Investigating Language Variation

from [brɪən] to [brodə] to [bʁujəʀ]

Martijn Wieling

Department of Computational Linguistics, University of Groningen

Language Technology Course, October 5th 2009

Overview

- Introduction
- Material
- Levenshtein distance
- Visualization of results
- Evaluation
- Extensions
  - Spoken word recognition
  - Bipartite spectral graph partitioning
- Discussion

History

- Much early work in variationist linguistics focused on investigating single features of dialects
  - e.g., lenition of /k/ to /ch/ in the word *ik*
- Goal: characterizing dialects and languages
- Isoglosses were used to visualize feature differences:

Why aggregation?

- Problem: which feature to select (highly subjective!)
- Different features show different patterns
- Taking together many features (aggregation) enables us to detect reliable relations (Nerbonne, 2009)
- The aggregative approach is very suitable for computational linguistics!

Why aggregation?

- Problem: which feature to select (highly subjective!)
- Different features show different patterns
- Taking together many features (aggregation) enables us to detect reliable relations (Nerbonne, 2009)
- The aggregative approach is very suitable for computational linguistics!

What data to aggregate over?

- We use pronunciation data (coded as IPA text) of many different words
  - Corresponds to what we hear
  - A large amount of this type of data is available
- But note that other types of data could also be used (e.g., morphological or phonological data)
- Focus of this talk on Dutch dialect pronunciations

Dutch dialect material

- Dutch dialect data source: the Goeman-Taeldeman-Van Reenen-Project data (GTRP; Goeman & Taeldeman, 1996)
- Transcriptions (IPA) of 1876 items for 613 localities
- Most recent Dutch dialect data source: 1980 – 1995
- We use a 562-word subset with diacritics removed

- Transcriptional differences BEL and NL → Focus on the Netherlands in this presentation (424 varieties)

Dutch dialect material

- Dutch dialect data source: the Goeman-Taeldeman-Van Reenen-Project data (GTRP; Goeman & Taeldeman, 1996)
- Transcriptions (IPA) of 1876 items for 613 localities
- Most recent Dutch dialect data source: 1980 – 1995
- We use a 562-word subset with diacritics removed

- Transcriptional differences BEL and NL $\rightarrow$ Focus on the Netherlands in this presentation (424 varieties)

Geographic distribution

Comparing pronunciations

- Levenshtein distance
  - Number of edit operations to transform one string into the other
  - Levenshtein distance between [mɔəlkə] and [mɛlək] is 4

mɔəlkə	subst. ɔ/ɛ	1
mɛəlkə	delete ə	1
mɛlkə	insert ə	1
mɛləkə	delete ə	1
mɛlək		
		4

m	ɔ	ə	l		k	ə	
m	ɛ		l	ə	k		
	1	1		1		1	

Comparing pronunciations

- Levenshtein distance
  - Number of edit operations to transform one string into the other
  - Levenshtein distance between [mɔəlkə] and [mɛlək] is 4

mɔəlkə	subst. ɔ/ɛ	1
mɛəlkə	delete ə	1
mɛlkə	insert ə	1
mɛləkə	delete ə	1
mɛlək		
		4

m	ɔ	ə	l		k	ə
m	ɛ		l	ə	k	
	1	1		1		1

Comparing pronunciations

- Levenshtein distance
  - Number of edit operations to transform one string into the other
  - Levenshtein distance between [mɔəlkə] and [mɛlək] is 4

mɔəlkə	subst. ɔ/ɛ	1
mɛəlkə	delete ə	1
mɛlkə	insert ə	1
mɛləkə	delete ə	1
mɛlək		
		4

m	ɔ	ə	l		k	ə
m	ɛ		l	ə	k	
	1	1		1		1

Improving the Levenshtein distance

- Traditional Levenshtein distance
  - Linguistic syllabicity constraint
  - No normalization (Heeringa et al., 2006)

- Improved Levenshtein distance
  - Sound segment distances are estimated from the data itself using an iterative Pointwise Mutual Information procedure (Wieling et al., 2009):

$$\mathrm{PMI}(x, y) = \log_2 \left(\frac{p(x, y)}{p(x)\, p(y)} \right)$$

  - This improves the following incorrect alignment:

l	ɛ	ɪ	k		ə	n
l	i		k	h	ɵ	n

Improving the Levenshtein distance

- Traditional Levenshtein distance
  - Linguistic syllabicity constraint
  - No normalization (Heeringa et al., 2006)
- Improved Levenshtein distance
  - Sound segment distances are estimated from the data itself using an iterative Pointwise Mutual Information procedure (Wieling et al., 2009):

$$\mathrm{PMI}(x, y) = \log_2 \left(\frac{p(x, y)}{p(x)\, p(y)} \right)$$

  - This improves the following incorrect alignment:

l	ɛ	ɪ	k		ə	n
l	i		k	h	ɵ	n

Calculating aggregate dialect distances

- To obtain the aggregate distance between each dialect pair, we simply average the Levenshtein distances of all word pairs (in our case 562)

Visualizing aggregate distances

- We are comparing pronunciations of different locations
- It makes sense to try to project the pronunciation distances onto a map
- There are several visualization options, e.g.:
  - Cluster map
  - Fuzzy cluster border map
  - Line map
  - Vector map
  - Multidimensional scaling (MDS) map

Cluster map

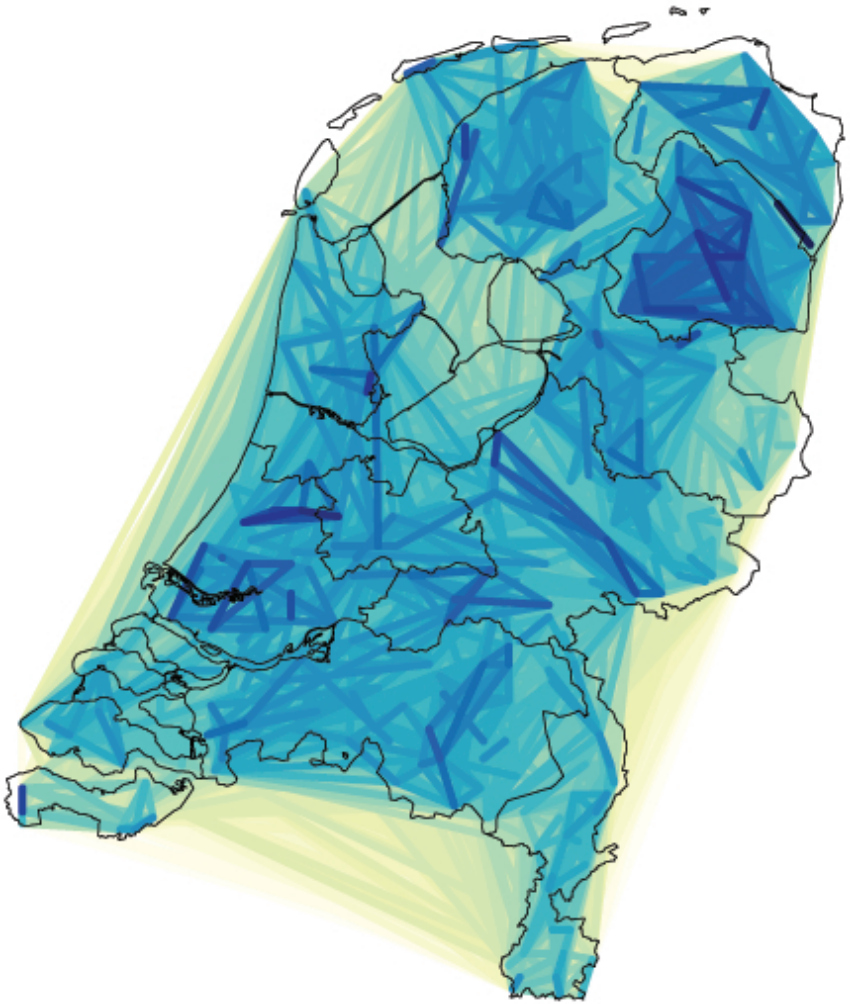
Closest varieties have the same color; not good to use!

Fuzzy cluster border map

Improvement over (unstable) clustering

Line map

Darker lines connect closer varieties

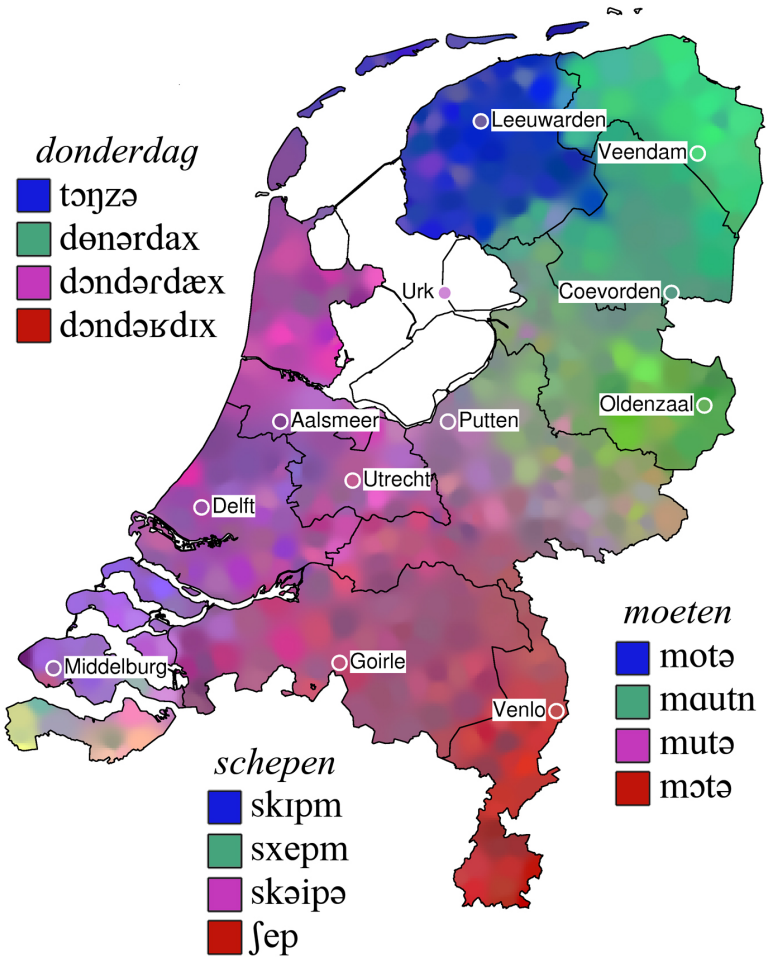


Vector map

Lines pointing to the neighborhood being most similar

MDS map of the Netherlands

Reduction to 3 dimensions mapped to RGB-color (explained variance: 87.5%)



External validation of aggregate distances

- These maps are nice to look at, but do they give a valid overview of the dialectal language variation?
- Yes! The Levenshtein distance seems to be a valid basis for determining dialect distances:
  - Gooskens & Heeringa (2004) found a significant correlation ($r \approx 0.7$) between perceptual linguistic distances and Levenshtein distances
  - Beijering et al. (2008) found similar results ($r \approx 0.6$)
- Additionally, the dialect areas we find are also identified by *experts* on Dutch dialectology to be distinct areas

External validation of aggregate distances

- These maps are nice to look at, but do they give a valid overview of the dialectal language variation?
- Yes! The Levenshtein distance seems to be a valid basis for determining dialect distances:
  - Gooskens & Heeringa (2004) found a significant correlation ($r \approx 0.7$) between perceptual linguistic distances and Levenshtein distances
  - Beijering et al. (2008) found similar results ($r \approx 0.6$)
- Additionally, the dialect areas we find are also identified by *experts* on Dutch dialectology to be distinct areas

Comparing different sets of aggregate distances

- As a comparison method we use LOCAL INCOHERENCE which assigns a score to a set of distances based on the idea that closer varieties should have more similar pronunciations (lower is better)

Extensions of the previous approach

- Two different extensions will be discussed next:
  - The first extension is based on intuitions from psycholinguistic work on spoken word recognition and modifies the Levenshtein algorithm to obtain a new set of pairwise distances
  - The second extension uses the aligned sound correspondences to simultaneously cluster varieties and obtain a linguistic basis for this clustering

Using intuitions from spoken word recognition

- Previously discussed Levenshtein algorithm: location of edit operation does not influence cost
- Psycholinguistic work on spoken word recognition:
  - Start of the word is more important than end of the word (Cohort Model; Marslen-Wilson, 1987)
  - Stressed syllable is important for word recognition (Altman & Carter, 1989)

Using intuitions from spoken word recognition

- Previously discussed Levenshtein algorithm: location of edit operation does not influence cost
- Psycholinguistic work on spoken word recognition:
  - Start of the word is more important than end of the word (Cohort Model; Marslen-Wilson, 1987)
  - Stressed syllable is important for word recognition (Altman & Carter, 1989)

Cohort based approach

- Adaptation of Levenshtein algorithm
- Edit operation cost highest at the start and reduces gradually

m	ɔ	ə	l		k	ə
m	ɛ		l	ə	k	
0	2.4	2.2	0	1.6	0	1

Stress based approach

- Not all stressed syllables clearly marked in GTRP
- Almost all words have initial stress
- Approximation → Edit operation costs higher at first three positions

Results

LOCAL INCOHERENCE (lower is better)

- Slightly increased performance for adapted algorithms

	LOCAL INCOHERENCE
start (log.)	1.91
stress	1.89
regular	1.94

- However, in practice highly comparable results...

Results

LOCAL INCOHERENCE (lower is better)

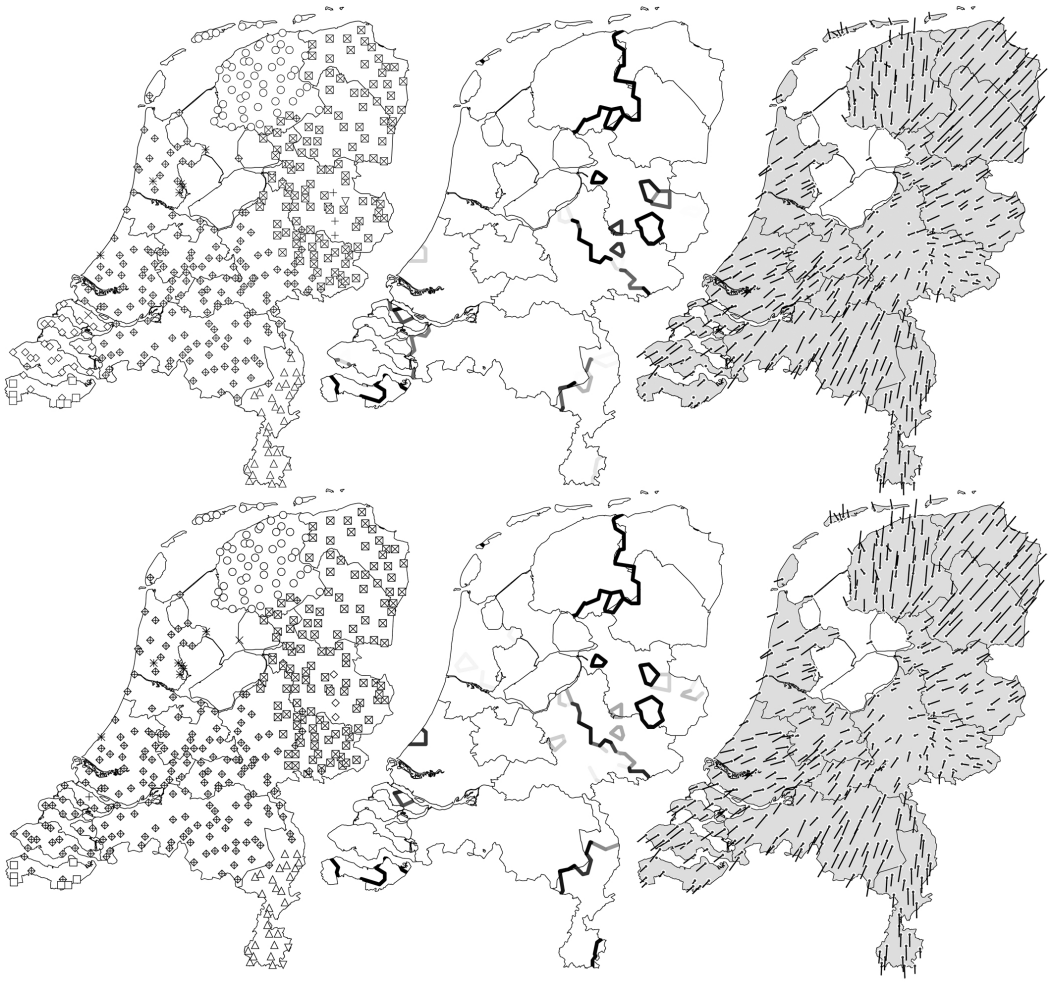
- Slightly increased performance for adapted algorithms

	LOCAL INCOHERENCE
start (log.) stress	1.91 1.89
regular	1.94

- However, in practice highly comparable results...

Levenshtein vs. adapted Levenshtein

$r = 0.95$



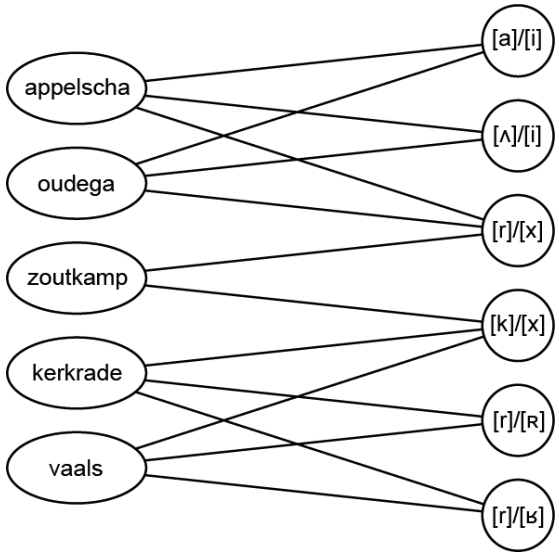
Co-clustering varieties and sound correspondences

- Regular clustering does not yield a linguistic basis (only *post hoc*; Heeringa, 2004)
- New research: Co-clustering to cluster varieties and sound correspondences simultaneously
  - Based on the spectrum of a graph

Generating a bipartite graph from alignments

- A bipartite graph is a graph whose vertices can be divided in two disjoint sets where every edge connects a vertex from one set to a vertex in another set. Vertices within a set are not connected.
- From the alignments, we extract the number of sound correspondences for each variety (compared to a reference site, we use Delft)
- We generated a bipartite graph of varieties v and sound correspondences s
  - There is an edge between v_i and s_j iff freq(s_j in v_i) > 0

Example of a bipartite graph *A*



	[a]/[i]	[ʌ]/[i]	[r]/[x]	[k]/[x]	[r]/[ʀ]	[r]/[ʁ]
Appelscha	1	1	1	0	0	0
Oudega	1	1	1	0	0	0
Zoutkamp	0	0	1	1	0	0
Kerkrade	0	0	0	1	1	1
Appelscha	0	0	0	1	1	1

Co-clustering procedure

- Used by Dhillon (2001) to co-cluster words and documents
- Based on finding the eigenvectors of the adjacency matrix of a bipartite graph and subsequently using the *k*-means algorithm on the eigenvectors to obtain the two-way clustering
  - The mathematical details are not covered in this talk (but see Wieling and Nerbonne, 2009)

Example of co-clustering a biparte graph (1/3)

- Based on the adjacency matrix $\boldsymbol{A}$:

	[a]/[i]	[ʌ]/[i]	[r]/[x]	[k]/[x]	[r]/[ʀ]	[r]/[ʁ]
Appelscha	1	1	1	0	0	0
Oudega	1	1	1	0	0	0
Zoutkamp	0	0	1	1	0	0
Kerkrade	0	0	0	1	1	1
Appelscha	0	0	0	1	1	1

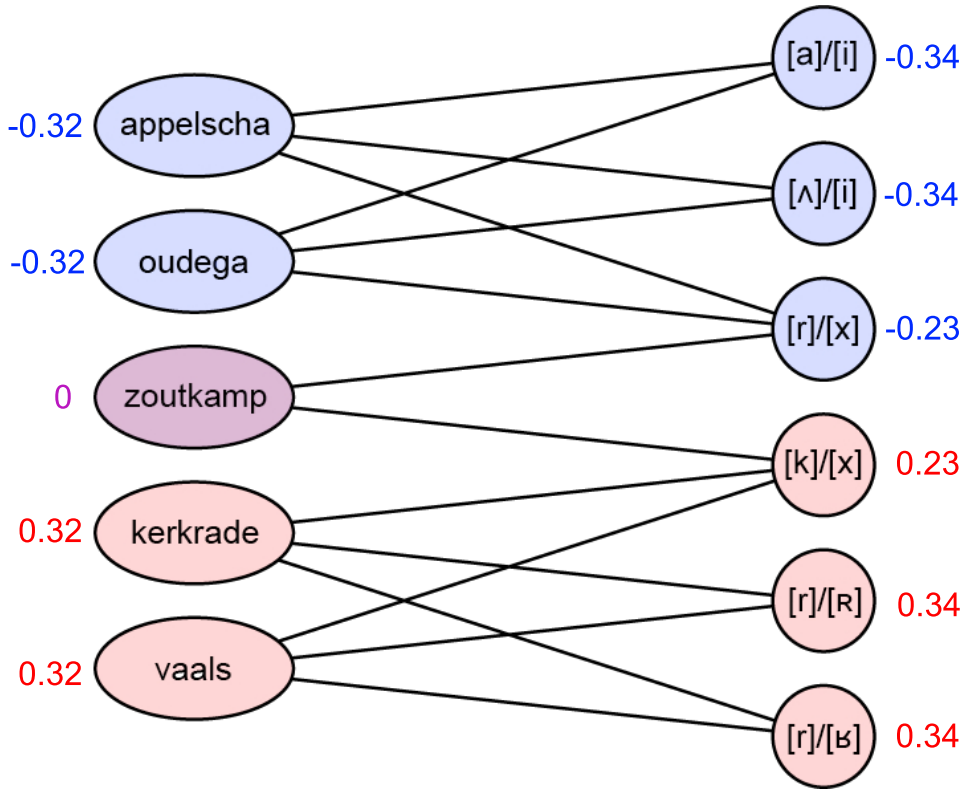
- We can calculate the eigenvectors (of the Laplacian) of $\boldsymbol{A}$:
  $\lambda_2 = .057$, $\boldsymbol{x} = [-.32\ -.32\ 0\ .32\ .32\ -.34\ -.34\ -.23\ .23\ .34\ .34]^T$
  $\lambda_3 = .53$, $\boldsymbol{x} = [.12\ .12\ -.7\ .12\ .12\ .25\ .25\ -.33\ -.33\ .25\ .25]^T$

Example of co-clustering a biparte graph (2/3)

- To cluster in $k = 2$ groups, we use:
  - λ_2 = .057, $\boldsymbol{x}$ = [-.32 -.32 0 .32 .32 -.34 -.34 -.23 .23 .34 .34]^T

Example of co-clustering a biparte graph (2/3)

- To cluster in $k = 2$ groups, we use:
  - λ_2 = .057, $\boldsymbol{x}$ = [-.32 -.32 0 .32 .32 -.34 -.34 -.23 .23 .34 .34]^T
- We obtain the following co-clustering:

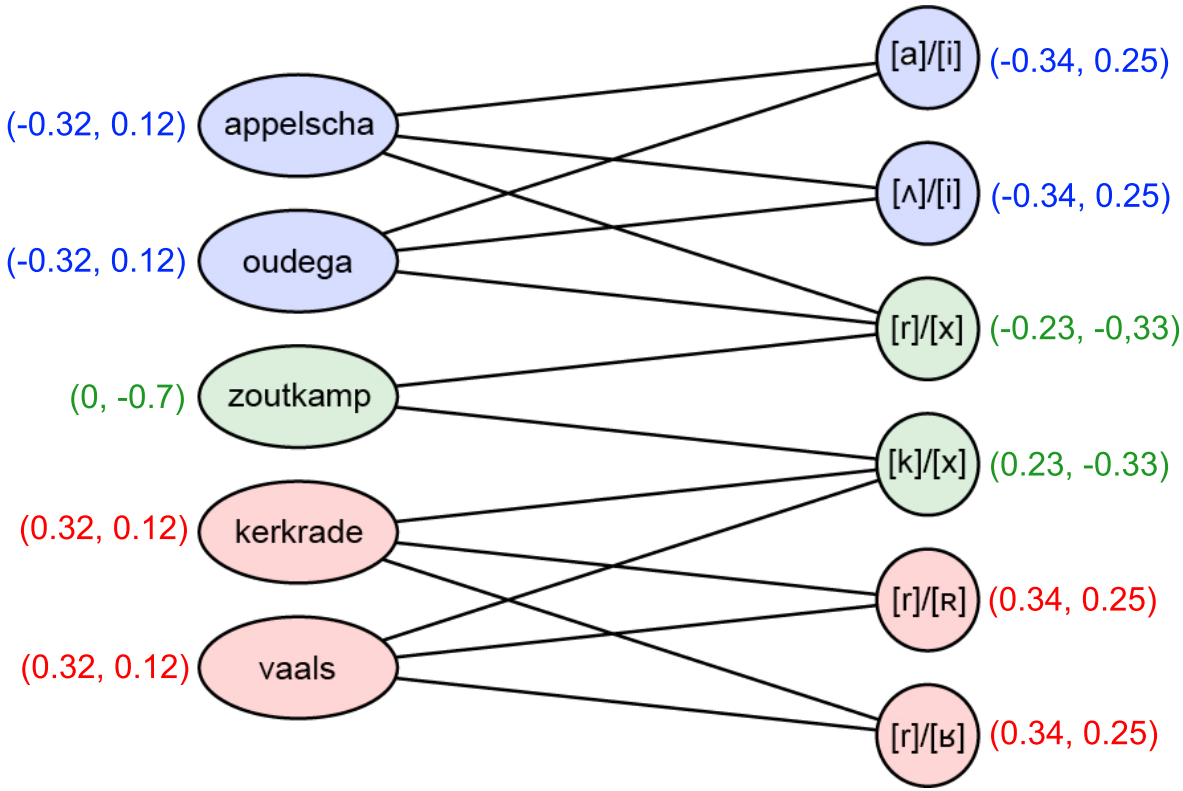


Example of co-clustering a biparte graph (3/3)

- To cluster in $k = 3$ groups, we use:
  - λ_2 = .057, $\boldsymbol{x}$ = [-.32 -.32 0 .32 .32 -.34 -.34 -.23 .23 .34 .34]^T
  - λ_3 = .53, $\boldsymbol{x}$ = [.12 .12 -.7 .12 .12 .25 .25 -.33 -.33 .25 .25]^T

Example of co-clustering a biparte graph (3/3)

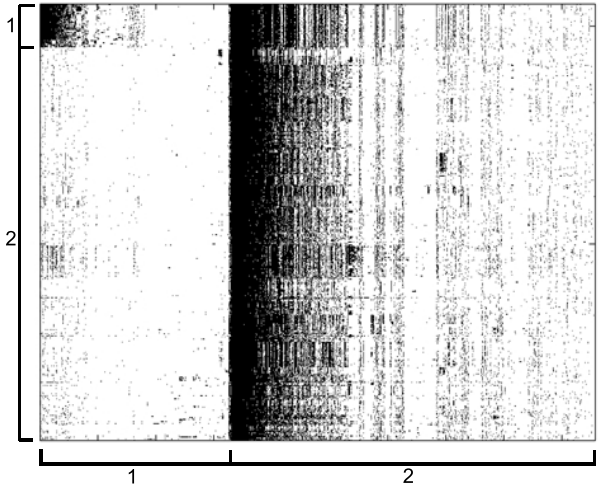
- To cluster in $k = 3$ groups, we use:
  - $\lambda_2 = .057$, $\boldsymbol{x} = [-.32\ -.32\ 0\ .32\ .32\ -.34\ -.34\ -.23\ .23\ .34\ .34]^T$
  - $\lambda_3 = .53$, $\boldsymbol{x} = [.12\ .12\ -.7\ .12\ .12\ .25\ .25\ -.33\ -.33\ .25\ .25]^T$
- We obtain the following co-clustering:

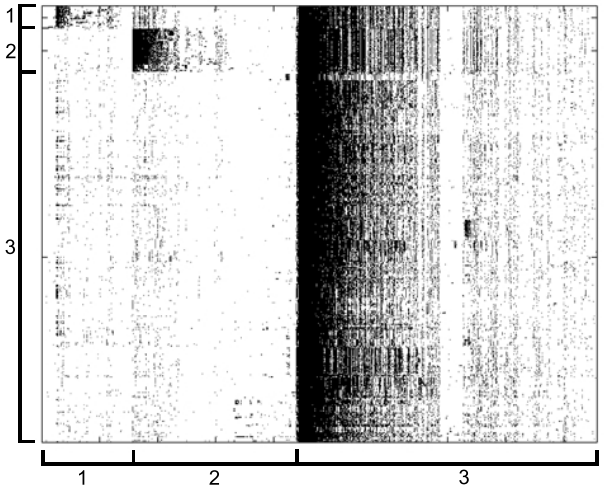


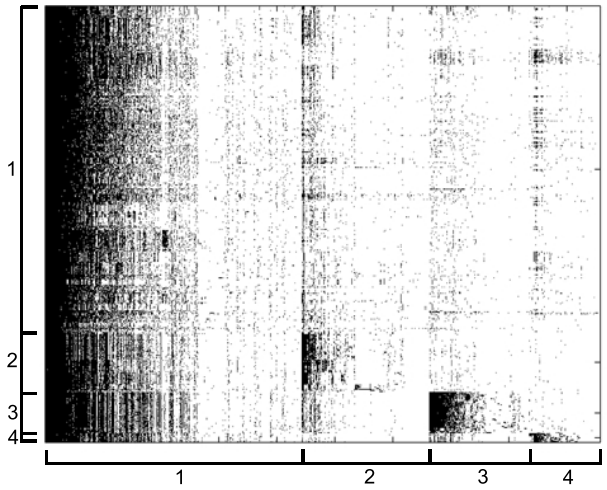
Results using Dutch dialect data

- In the following slides the results using the bipartite spectral graph partitioning method on the Dutch dialect data will be shown.

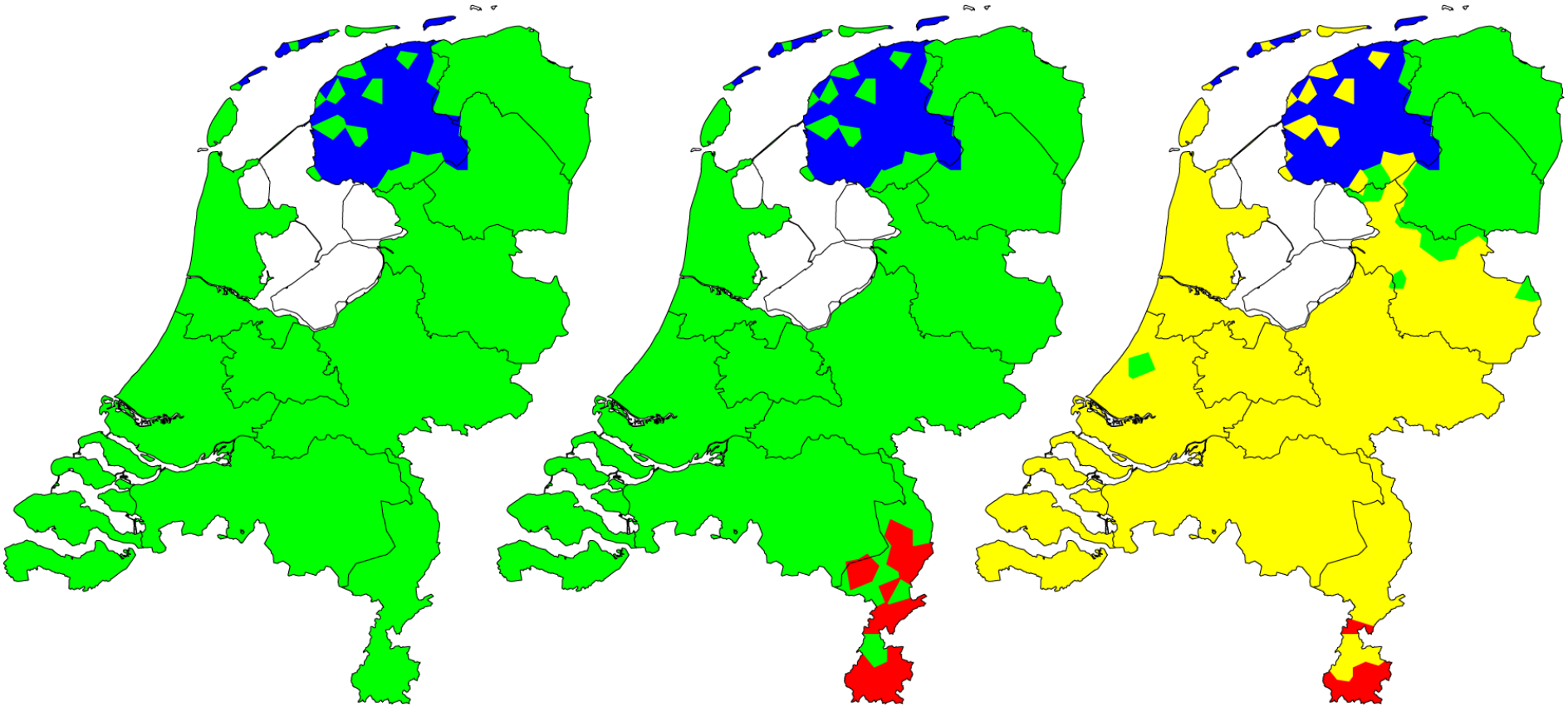
Results: {2,3,4} co-clusters in the data







Results: {2,3,4} clusters of varieties



Results: {2,3,4} clusters of sound correspondences

Red: objectively determined to be in top-10 of most important sound correspondences

- Some sound correspondences specific for the Frisian area

Reference	[ʌ]	[ʌ]	[a]	[o]	[u]	[x]	[x]	[r]
Frisian	[ɪ]	[i]	[i]	[ɛ]	[ɛ]	[j]	[z]	[x]

- Some sound correspondences specific for the Limburg area

Reference	[r]	[r]	[k]	[n]	[n]	[w]
Limburg	[ʀ]	[ʁ]	[x]	[ʀ]	[ʁ]	[f]

- Some sound correspondences specific for the Low Saxon area

Reference	[ə]	[ə]	[ə]	[-]	[a]
Low Saxon	[m]	[ŋ]	[ɴ]	[ʔ]	[e]

Conclusion

- A good way to approach language variation is from the aggregate level
- Methods from computational linguistics are highly suitable to investigate the large amounts of data present at this level
- It is easy to adapt and evaluate these methods to test alternative hypotheses based on, e.g., psycholinguistic research
- The discussed graph-theoretic method is a valuable method as it provides a linguistic basis for the aggregate results

Thank you for listening!

Any questions?