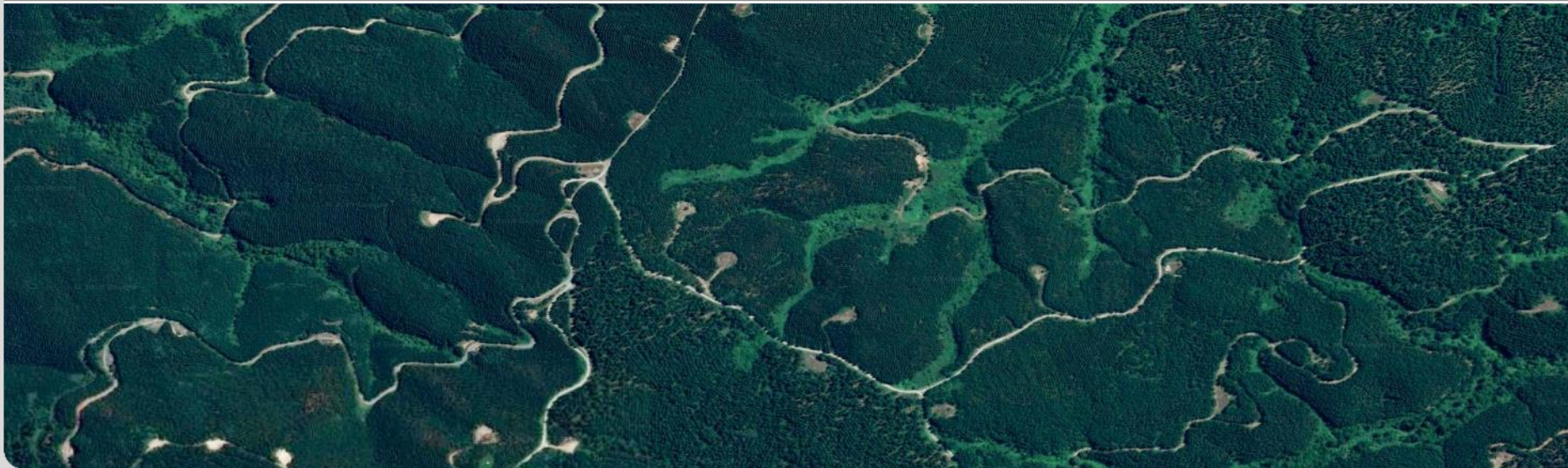


# Methodenbootcamp Sustainability 2.0: Klassifizierung von Satellitenbildern

**Christian Nestroy**  
**Benedikt Brückner**

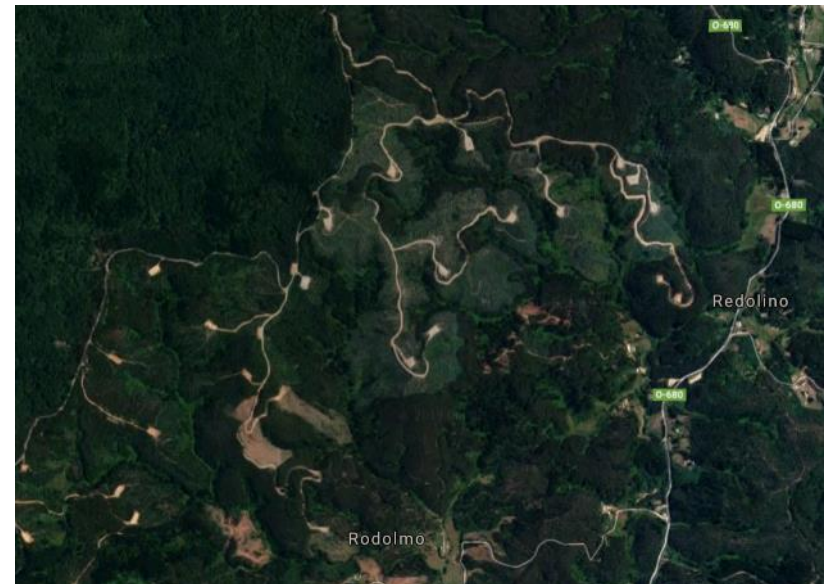
Institut für Industriebetriebslehre und Industrielle Produktion (IIP),  
Lehrstuhl für Betriebswirtschaftslehre, insbesondere Produktionswirtschaft und Logistik



# Motivation



Karte



Satellit

# Motivation

- Spezielle topographische Gestalt Chiles
- Große Bedeutung des Straßenverkehrs
- Biomass-to-Liquid als Alternative zur Reduktion der Treibhausgasemissionen
- Holzabbau und –transport verantwortlich für bis zu 50% des Kraftstoffpreises
- Netzwerk an Forstwegen erfassen, um besser modellieren und planen zu können

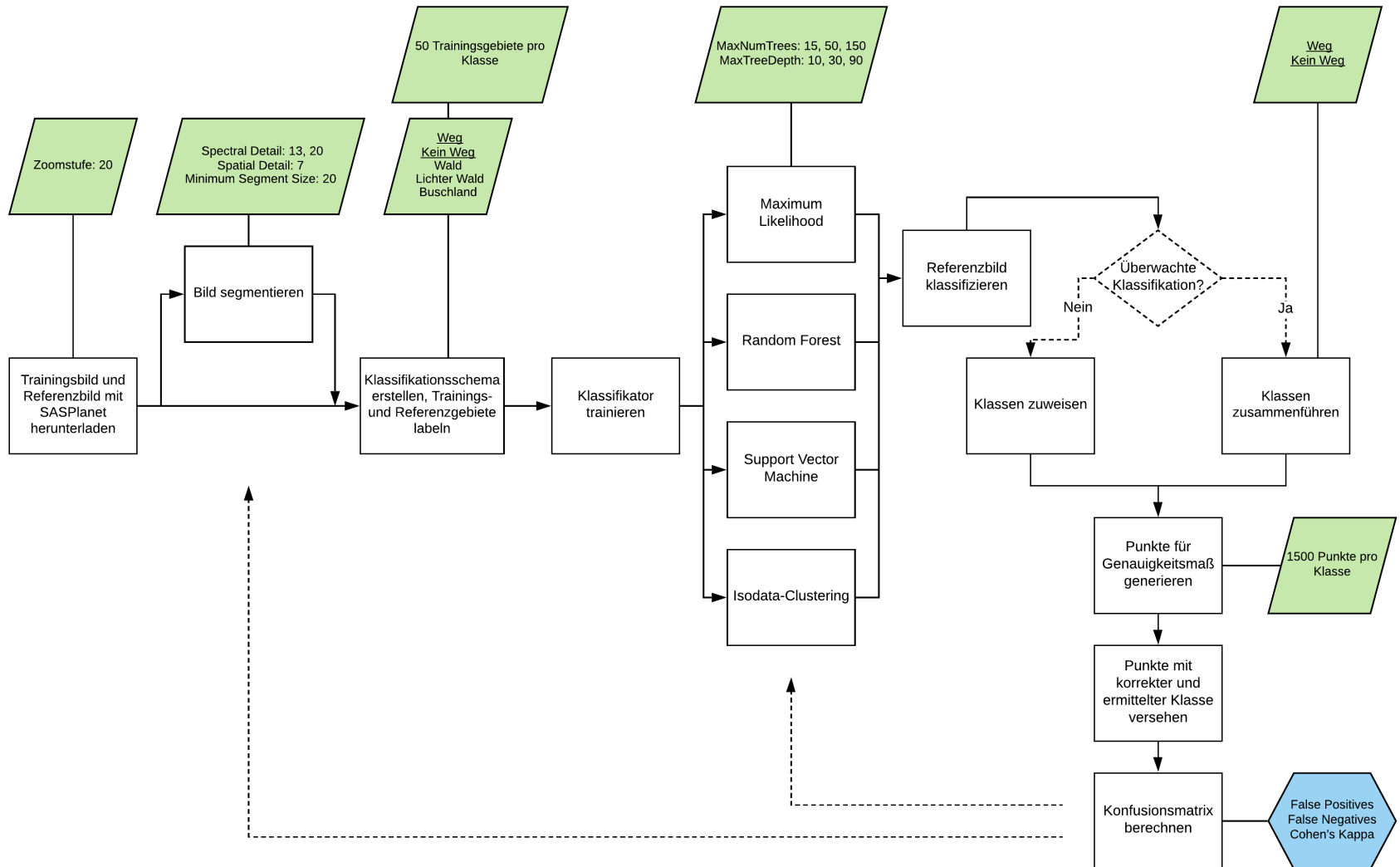


# Gliederung

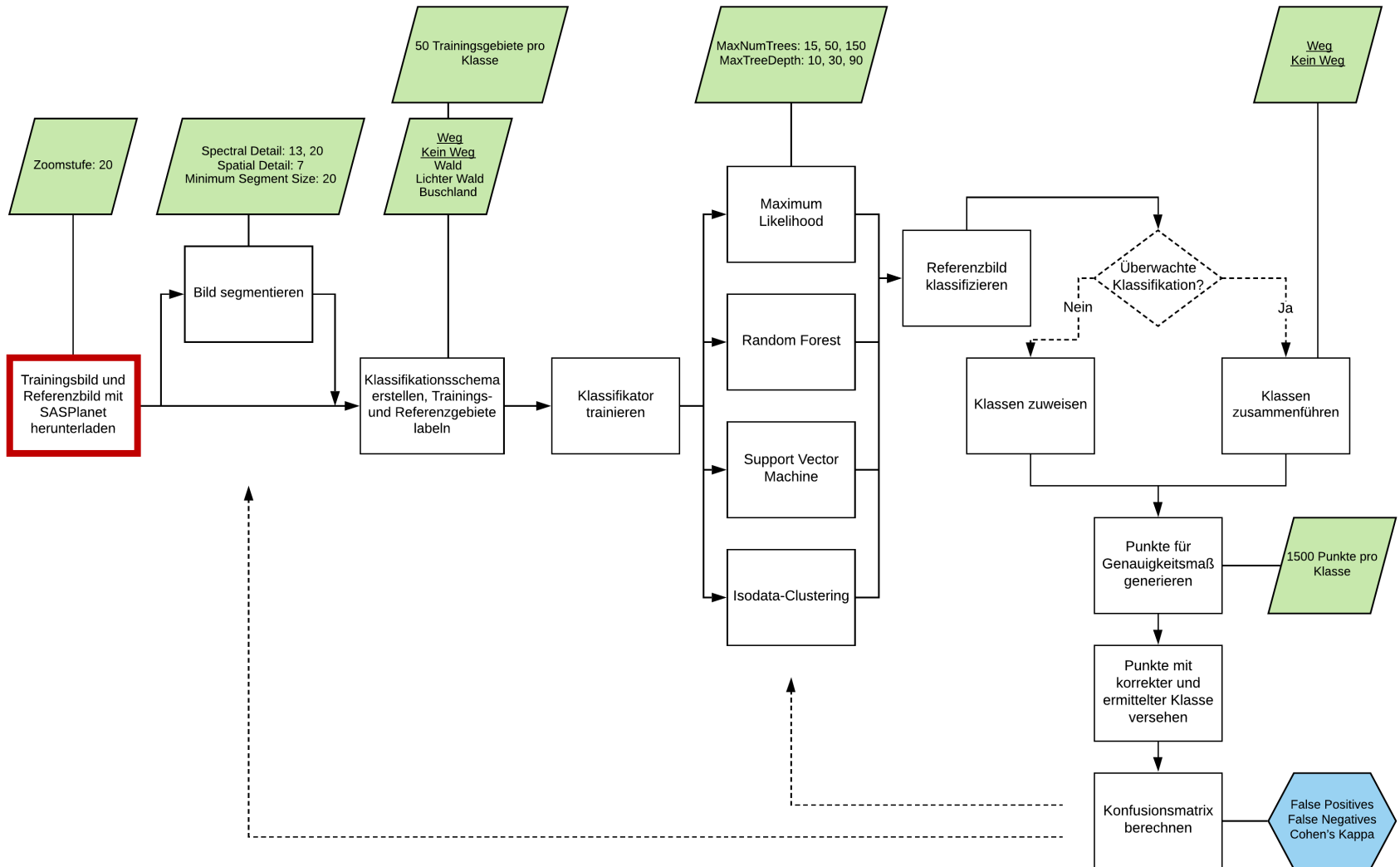
- Motivation
- Zielstellung
- Maschinelles Lernen und Bildsegmentierung
- Methodik
- Ergebnisse
- Fazit und Ausblick

# Zielstellung

- Klassifikation von Satellitenbildern mittels Methoden des Maschinellen Lernens
- Automatisierte Erkennung von Forstwegen
- Verwendung der Erweiterung Image Analyst im Geoinformationssystem ArcGIS Pro
- Voraussetzung für Anschluss an bestehendes Transportnetz im Network Analyst



# Beschaffung von Satellitenbildern

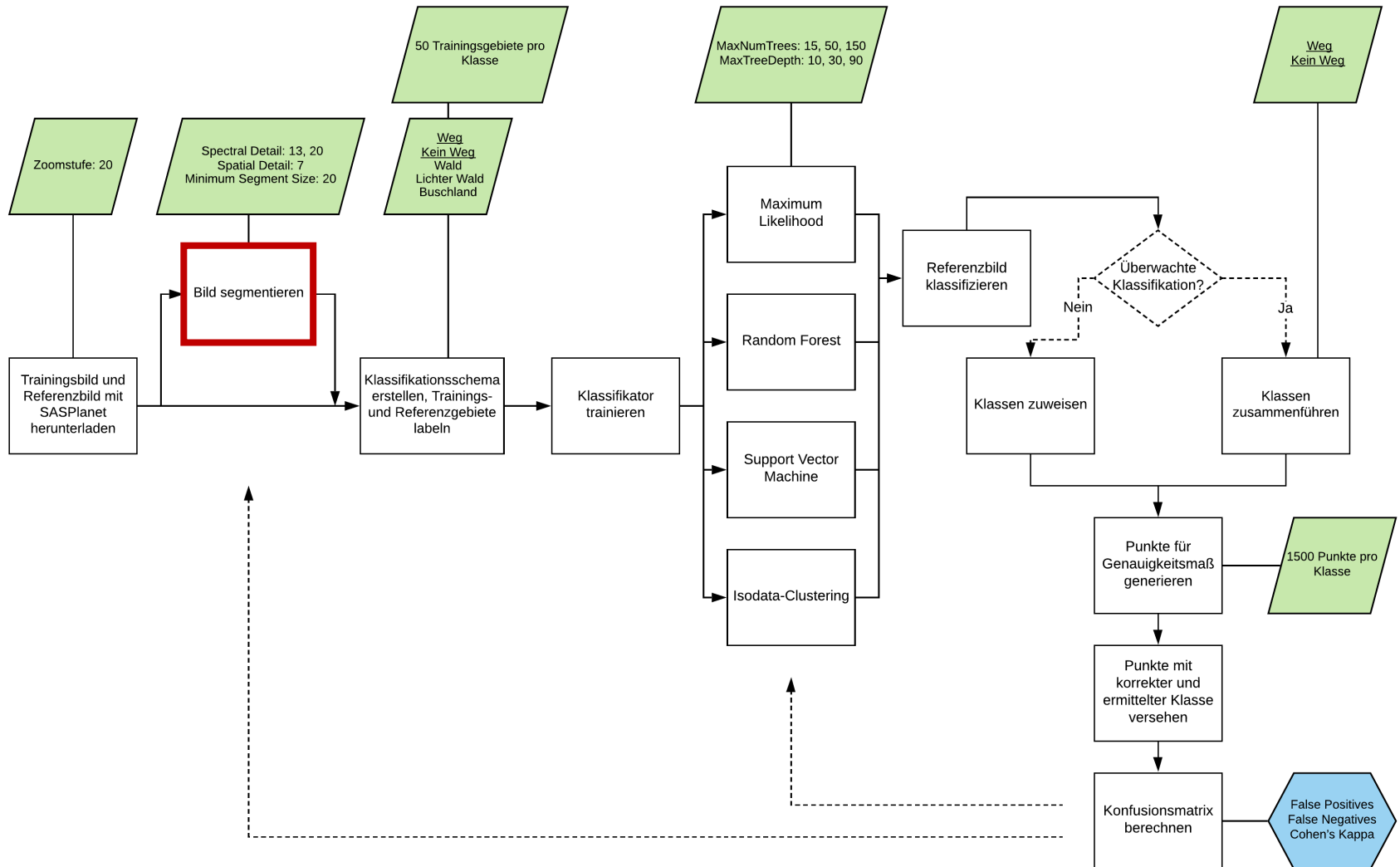


# Beschaffung von Satellitenbildern

- Qualität der verwendeten Satellitenbilder beeinflusst das Ergebnis entscheidend
- Satellitenbilder aus Google Maps
  - kostenlos, aber ausreichende Qualität
  - Download mittels „SASPlanet“
- Verschiedene Auflösungen beim Download möglich
  - Kompromiss aus Qualität und Verarbeitbarkeit: ca. 90MB Dateigröße für 4 km<sup>2</sup> Fläche



# Bildsegmentierung



# Bildsegmentierung

- Ziel: Zusammenfassen von Bildbereichen und deren Homogenisierung
  - Einsatz von Clusteringalgorithmen

- Beispiel:

Original



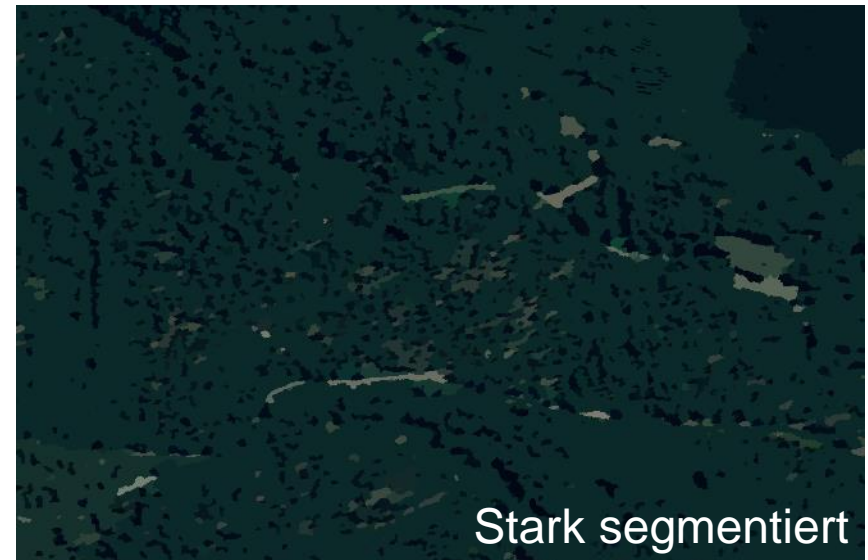
Segmentiert



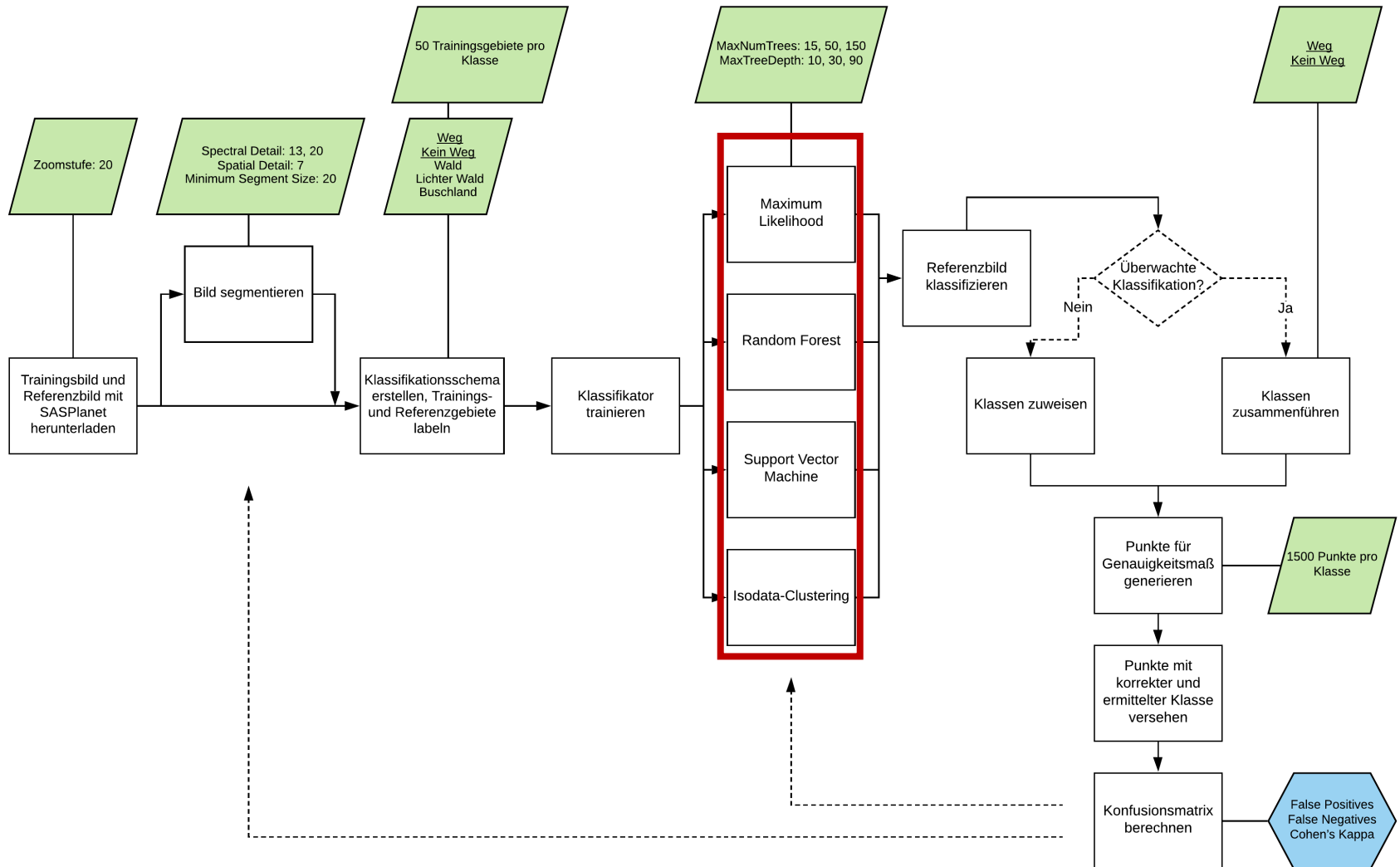
# Bildsegmentierung

- Über Parameter wird kontrolliert
  - wie stark sich Punkte unterscheiden dürfen, die in einem Cluster zusammengefasst werden
  - welche Mindestgröße ein Cluster haben muss
- Finden der optimalen Parameter ist aufgrund von Interdependenzen schwierig → Tests
  - Ergebnis: Klassifikation mittels Originalbildern und schwacher bis mittelstarker Segmentierung testen

# Bildsegmentierung



# Klassifikationsverfahren



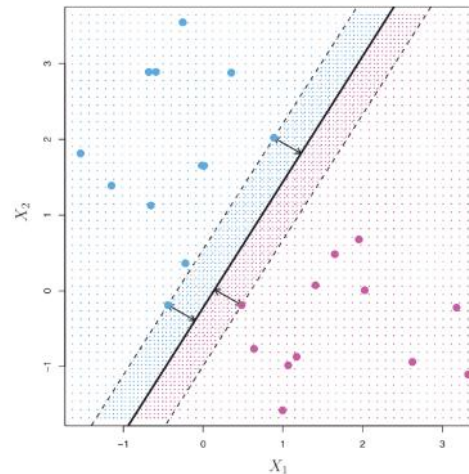
# Klassifikationsverfahren

- Nutzung der in ArcGIS Pro implementierten Verfahren
- Überwachte Verfahren
  - Maximum Likelihood
  - Random Forest
  - Support Vector Machine
- Unüberwachtes Verfahren
  - Isodata-Clustering

# Support Vector Machine

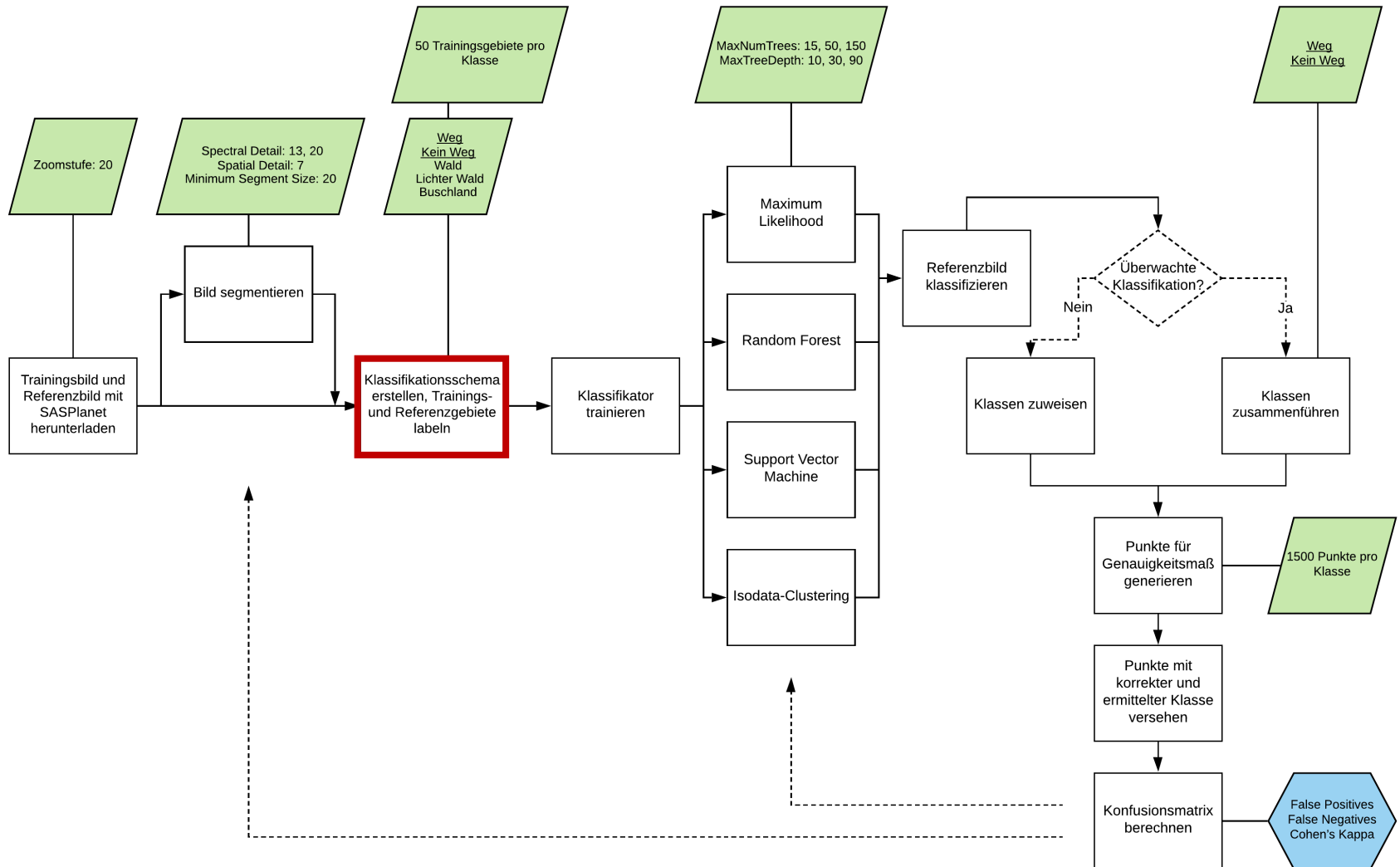
- Idee: Konstruiere eine Hyperebene im  $\mathbb{R}^p$ , die die Trainingsdaten bestmöglich separiert
- Wahl der Hyperebene mit dem maximalen Margin

$$\begin{aligned}
 & \max_{\beta_0, \beta_1, \dots, \beta_p} M \\
 & \text{subject to } \sum_{j=1}^p \beta_j^2 = 1 \\
 & y_i(\beta_{i0} + \beta_{i1}x_{i1} + \beta_{i2}x_{i2} + \dots + \beta_{ip}x_{ip}) \geq M \quad \forall i = 1, \dots, n
 \end{aligned}$$






# Klassifikationsschema und Trainingsgebiete






# Klassifikationsschema und Trainingsgebiete

Trainingsgebiet-Manager : Trainingsbild\_segme... :45

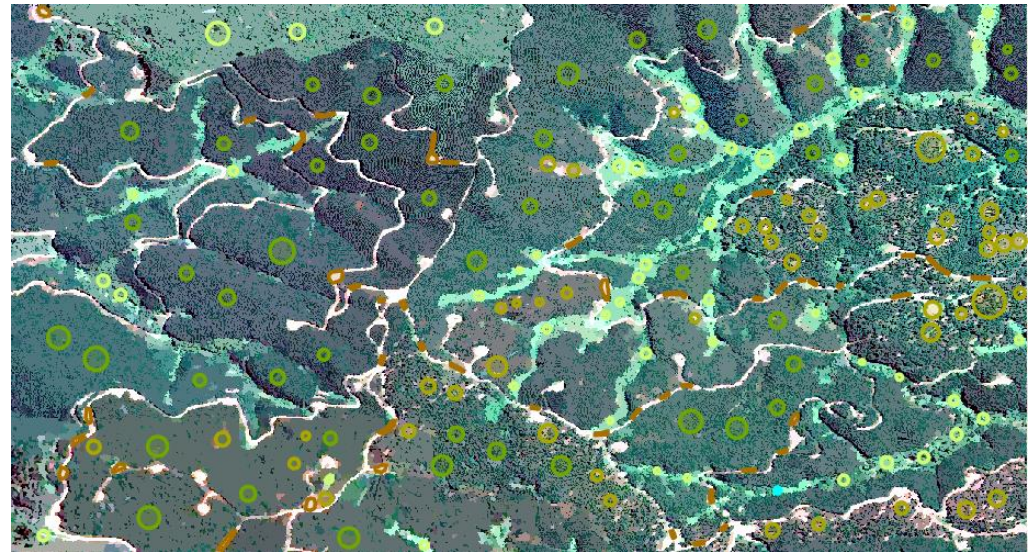


**Klassifikationsschema Seminararbeit**

- Kein Weg
  - Wald
  - Buschland
  - Lichter Wald
  - Weg



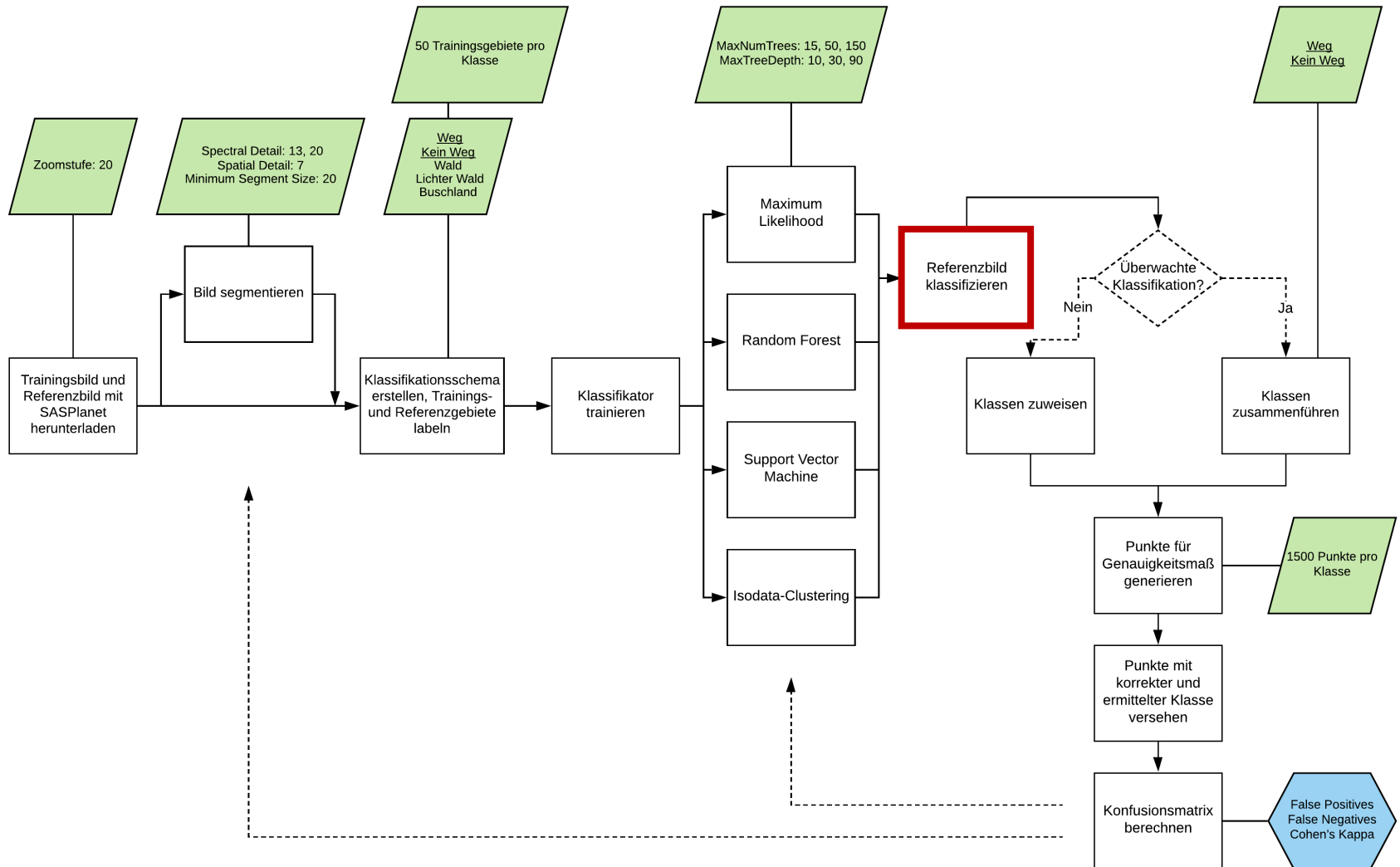
Klasse	Anz. d. Stichpro	Pixel (%)
Weg	1	0,27
Weg	1	0,29
Weg	1	0,12
Wald	1	0,80
Wald	1	2,10
Wald	1	0,55
Wald	1	0,63



# Klassifikationsschema und Trainingsgebiete

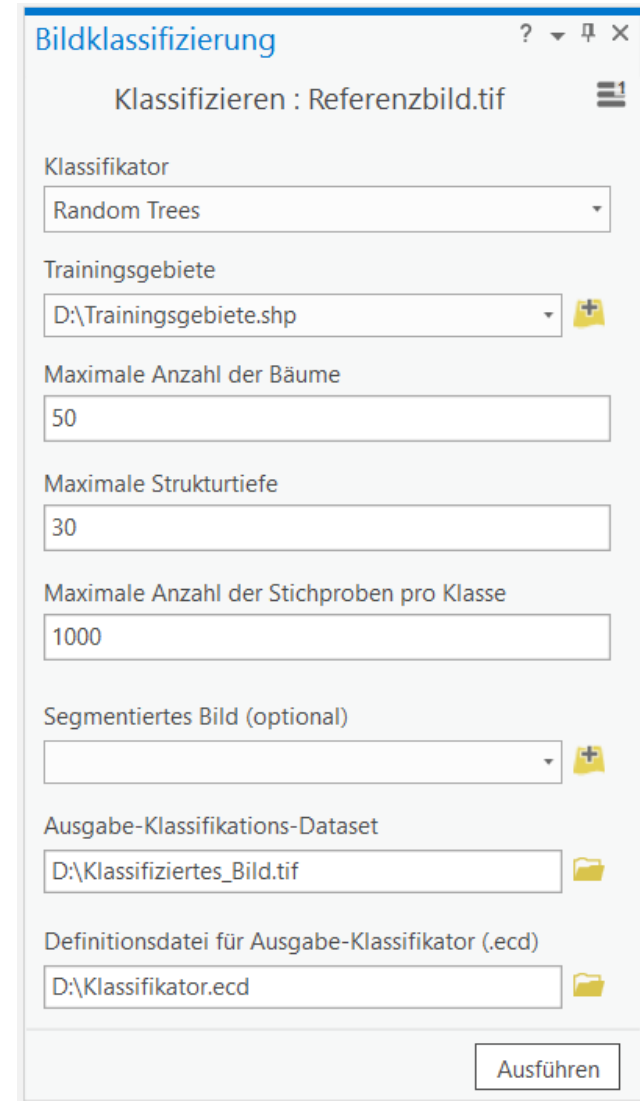
- Zweckmäßige Auswahl des Trainingsbildes
- Überwachte Klassifikation: Welche Klassen sind vorhanden?
- Evtl. Vorklassifikation, um Wasserflächen oder Stadtgebiete bereits auszuschließen
- Ausreichende Anzahl an Trainingsgebieten definieren, Verteilung beachten

# Klassifizierung in ArcGIS



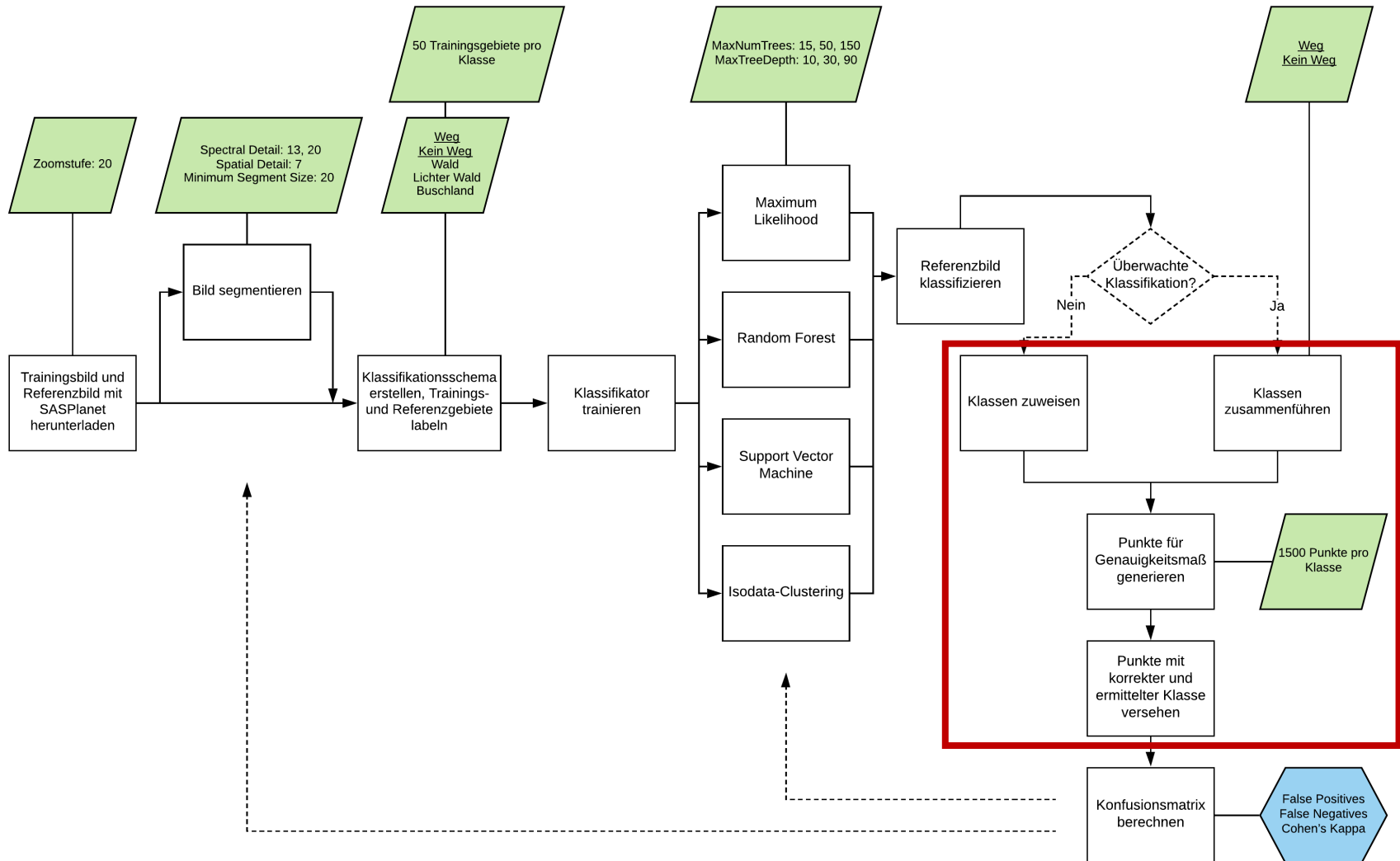
# Klassifizierung in ArcGIS

- Mittels des Image Classification Wizard
  - Auswahl des Klassifikators
  - Angabe von Ein-/Ausgabepfad
  - Angabe der Parameter
  
- Nach der Ausführung
  - Ausgabe des trainierten Klassifikators
  - Ausgabe des klassifizierten Bildes
  
- Problem: Im GUI keine trainierten Klassifikatoren einlesbar
  
- Lösung: Verwendung der Python-Schnittstelle
  - Trainiere auf Trainingsbild
  - Klassifiziere Referenzbild



The screenshot shows the 'Bildklassifizierung' (Image Classification) dialog box in ArcGIS. The title bar indicates the current task is 'Klassifizieren : Referenzbild.tif'. The 'Klassifikator' (Classifier) is set to 'Random Trees'. The 'Trainingsgebiete' (Training Areas) are specified as 'D:\Trainingsgebiete.shp'. The 'Maximale Anzahl der Bäume' (Maximum Number of Trees) is set to 50, and the 'Maximale Strukturtiefe' (Maximum Structure Depth) is set to 30. The 'Maximale Anzahl der Stichproben pro Klasse' (Maximum Number of Samples per Class) is set to 1000. The 'Segmentiertes Bild (optional)' (Segmented Image (optional)) field is empty. The 'Ausgabe-Klassifikations-Dataset' (Output Classification Dataset) is 'D:\Klassifiziertes\_Bild.tif'. The 'Definitionsdatei für Ausgabe-Klassifikator (.ecd)' (Definition file for Output Classifier (.ecd)) is 'D:\Klassifikator.ecd'. An 'Ausführen' (Run) button is located at the bottom right.

# Postprocessing



# Postprocessing

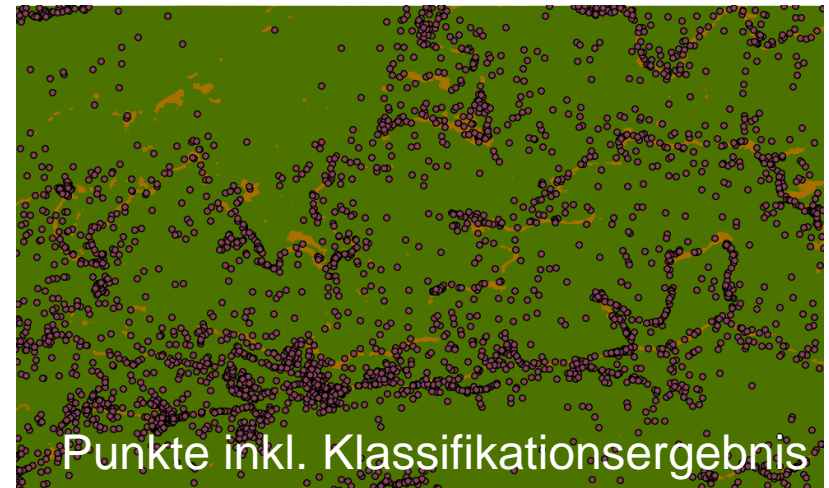
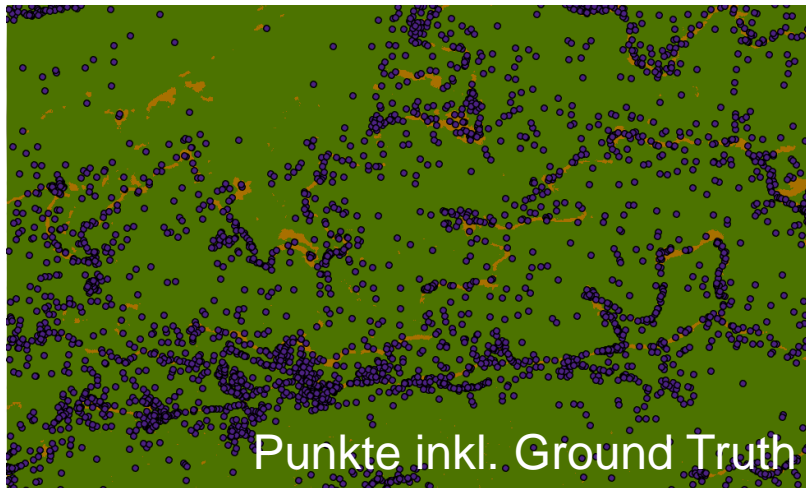
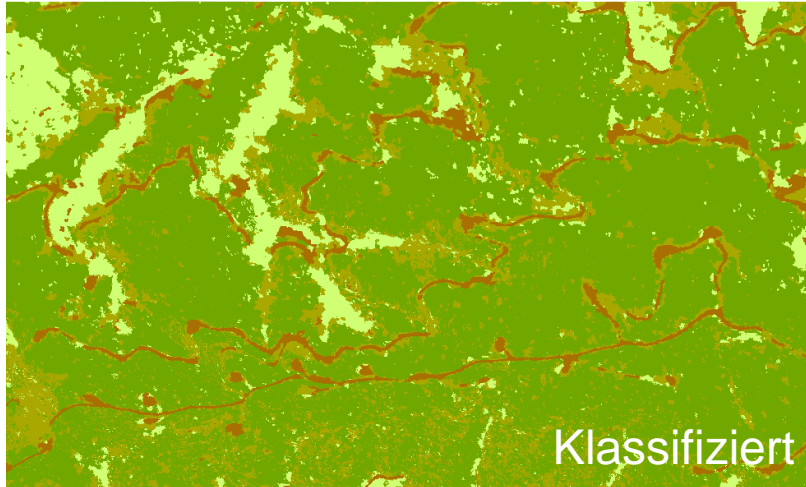
- Klassen zusammenführen, um Klassifikationsschema an Aufgabenstellung anzupassen
- Erstellte Klassen den vordefinierten Klassen zuweisen bei unüberwachter Klassifikation
- Generierung von Stichprobenpunkten für Genauigkeitsbewertung
- Berechnung der Konfusionsmatrix und des Genauigkeitsmaßes  $\kappa$

True  
Positive  
False  
Negative

Konfusionsmatrix_Ref...axns1000_merged						
Feld:		Auswahl:				
OID	Class Value	C_10	C_20	Total	U Accuracy	Kappa
0	C_10	1004	34	1038	0,967245	0
1	C_20	496	1464	1960	0,746939	0
2	Total	1500	1498	2998	0	0
3	P_Accuracy	0,669333	0,977303	0	0,823215	0
4	Kappa	0	0	0	0	0,646503

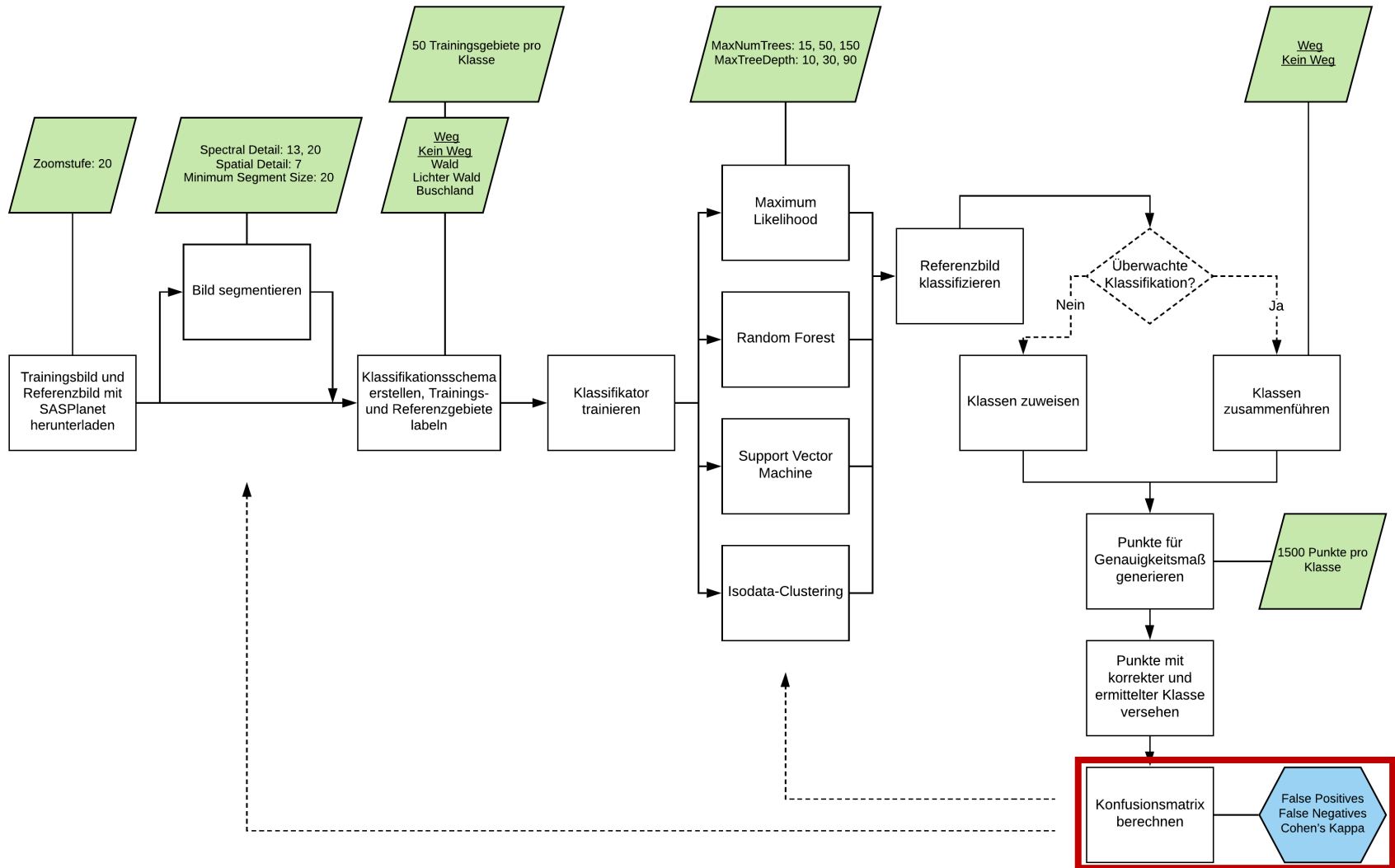
False  
Positive  
True  
Negative  
  
Cohen's  
Kappa

# Postprocessing





# Ergebnisse



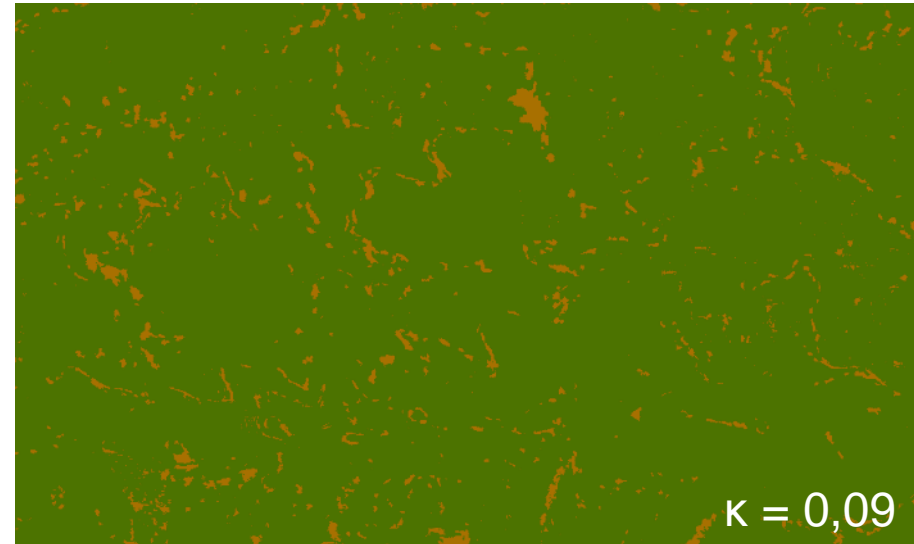
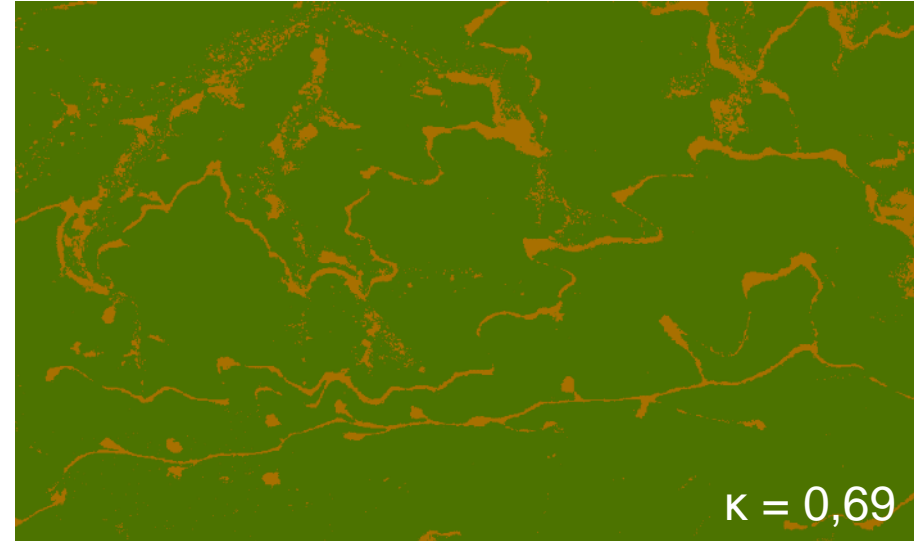


# Ergebnisse

- Durchführung von Tests
  - Trainingsbild mit Trainingsgebieten
  - manuell gelabeltes Referenzbild zur Berechnung der Konfusionsmatrizen
- Vier verschiedene Varianten der Bilder (drei verschieden stark segmentierte und das Originalbild)
- Verschiedene Parameter der Lernverfahren

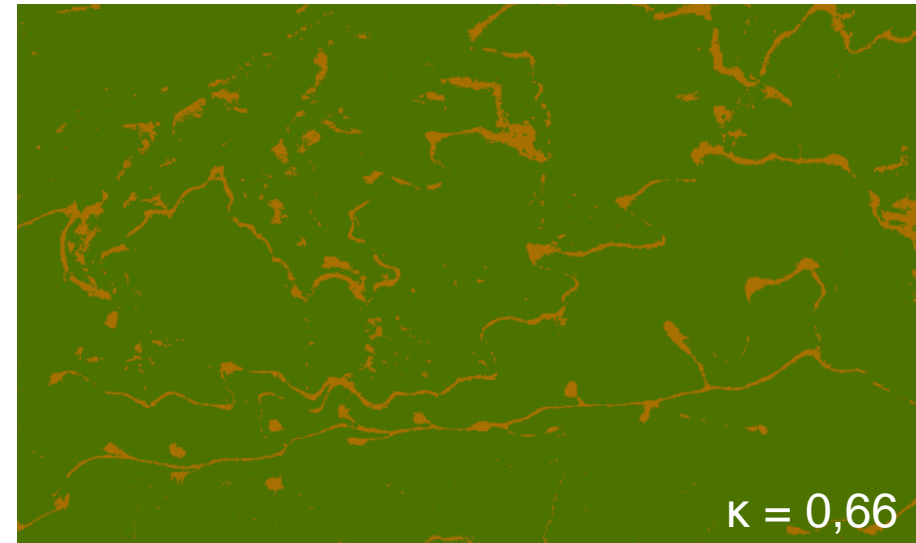
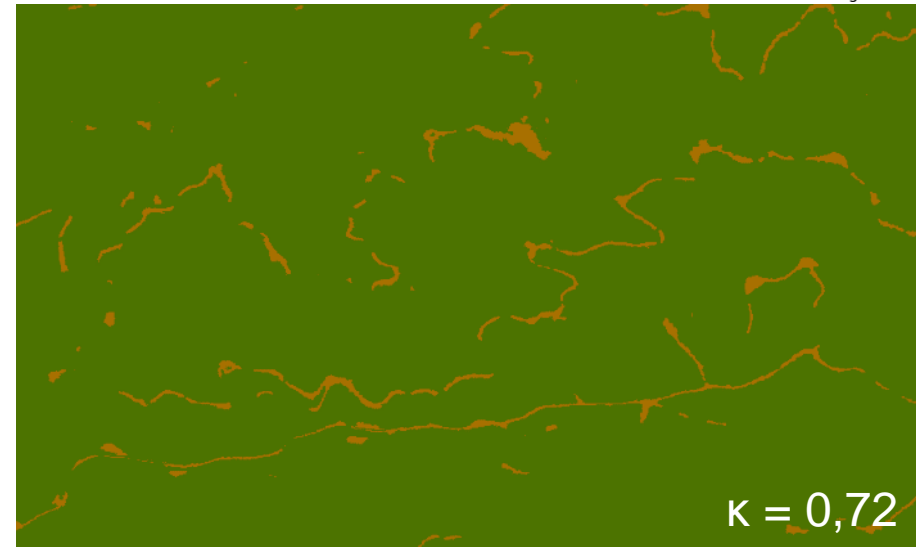
# Ergebnisse

- Isodata-Clustering (unüberwacht)
  - Keine Segmentierung am besten
  - Starke Segmentierung → Extrem schlechte Ergebnisse
  - $\kappa$  zwischen 0,09 und 0,69



# Ergebnisse

- Support Vector Machine
  - Starke Segmentierung am besten
  - Sehr robust
  - $\kappa$  zwischen 0,66 und 0,72



# Fazit

- Solide Ergebnisse bei Klassifikation von Satellitenbildern in ArcGIS Pro
- Random Forest und Support Vector Machine dominieren, Clustering nicht empfehlenswert
- Segmentierung vorteilhaft bei guter Parameterwahl
- Unterschiede zwischen überwachten und unüberwachtem Verfahren in Methodik und Python-Skripten
- Dokumentation und Tutorial in ArcGIS eher knapp gehalten
- ArcGIS-Funktionen teilweise nur in GUI oder Python-API enthalten

# Ausblick

- Systematische Wahl von Trainingsbild und Verfahrensparametern
- Kontrastabweichungen → Google Maps als Bildquelle für große Bereiche kritisch
- Modellierung des Transportnetzes im Modul Network Analyst ausstehend
- Deep Learning evtl. vielversprechend, Realisierung in ArcGIS schwierig

# FRAGEN UND DISKUSSION

# BACKUP

# Maximum-Likelihood-Klassifizierung

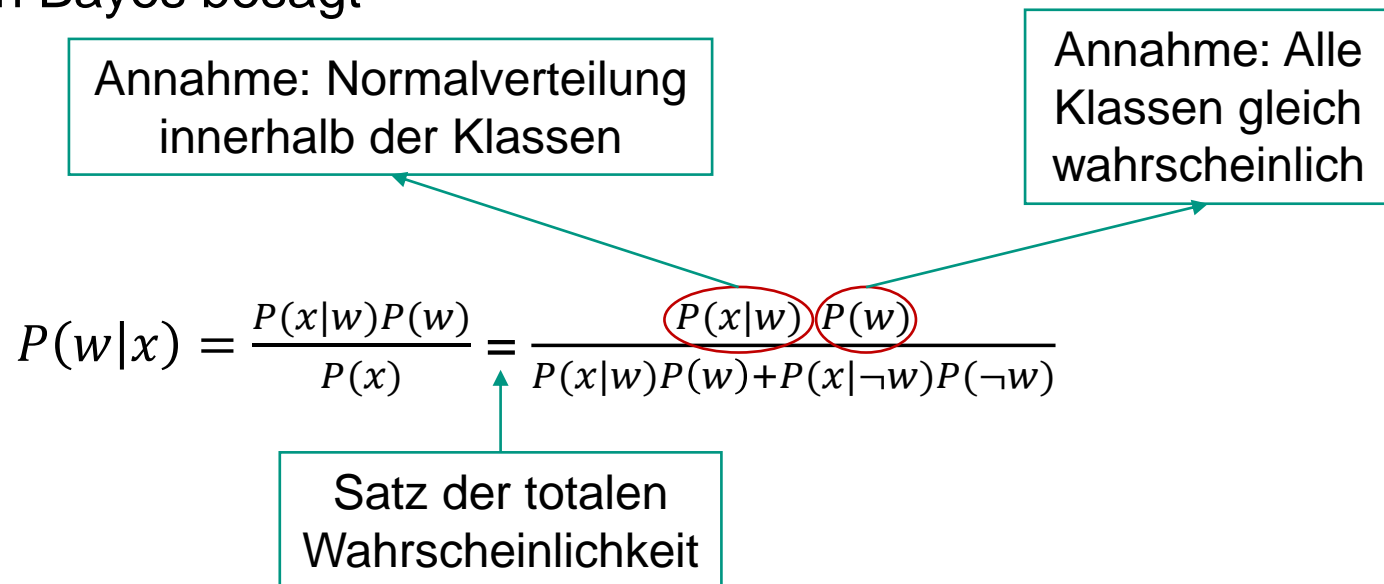
- Einfaches Verfahren, das auf dem Satz von Bayes basiert
- Erklärung am Beispiel
  - Seien zwei Klassen  $w$  (Weg) und  $\neg w$  (kein Weg) gegeben
  - Sei  $x$  ein Bildpunkt, der klassifiziert werden soll
  - Es sind einige Trainingsbeispiele gegeben Bildpunkte mit bekannter Klasse
- Wir möchten die Klasse von  $x$  bestimmen
- Satz von Bayes besagt

Annahme: Normalverteilung  
innerhalb der Klassen

Annahme: Alle  
Klassen gleich  
wahrscheinlich

$$P(w|x) = \frac{P(x|w)P(w)}{P(x)} = \frac{P(x|w)P(w)}{P(x|w)P(w) + P(x|\neg w)P(\neg w)}$$

Satz der totalen  
Wahrscheinlichkeit

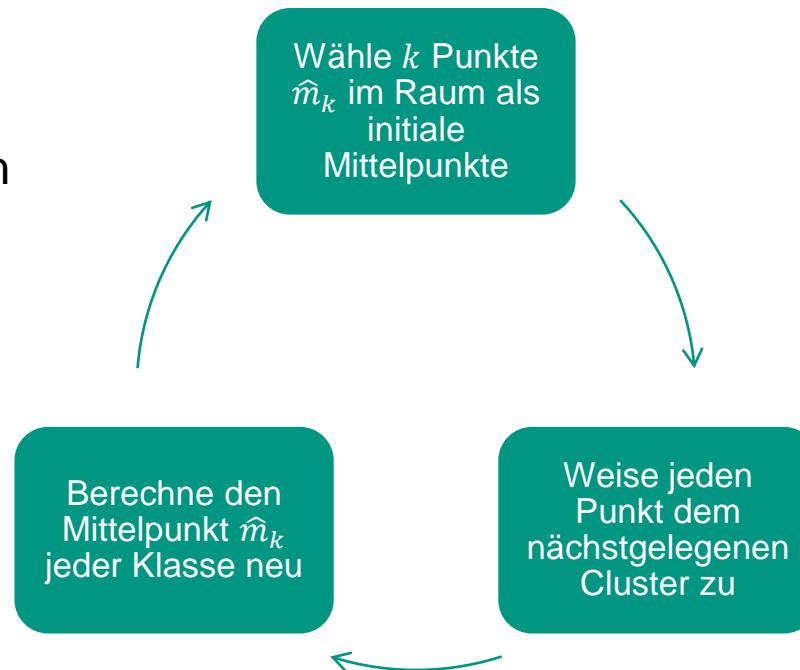




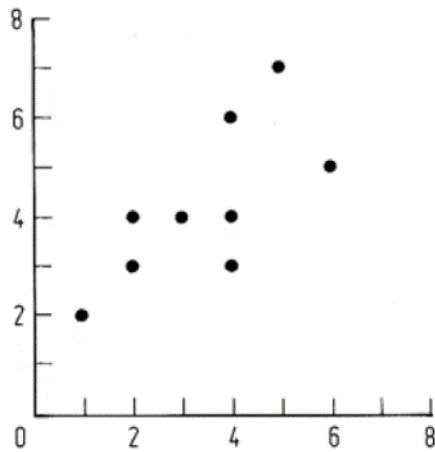
# Isodata-Clustering

- Unüberwachtes Lernverfahren
- Ziel: Zusammenfassen von gegebenen Punkten in  $k$  Klassen
- Grundalgorithmus: k-means-Clustering

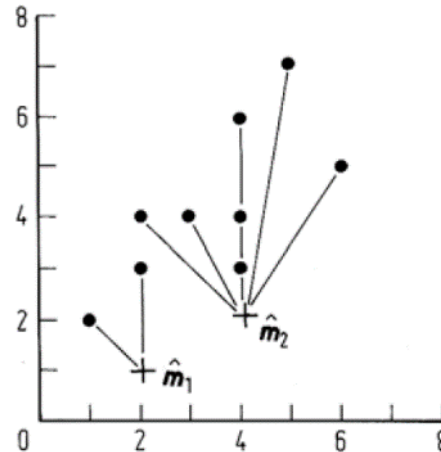
Abbruch, wenn sich  
Mittelpunkte kaum  
noch ändern



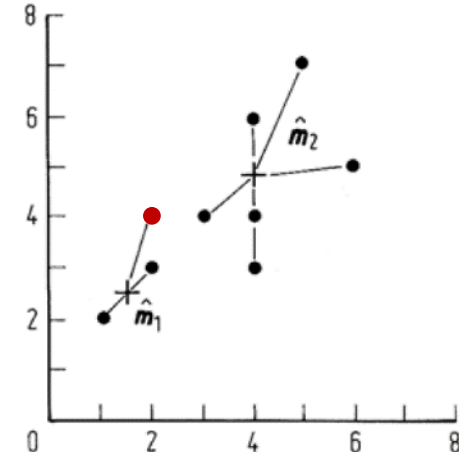
# Isodata-Clustering



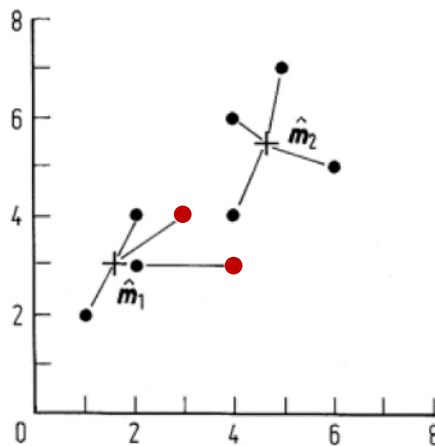
Ausgangssituation



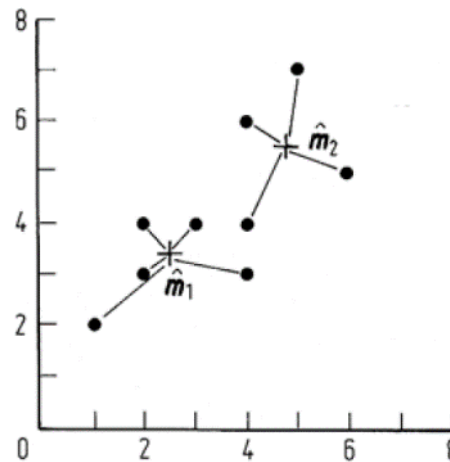
Initialisierung



Iteration 1



Iteration 2



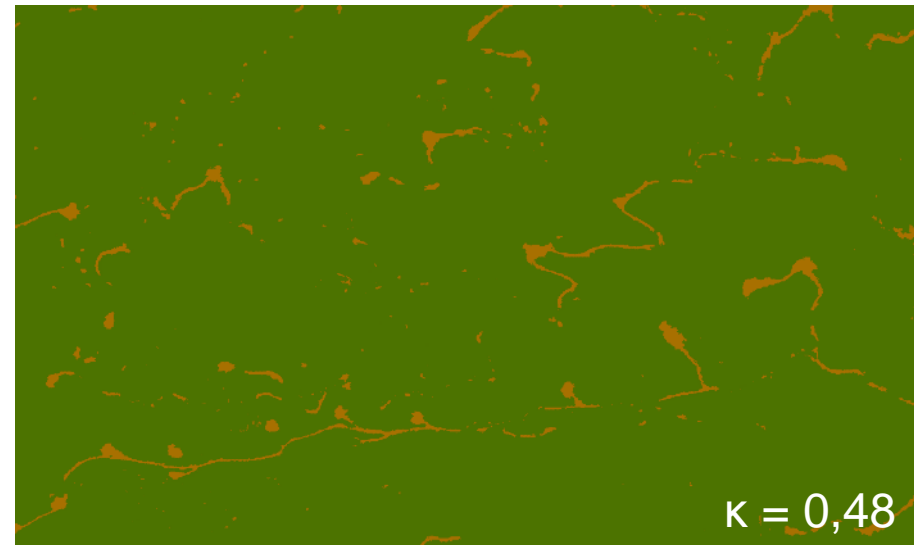
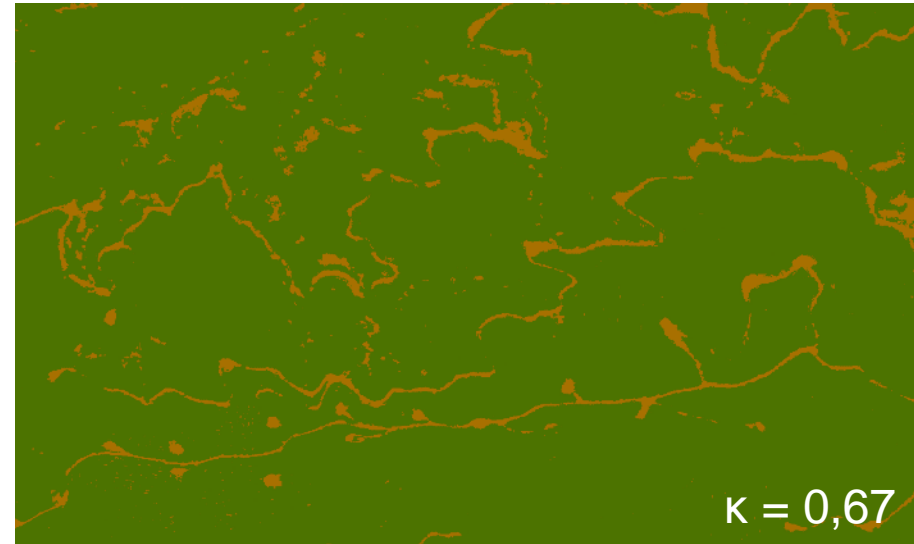
Iteration 3 (letzte Iteration)

# Cohen's Kappa

- Cohen's Kappa ist definiert als  $\kappa = \frac{p_o - p_e}{1 - p_e}$
- Wobei:
  - $p_o$ : Gemessene Übereinstimmung von vorhergesagter und korrekter Klasse
  - $p_e$ : Übereinstimmung, die zu erwarten wäre, wenn der Klassifikator die Gebiete zufallsbasiert unter Beachtung der Klassenhäufigkeiten klassifizieren würde
- Wertebereich:  $\kappa \in \left[-\frac{p_e}{1 - p_e}, 1\right]$

# Ergebnisse

- Maximum-Likelihood
  - Schwache Segmentierung am besten
  - $\kappa$  zwischen 0,48 und 0,67



# Ergebnisse

- Random Forest
  - Schwache Segmentierung am besten
  - Sehr robust
  - $\kappa$  zwischen 0,58 und 0,72

