From words to features to trees: Computing a world tree of languages from word lists

Gerhard Jäger

Tübingen University

35th European Summer School in Logic, Language and Information

August 7, 2024

WORDS BONES GENES TOOLS
Tracking Linguistic, Cultural, and Biological Trajectories of the Human Past

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

DFG

erc
European Research Council
Established by the European Commission

Typological distributions

- common practice since Greenberg (1963):
  - collect a sample of languages
  - classify them according to some typological feature

  ⇒ skewed distribution indicates something interesting going on
- Problem: languages are not independent samples
- skewed distribution may reflect
  - skewed diversification rate across families
  - properties of an ancestral bottleneck
- balanced sampling mitigates the first, but not the second problem

Typological distributions

Maslova (2000):

> *"If the A-distribution for a given typology cannot be assumed to be stationary, a distributional universal cannot be discovered on the basis of purely synchronic statistical data."*
>
> *"In this case, the only way to discover a distributional universal is to **estimate transition probabilities** and as it were to 'predict' the stationary distribution on the basis of the equations in (1)."*

Defining models

- feature values evolve according to a *continuous time Markov chain* (CTMC)
- evolution along a phylogeny
- phylogenetic tree is only partially known - represented here as posterior distribution of Bayesian phylogenetic inference from lexical data (from ASJP)

Discrete time Markov chains

Ewens and Grant (2005), 4.5–4.9, 11

Definition

A *discrete time Markov chain* over a countable state space S is a function from $\mathbb{N}$ into random variables X over S with the *Markov property*

$$\mathbb{P}(X_{n+1} = x | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \mathbb{P}(X_{n+1} = x | X_n = x_n)$$

which is *stationary*:

$$\forall m, n : \mathbb{P}(X_{n+1} = x_i | X_n = x_j) = \mathbb{P}(X_{m+1} = x_i | X_m = x_j)$$

Discrete time Markov chains

A dt Markov chain with finite state space is characterized by

- its *initial distribution* X_0, and
- its *transition Matrix* P, where

$$p_{ij} \quad = \quad \mathbb{P}(X_{n+1} = x_j | X_n = x_i)$$

P is a *stochastic matrix*, i.e. $\forall i \sum_j p_{i,j} = 1$.

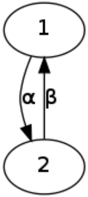
Definition

“Markov(λ, P)” is the dt Markov chain with initial distribution λ and transition matrix P.

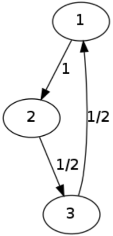
Discrete time Markov chains

Transition matrices over a finite state space can conveniently be represented as weighted graphs.

$$P = \left(\begin{array}{c} 1-\alpha, \alpha \\ \beta, 1-\beta \end{array} \right)$$



$$P = \left(\begin{array}{ccc} 0 & 1 & 0 \\ 0 & 1/2 & 1/2 \\ 1/2 & 0 & 1/2 \end{array} \right)$$



Discrete time Markov chains

- We say $i \to j$ if there is a path (with positive probabilities in each step) from x_i to x_j.
- The symmetric closure of this relation, $i \leftrightarrow j$, is an equivalence relation. It partitions a Markov chain into *communicating classes*.
- A Markov chain is *irreducible* iff it consists of a single communicating class.
- A state x_i is *recurrent* iff
$$\forall n \exists m : \mathbb{P}(X_{n+m} = x_i) > 0$$
- A state is *transient* iff it is not recurrent.

Discrete time Markov chains

- For each communicating class C: Either all of its states are transient or all of its states are recurrent.

Discrete time Markov chains

By convention, we assume that λ is a row vector. The distribution at time n is given by

$$\mathbb{P}(X_t = x_i) = (\lambda P^n)_i$$

Discrete time Markov chains

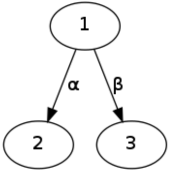
For each stochastic matrix P there is at least one distribution π with

$$\pi P = P$$

(π is a left eigenvector for P.) π is called an **invariant distribution.**

π need not be unique:

$$P = \begin{pmatrix} 1-\alpha-\beta & \alpha & \beta \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$



$\pi = (0, \gamma, \delta)$ is a left eigenvector for P for each $\gamma, \delta \in [0, 1]$.

Discrete time Markov chains

If an irreducible Markov chain converges, then it converges to an invariant distribution:

If $\lim_{n\to\infty} P^n = A$, then

- there is a distribution π with $A_i = \pi$ for all i, and
- π is invariant.

π is called the **equilibrium distribution**. Not every Markov chain has an equilibrium:

$$P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Discrete time Markov chains

Definition

- The **period** k of state x_i is defined as

$$k = \gcd\{n : \mathbb{P}(X_n = i | X_0 = i) > 0\}$$

- A state is **aperiodic** iff its period $= 1$.
- A Markov chain is **aperiodic** iff each of its states is aperiodic.

Theorem

If a finite Markov chain is irreducible and aperiodic, then

- *it has exactly one invariant distribution,* π, *and*
- π *is its equilibrium.*

Discrete time Markov chains

Theorem

If a finite Markov chain is irreducible and aperiodic, with equilibrium distribution π, then

$$\lim_{n\to\infty} \frac{|\{k < n|X_k = x_i\}|}{n} = \pi_i$$

Intuitively: the relative frequency of times spent in a state converges to the equilibrium probability of that state.

Continuous time Markov chains

- If P is the transition matrix of a discrete time Markov process, then so is P^n.
- In other words, P^n give the transition probabilities for a time interval n.
- Generalization:
  - $P(t)$ is transition matrix as a function of time t.
  - For discrete time: $P(t) = P(1)^t$.
  - How can this be generalized to continuous time?

Matrix exponentials

Definition

$$e^A \doteq \sum_{k=0}^{\infty} \frac{A^k}{k!}$$

Some properties:

- $e^0 = I$
- If $AB = BA$, then $e^{A+B} = e^A e^B$
- $e^{nA} = (e^A)^n$
- If Y is invertible, then $e^{YAY^{-1}} = Ye^A Y^{-1}$
- $e^{\mathrm{diag}(x_1,\ldots,x_n)} = \mathrm{diag}(e^{x_1}, \ldots, e^{x_n})$

Continuous time Markov chains

Definition (Q-matrix)

A square matrix Q is a **Q-matrix** or **rate matrix** iff

- $q_{ii} \leq 0$ for all i,
- $q_{ij} \geq 0$ iff $i \neq j$, and
- $\sum_j q_{ij} = 0$ for all i.

Theorem

If P is a stochastic matrix, then there is exactly one Q-matrix Q with

$$e^Q = P.$$

Continuous time Markov chains

Definition

Let Q be a Q-matrix and λ the initial probability distribution. Then

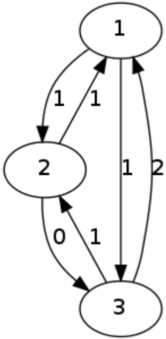
$$X(t) \doteq \lambda e^{tQ}$$

is a **continuous time Markov chain**.

Continuous time Markov chains

Q-matrices can be represented as graphs in the straightforward way (with loops being omitted).

$$Q = \begin{pmatrix} -2 & 1 & 1 \\ 1 & -1 & 0 \\ 2 & 1 & -3 \end{pmatrix}$$



Description in terms of jump chain/holding times

Let Q be a Q-matrix. The corresponding **jump matrix** Π is defined as

$$\pi_{ij} = \begin{cases} -{}^{q_{ij}}/_{q_{ii}} & \text{if } j \neq i \text{ and } q_{ii} \neq 0 \\ 0 & \text{if } j \neq i \text{ and } q_{ii} = 0 \end{cases}$$

$$\pi_{ii} = \begin{cases} 0 & \text{if } q_{ii} \neq 0 \\ 1 & \text{if } q_{ii} = 0 \end{cases}$$

$$Q = \begin{pmatrix} -2 & 1 & 1 \\ 1 & -1 & 0 \\ 2 & 1 & -3 \end{pmatrix} \quad \Pi = \begin{pmatrix} 0 & {}^1/_2 & {}^1/_2 \\ 1 & 0 & 0 \\ {}^2/_3 & {}^1/_3 & 0 \end{pmatrix}$$

Description in terms of jump chain/holding times

Let Q be a Q-matrix and Π the corresponding jump matrix. The Markov process described by $\langle \lambda, Q \rangle$ can be conceived as:

1. Choose an initial state according to distribution λ.
2. If in state i, wait a time t that is exponentially distributed with parameter $-q_{ii}$.
3. Then jump into a new state j chosen according to the distribution $\Pi_{i\cdot}$.
4. Goto 2.

Continuous time Markov chains

Let $M = \langle\lambda, Q\rangle$ be a continuous time Markov chain and Π be the corresponding jump matrix.

- A state is recurrent (transient) for M if it is recurrent (transient) for a discrete time Markov chain with transition matrix Π.
- The communicating classes of M are those defined by Π.
- M is irreducible iff Π is irreducible.

Continuous time Markov chains

Theorem

If Q is irreducible and recurrent. Then there is a unique distribution π with

- $\pi Q = 0$
- $\pi e^{tQ} = \pi$
- $\lim_{t \to \infty} (e^{tQ})_{ij} = \pi_j$

Time reversibility

- Does **not** mean that $a \to b$ and $b \to a$ are equally likely.
- Rather, the condition is

$$\begin{aligned} \pi_a p(t)_{ab} &= \pi_b p(t)_{ba} \\ \pi_a q_{ab} &= \pi_b q_{ba} \end{aligned}$$

- This means that sampling an a from the equilibrium distribution and observe a mutation to b in some interval t is as likely as sampling a b in equilibrium and see it mutate into a after time t.

Two-states model, equal rates

$$Q = \begin{pmatrix} -r & r \\ r & -r \end{pmatrix} \quad P(t) = \tfrac{1}{2}\begin{pmatrix} 1+e^{-2rt} & 1-e^{-2rt} \\ 1-e^{-2rt} & 1+e^{-2rt} \end{pmatrix}$$

$$\pi = (1/2, 1/2)$$

Two-states model, different rates

$$Q = \begin{pmatrix} -r & r \\ s & -s \end{pmatrix} \quad P(t) = \frac{1}{r+s} \begin{pmatrix} s + re^{-(r+s)t} & r - re^{-(r+s)t} \\ s - se^{-(r+s)t} & r + se^{-(r+s)t} \end{pmatrix}$$

$$\pi = \left({}^{s}/_{r+s}, {}^{r}/_{r+s} \right)$$

Two-states model, different rates

- if we measure time in expected number of mutations, we have

$$r + s = 1$$

- therefore:

Two-state model

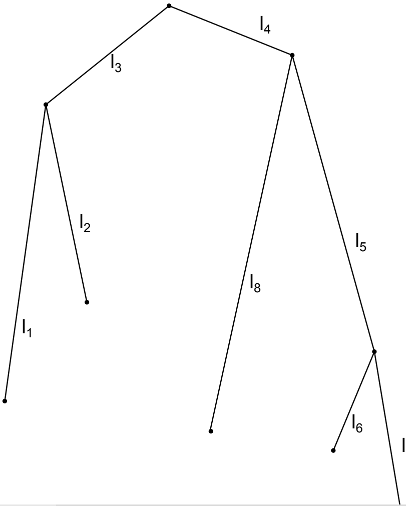
$$Q = \begin{pmatrix} -r & r \\ s & -s \end{pmatrix} \quad P(t) = \begin{pmatrix} s + re^{-t} & r - re^{-t} \\ s - se^{-t} & r + se^{-t} \end{pmatrix}$$

$$\pi = (s, r)$$

The two-state model is always time reversible.

Likelihood of a tree

background reading: Ewens and Grant (2005), 15.7



- simplifying assumption: evolution at different branches is independent
- suppose we know probability distributions v_t and v_b over states at top and bottom of branch l_k
- $\mathcal{L}(l_k) = v_t^T P(l_k) v_b$

Likelihood of a tree

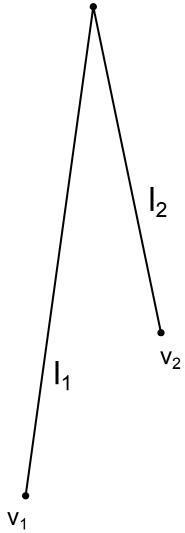
- likelihoods of states $(0, 1)$ at root are

$$v_1^T P(l_1) v_2^T P(l_2)$$

- log-likelihoods

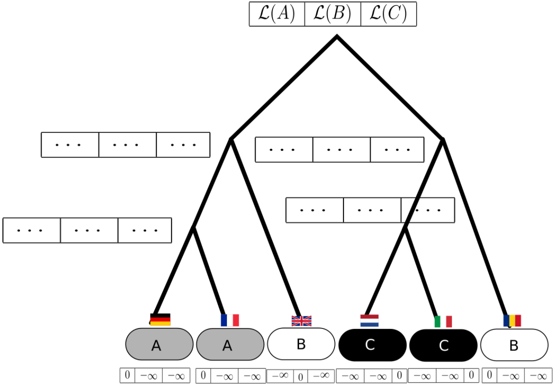
$$\log(v_1^T P(l_1)) + \log(v_2^T P(l_2))$$

- log-likelihood of larger tree: recursively apply this method from tips to root



Likelihood of a tree

$$\mathcal{L}(mother)_i \quad = \prod_{d\in daughters}\sum_{1\leq j\leq n}(P(t)_{i,j}\mathcal{L}(d)_j),$$


$\mathcal{L}(A)$ | $\mathcal{L}(B)$ | $\mathcal{L}(C)$
... | ... | ...
... | ... | ...
... | ... | ...
... | ... | ...
A
A
B
C
C
B
0 | $-\infty$ | $-\infty$
0 | $-\infty$ | $-\infty$
$-\infty$ | 0 | $-\infty$
$-\infty$ | $-\infty$ | 0
$-\infty$ | $-\infty$ | 0
0 | $-\infty$ | $-\infty$

(Log-)Likelihood of a tree

- overall likelihood for entire tree depends on probability distribution on root
- if we assume that root node is in equilibrium:

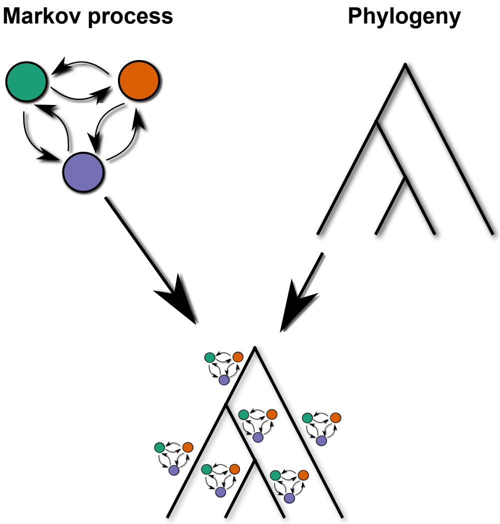
$$\mathcal{L}(\text{tree}) = (s, r)^T \mathcal{L}(\text{root})$$

- does not depend on location of the root (→ time reversibility)
- this is for one character — likelihood for all data is product of likelihoods for each character

(Log-)Likelihood of a tree

- likelihood of tree depends on
  - branch lengths
  - rates for each character
- likelihood for tree *topology:*

$$\mathcal{L}(\text{topology}) = \max_{l_k:\ k \text{ is a branch}} \mathcal{L}(\text{tree}|\vec{l}_k)$$



Markov process

Phylogeny

Figure: Schematic structure of the phylogenetic CTMC model. Independent but identical instances of

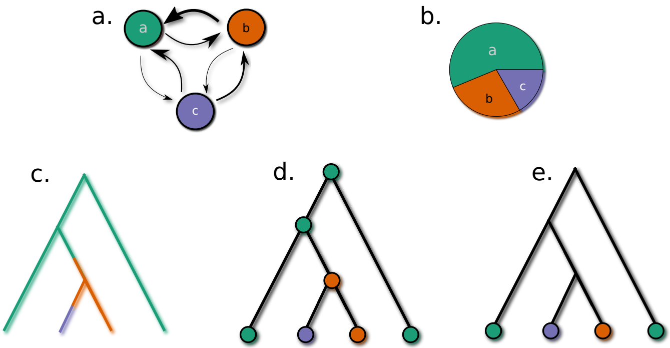


Figure: a. CTMC b. Equilibrium distribution c. Fully specified history of a phylogenetic Markov chain d. Marginalizing over events at branches e. Marginalizing over states at internal nodes

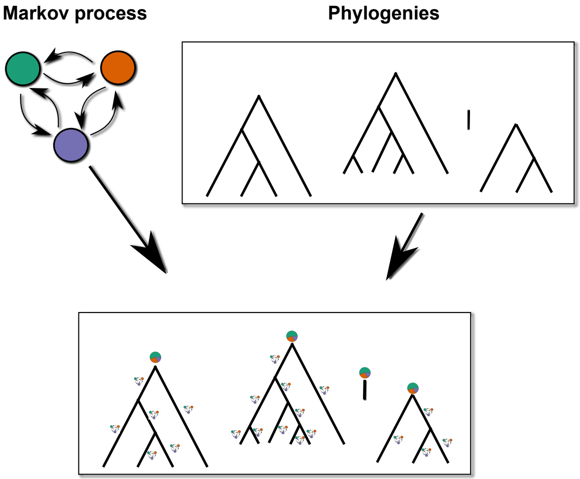
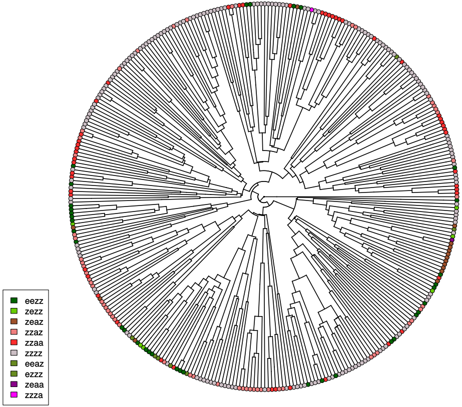


Figure: Phylogenetic Markov CTMC with a collection of phylogenies

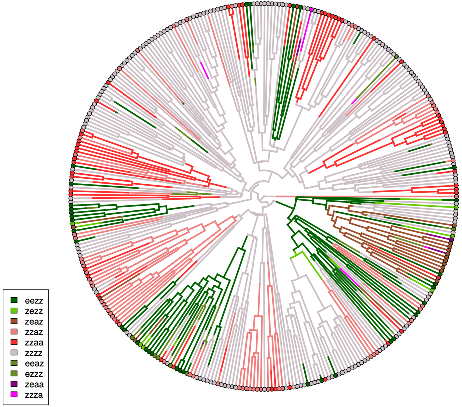
Estimating rates of change

- if phylogeny and states of extant languages are known...



Estimating rates of change

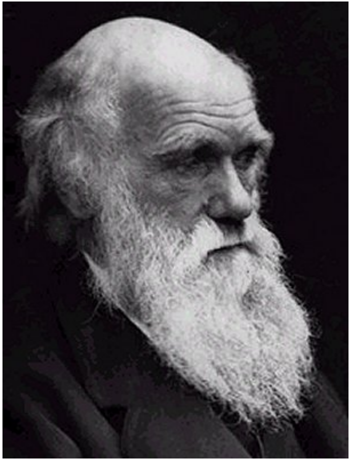
- if phylogeny and states of extant languages are known...
- ... transition rates and ancestral states can be estimated based on Markov model



Language change and evolution

"The formation of different languages and of distinct species, and the proofs that both have been developed through a gradual process, are curiously parallel. [...] We find in distinct languages striking homologies due to community of descent, and analogies due to a similar process of formation. The manner in which certain letters or sounds change when others change is very like correlated growth. [...] The frequent presence of rudiments, both in languages and in species, is still more remarkable. [...]
Languages, like organic beings, can be classed in groups under groups; and they can be classed either naturally according to descent, or artificially by other characters. Dominant languages and dialects spread widely, and lead to the gradual extinction of other tongues."

(Darwin, The Descent of Man)



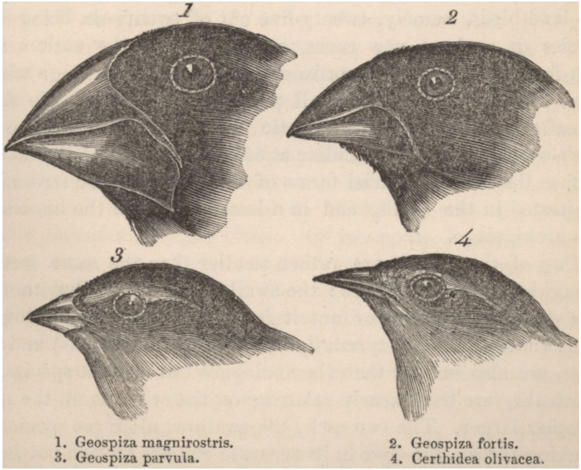
Language change and evolution

Vater Unser im Himmel, geheiligt werde Dein Name

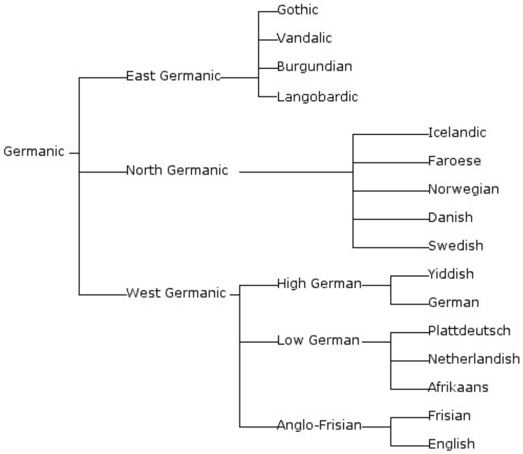
Onze Vader in de Hemel, laat Uw Naam geheiligd worden

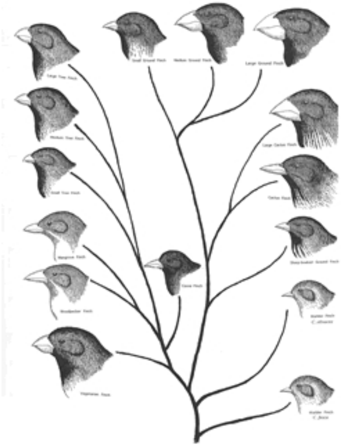
Our Father in heaven, hallowed be your name

Fader Vor, du som er i himlene! Helliget vorde dit navn



Language change and evolution



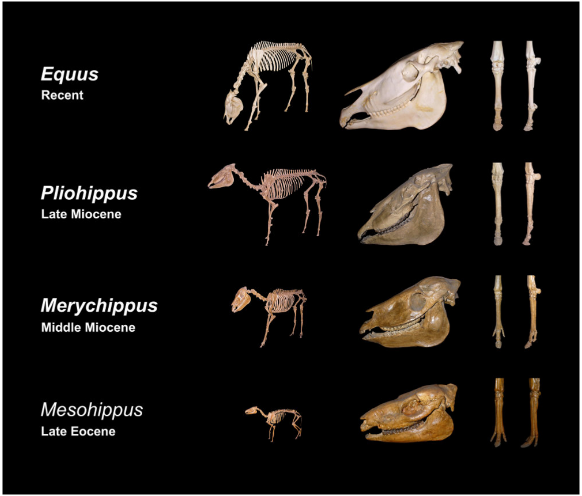


Language change and evolution

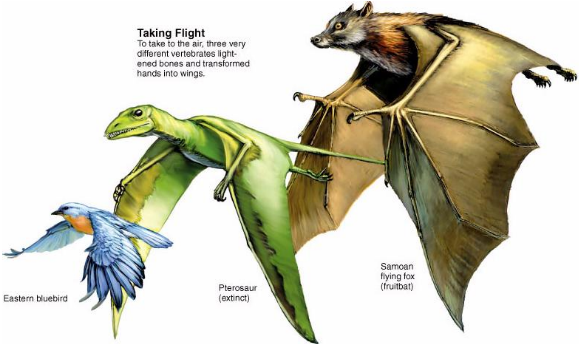
Mittelhochdeutsch:
Got vater unser, dâ du bist in dem himelrîche gewaltic alles des dir ist, geheiliget sô werde dîn nam

Althochdeutsch:
Fater unser thû thâr bist in himile, si giheilagôt thîn namo

Gotisch:
Atta unsar þu in himinam, weihnai namo þein



Convergent evolution

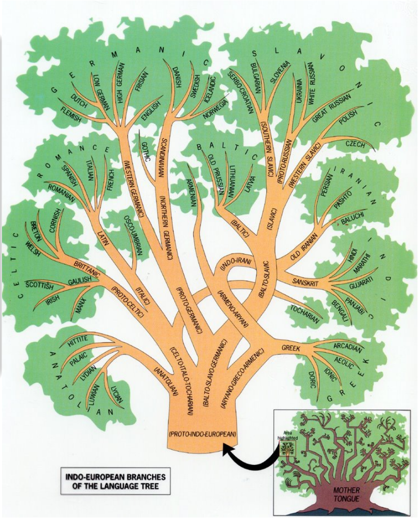

Taking Flight
To take to the air, three very different vertebrates lightened bones and transformed hands into wings.
Eastern bluebird
Pterosaur
(extinct)
Samoan
flying fox
(fruitbat)

- Old English *docga* > English *dog*
- Proto-Paman **gudaga* > Mbabaram *dog* ('dog')

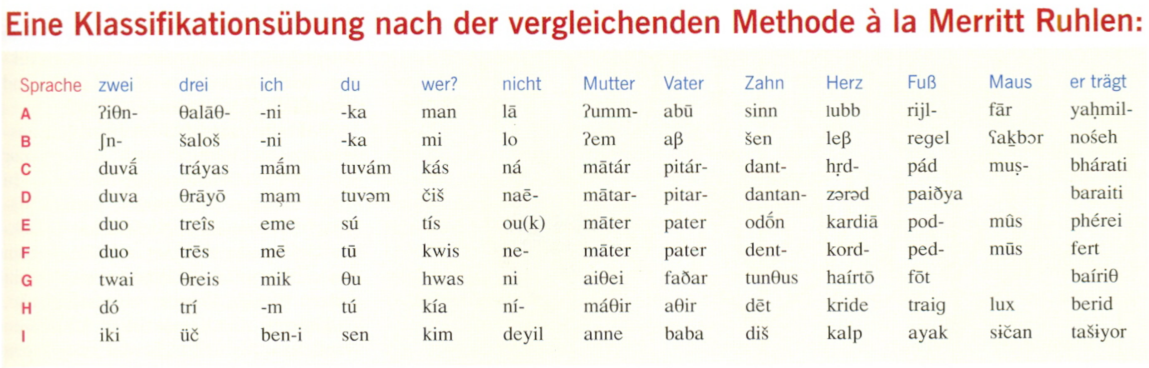
Language phylogeny

Comparative method

1. identifying *cognates*, i.e. obviously related morphemes in different languages, such as *new/nowy*, *two/dwa*, or *water/voda*
2. reconstruction of *common ancestor* and *sound laws* that explain the change from reconstructed to observed forms
3. applying this iteratively leads to phylogenetic language trees



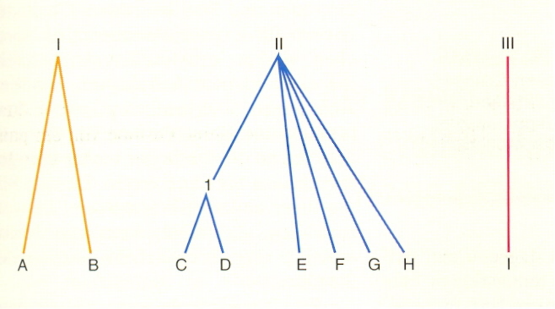
Similarity between languages

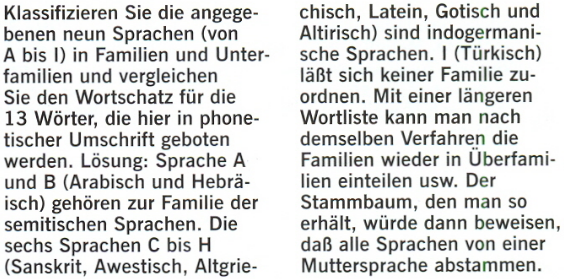


Eine Klassifikationsübung nach der vergleichenden Methode à la Merritt Ruhlen:

Sprache	zwei	drei	ich	du	wer?	nicht	Mutter	Vater	Zahn	Herz	Fuß	Maus	er trägt
A	ʔiθn-	θalāθ-	-ni	-ka	man	lā	ʔumm-	abū	sinn	lubb	rijl-	fār	yaḥmil-
B	ʃn-	šaloš	-ni	-ka	mi	lo	ʔem	aβ	šen	leβ	regel	ʕak̲bɔr	nośeh
C	duvā́	tráyas	mā́m	tuvám	kás	ná	mātár	pitár-	dant-	hṛd-	pád	muṣ-	bhárati
D	duva	θrāyō	mạm	tuvəm	čiš	naē-	mātar-	pitar-	dantan-	zərəd	paiðya		baraiti
E	duo	treîs	eme	sú	tís	ou(k)	māter	pater	odṓn	kardiā	pod-	mûs	phérei
F	duo	trēs	mē	tū	kwis	ne-	māter	pater	dent-	kord-	ped-	mūs	fert
G	twai	θreis	mik	θu	hwas	ni	aiθei	faðar	tunθus	haírtō	fōt		baíriθ
H	dó	trí	-m	tú	kía	ní-	máθir	aθir	dēt	kride	traig	lux	berid
I	iki	üč	ben-i	sen	kim	deyil	anne	baba	diš	kalp	ayak	sıčan	taşıyor

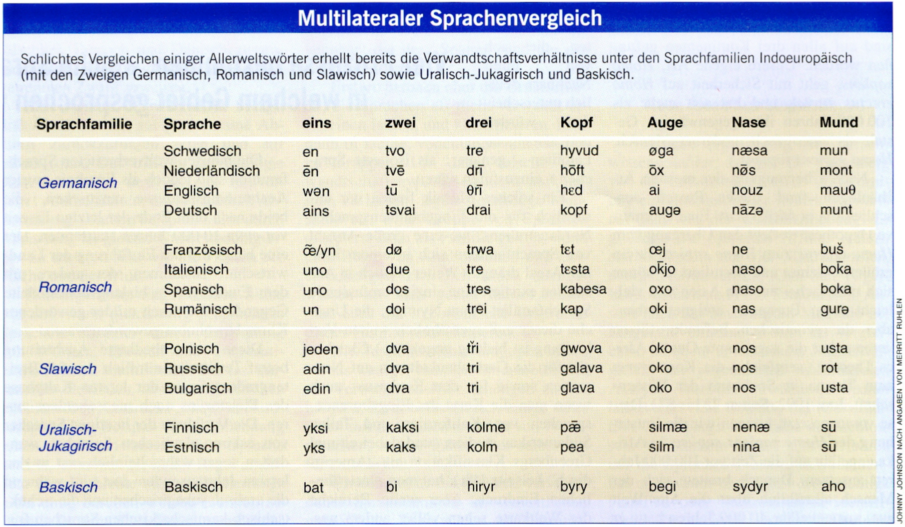
Similarity between languages





Klassifizieren Sie die angegebenen neun Sprachen (von A bis I) in Familien und Unterfamilien und vergleichen Sie den Wortschatz für die 13 Wörter, die hier in phonetischer Umschrift geboten werden. Lösung: Sprache A und B (Arabisch und Hebräisch) gehören zur Familie der semitischen Sprachen. Die sechs Sprachen C bis H (Sanskrit, Awestisch, Altgriechisch, Latein, Gotisch und Altirisch) sind indogermanische Sprachen. I (Türkisch) läßt sich keiner Familie zuordnen. Mit einer längeren Wortliste kann man nach demselben Verfahren die Familien wieder in Überfamilien einteilen usw. Der Stammbaum, den man so erhält, würde dann beweisen, daß alle Sprachen von einer Muttersprache abstammen.

Similarity between languages



Multilateraler Sprachenvergleich

Schlichtes Vergleichen einiger Allerweltswörter erhellt bereits die Verwandtschaftsverhältnisse unter den Sprachfamilien Indoeuropäisch (mit den Zweigen Germanisch, Romanisch und Slawisch) sowie Uralisch-Jukagirisch und Baskisch.

Sprachfamilie	Sprache	eins	zwei	drei	Kopf	Auge	Nase	Mund
Germanisch	Schwedisch	en	tvo	tre	hyvud	ø̄ga	næsa	mun
	Niederländisch	ēn	tvē	drī	hōft	ōx	nø̄s	mont
	Englisch	wən	tū	θrī	hɛd	ai	nouz	mauθ
	Deutsch	ains	tsvai	drai	kopf	augə	nāzə	munt
Romanisch	Französisch	œ̃/yn	dø	trwa	tɛt	œj	ne	buš
	Italienisch	uno	due	tre	tɛsta	ok̄jo	naso	bok̄a
	Spanisch	uno	dos	tres	kabesa	oxo	naso	boka
	Rumänisch	un	doi	trei	kap	oki	nas	gurə
Slawisch	Polnisch	jeden	dva	tři	gwova	oko	nos	usta
	Russisch	adin	dva	tri	galava	oko	nos	rot
	Bulgarisch	edin	dva	tri	glava	oko	nos	usta
Uralisch-Jukagirisch	Finnisch	yksi	kaksi	kolme	pǣ	silmæ	nenæ	sū
	Estnisch	yks	kaks	kolm	pea	silm	nina	sū
Baskisch	Baskisch	bat	bi	hiryr	byry	begi	sydyr	aho

JOHNNY JOHNSON NACH ANGABEN VON MERRITT RUHLEN

Language phylogeny

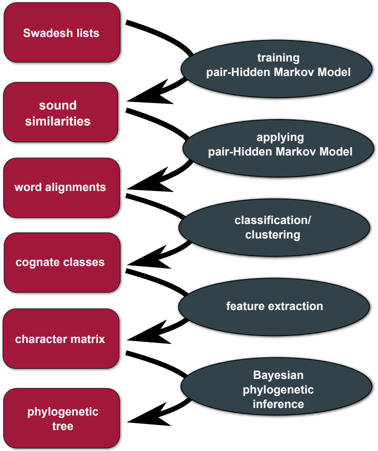
Scope of the method

- reconstructed vocabulary shrinks with growing time depth
- maximal time horizon seems to be about 8,000 years
- grammatical morphemes and categories arguably more stable and less apt to borrowing
- problem here: limited number of features, cross-linguistic variation constrained by language universals, frequently convergent evolution
- comparative method is hard to apply in regions with high linguistic diversity and without written documents (Paleo-America, Papua)
- tree structure might be inappropriate if there is a significant effect of language contact (cf. Australia)

Computational Methods

- both cognate detection and tree construction lend themselves to algorithmic implementation
- Advantages:
  - easy to scale up
  - comparability of results
  - affords statistical evaluation
- Disadvantages:
  - cognacy judgments require lots of linguistic insight and experience
  - tree construction should be subject to historical (including archeological) and geographical plausibility

From words to trees



Swadesh lists

training pair-Hidden Markov Model

sound similarities

applying pair-Hidden Markov Model

word alignments

classification/ clustering

cognate classes

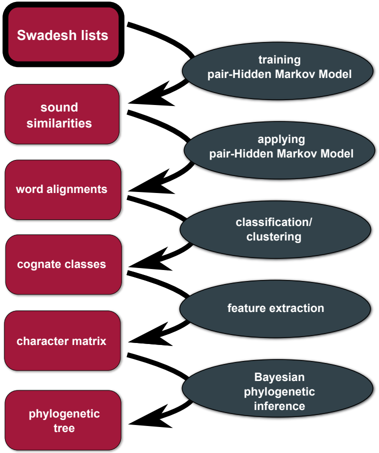
feature extraction

character matrix

Bayesian phylogenetic inference

phylogenetic tree

From words to trees



Swadesh lists

training
pair-Hidden Markov Model

sound
similarities

applying
pair-Hidden Markov Model

word alignments

classification/
clustering

cognate classes

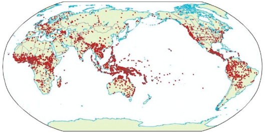
feature extraction

character matrix

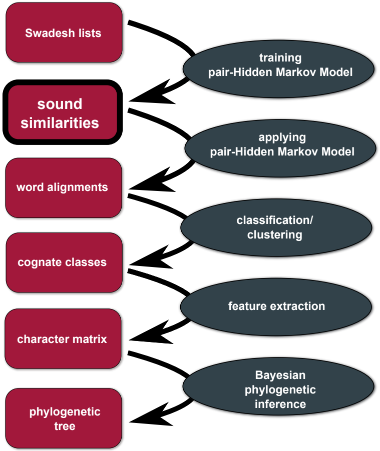
Bayesian
phylogenetic
inference

phylogenetic
tree

concept	Latin	English
I	ego	Ei
you	tu	yu
we	nos	wi
one	unus	w3n
two	duo	tu
person	persona, homo	pers3n
fish	piskis	fiS
dog	kanis	dag
louse	pedikulus	laus
tree	arbor	tri
leaf	foly∼u*	lif
skin	kutis	skin
blood	saNgw∼is	bl3d
bone	os	bon
horn	kornu	horn
ear	auris	ir
eye	okulus	Ei



From words to trees



Swadesh lists

training pair-Hidden Markov Model

sound similarities

applying pair-Hidden Markov Model

word alignments

classification/ clustering

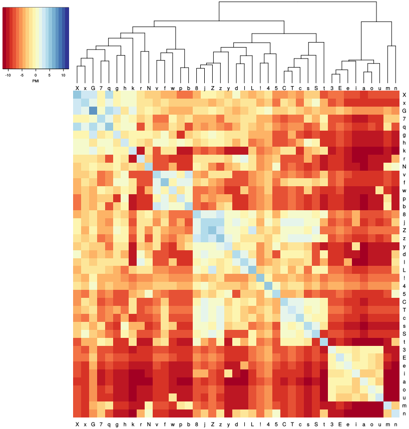
cognate classes

feature extraction

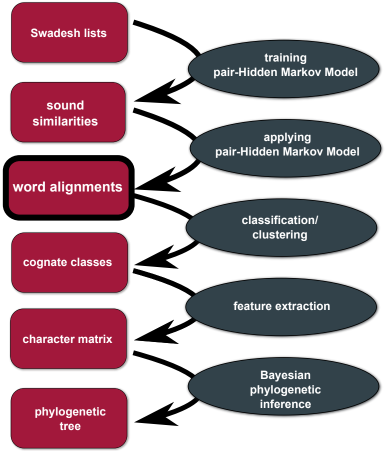
character matrix

Bayesian phylogenetic inference

phylogenetic tree



From words to trees



Swadesh lists

training pair-Hidden Markov Model

sound similarities

applying pair-Hidden Markov Model

word alignments

classification/ clustering

cognate classes

feature extraction

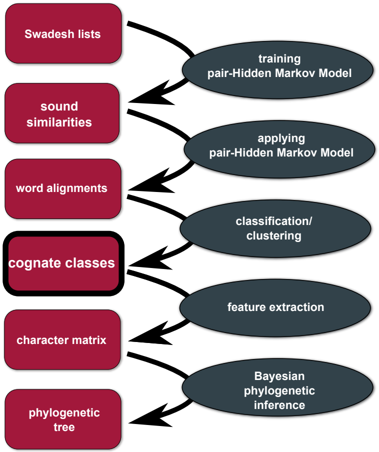
character matrix

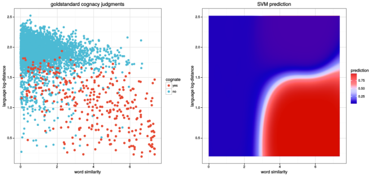
Bayesian phylogenetic inference

phylogenetic tree

Language	*fish:z*	*tongue:1*	*smoke:1*
Abui-Atangmelang	-af-u		
Abui-Fuimelang	-af-u	tal-i-fi--	
Adang	aab--	tal-E-b---	awai--b-a-n-o-7o-
Blagar-Bakalang	-ab--	--j-e-bur-	--ad--b-a-n-aNka-
Blagar-Bama	aab--	teg-e-bur-	------b-e-n-a-xa-
Blagar-Kulijahi	-ab--	tej-e-bur-	------b-e-n-aNka-
Blagar-Nule	aab--	tej-e-bur-	--ad--b-e-n-aNka-
Blagar-Tuntuli	aab--	tej-e-bur-	a-adgeb-a-n-a-q--
Blagar-Warsalelang	-ab--	tel-e-bur-	a-ad--b-a-n-a-x--
Bunaq			------b-o-t-o-h--
Deing	haf--		------buu-n------
Hamap	7ab--	nar-ø-buN-	------b-a-n-o-7--
Kabola	hab--	tal-e-b---	awal--b-e-n-e-7o-
Kaera-Padangsul	-ab--	talee-b---	a-ad--b-e-naa-x--
Kafoa	-afUi	tal-i-p---	------f-o-n-a----
Kamang	-ap-i	nal---pu--	------p-u-n----a-
Kiraman	-Eb--	nal-i-bar-	--ar--b-a-n-o-kan
Klon	-eb-i	gel-E-b---	--ed-ab-o-n------
Kui	-eb--	tal-i-ber-	--ar--b-o-n-o-k--
Kula	-ap-i	-il-I-p---	------p---n-ekka-
Nedebang	aaf-i	gel-e-fu--	--ar-ab-u-n------
Reta	aab--	nal-e-bul-	a-ad--b-o-n-a----
Sar-Adiabang	haf--	--p-e-fal-	--ar--buu-n------
Sar-Nule	haf--	nal-e-faj-	
Sawila	-ap-i	gal-impuru	------p-u-n-a-ka-
Teiwa-Madar	xaf--	gel-i-vi--	------buu-n------
Wersing	-ap-i	nej-e-bur-	--ad-ap-u-n-a-k--
Wpantar	hap--	nal-e-bu--	------b-unn-a----

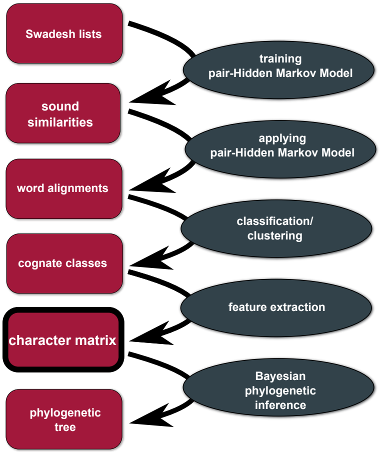
From words to trees

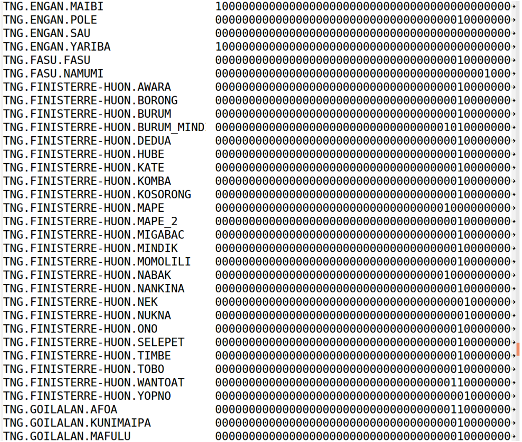

Swadesh lists
training pair-Hidden Markov Model
sound similarities
applying pair-Hidden Markov Model
word alignments
classification/ clustering
cognate classes
feature extraction
character matrix
Bayesian phylogenetic inference
phylogenetic tree



	English	Spanish	Modern Greek	Standard German
I	Ei:A	yo:B	exo:C	iX:D
you	yu:A	ustet:B, tu:C	esi:D	du:E
we	wi:A	nosotros:B	emis:C	vir:A
one	w3n:A	uno:B	enas:C, ena:C	ains:D
two	tu:A	dos:B	8y∼o:C, 8io:D	cvai:E
person	pers3n:A	persona:A	an8∼ropos:B	mEnS:C
fish	fiS:A	peskado:A, pes:A	psari:B	fiS:A
dog	dag:A	pero:B	sTili:C, sTilos:C	hunt:D
come	k3m:A	veni:B	erx∼o:C	kh∼om3n:A
sun	s3n:A	sol:B	ily∼os:C, iLos:C	zon3:A
star	star:A	estreya:A	asteri:A, astro:A	StErn:A
water	wat3r:A	agw∼a:B	nero:C	vas3r:A
stone	ston:A	piedra:B	petra:B	Stain:A
fire	fEir:A	fuego:B	foty∼a:C	foia:D
path	pE8:A	senda:B	8romos:C	pf∼at:A, vek:D
mountain	maunt3n:A	sero:B, monta5a:A	vuno:C, oros:D	bErk:E
full	ful:A	yeno:B	yematos:C, pliris:D	fol:A
new	nu:A	nuevo:A	neos:A, Tenury∼os:B	noi:A
name	nem:A	nombre:A	onoma:A	nam3:A

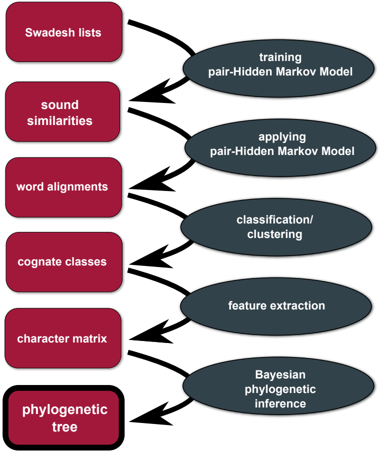
From words to trees





```
TNG.ENGAN.MAIBI                 10000000000000000000000000000000000000000000
TNG.ENGAN.POLE                  00000000000000000000000000000000000010000000
TNG.ENGAN.SAU                   00000000000000000000000000000000000000000000
TNG.ENGAN.YARIBA                10000000000000000000000000000000000000000000
TNG.FASU.FASU                   00000000000000000000000000000000000010000000
TNG.FASU.NAMUMI                 00000000000000000000000000000000000000001000
TNG.FINISTERRE-HUON.AWARA       00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.BORONG      00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.BURUM       00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.BURUM_MIND  00000000000000000000000000000000001010000000
TNG.FINISTERRE-HUON.DEDUA       00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.HUBE        00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.KATE        00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.KOMBA       00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.KOSORONG    00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.MAPE        00000000000000000000000000000000001000000000
TNG.FINISTERRE-HUON.MAPE_2      00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.MIGABAC     00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.MINDIK      00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.MOMOLILI    00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.NABAK       00000000000000000000000000000000001000000000
TNG.FINISTERRE-HUON.NANKINA     00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.NEK         00000000000000000000000000000000000001000000
TNG.FINISTERRE-HUON.NUKNA       00000000000000000000000000000000000001000000
TNG.FINISTERRE-HUON.ONO         00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.SELEPET     00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.TIMBE       00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.TOBO        00000000000000000000000000000000000010000000
TNG.FINISTERRE-HUON.WANTOAT     00000000000000000000000000000000000110000000
TNG.FINISTERRE-HUON.YOPNO       00000000000000000000000000000000000001000000
TNG.GOILALAN.AFOA               00000000000000000000000000000000000110000000
TNG.GOILALAN.KUNIMAIPA          00000000000000000000000000000000000010000000
TNG.GOILALAN.MAFULU             00000000000000000000000000000000000010000000
```

From words to trees



Swadesh lists

training
pair-Hidden Markov Model

sound
similarities

applying
pair-Hidden Markov Model

word alignments

classification/
clustering

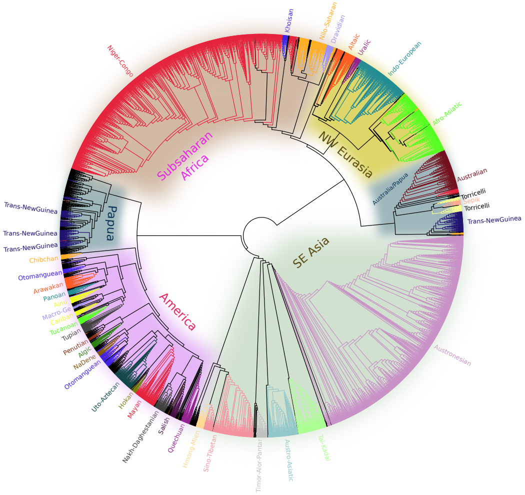
cognate classes

feature extraction

character matrix

Bayesian
phylogenetic
inference

phylogenetic
tree



From word lists to distances

The Automated Similarity Judgment Program

- Project at MPI EVA in Leipzig around Søren Wichmann
- covers more than 6,000 languages and dialects
- basic vocabulary of 40 words for each language, in uniform phonetic transcription
- freely available

used concepts: *I, you, we, one, two, person, fish, dog, louse, tree, leaf, skin, blood, bone, horn, ear, eye, nose, tooth, tongue, knee, hand, breast, liver, drink, see, hear, die, come, sun, star, water, stone, fire, path, mountain, night, full, new, name*

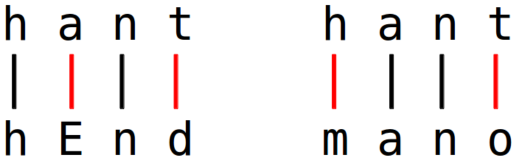
Automated Similarity Judgment Project

concept	Latin	English
I	ego	Ei
you	tu	yu
we	nos	wi
one	unus	w3n
two	duo	tu
person	persona, homo	pers3n
fish	piskis	fiS
dog	kanis	dag
louse	pedikulus	laus
tree	arbor	tri
leaf	foly∼u*	lif
skin	kutis	skin
blood	saNgw∼is	bl3d
bone	os	bon
horn	kornu	horn
ear	auris	ir
eye	okulus	Ei

concept	Latin	English
nose	nasus	nos
tooth	dens	tu8
tongue	liNgw∼E	t3N
knee	genu	ni
hand	manus	hEnd
breast	pektus, mama	brest
liver	yekur	liv3r
drink	bibere	drink
see	widere	si
hear	audire	hir
die	mori	dEi
come	wenire	k3m
sun	sol	s3n
star	stela	star
water	akw∼a	wat3r
stone	lapis	ston
fire	iNnis	fEir

Word distances

- based on string *alignment*
- baseline: Levenshtein alignment ⇒ count matches and mis-matches

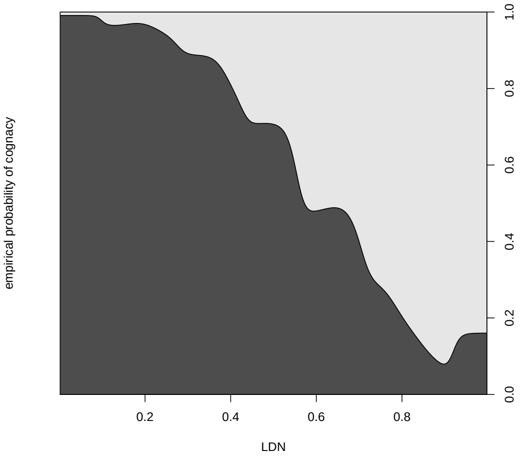


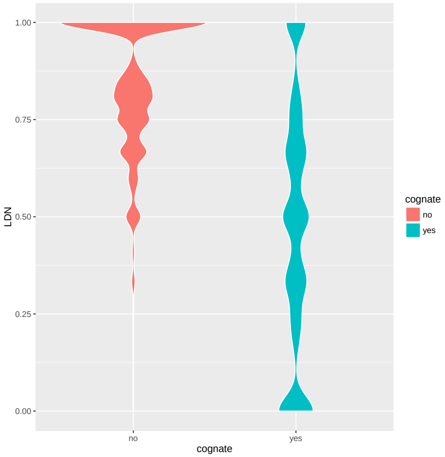
h a n t
h E n d

h a n t
m a n o

- too crude as it totally ignores sound correspondences

How well does normalized Levenshtein distance predict cognacy?





Problems

- binary distinction: match vs. non-match
- frequently genuin sound correspondences in cognates are missed:

c	v	a	i
-	-	**t**	u

n	a	**z**	3
n	-	o	**s**

-	-	-	**f**	i	S
p	i	s	k	i	s

- corresponding sounds count as mismatches even if they are aligend correctly

h	a	n	t
h	E	n	d

h	a	n	t
m	a	n	o

- substantial amount of chance similarities

Capturing sound correspondences

- weighted alignment using **P**ointwise **M**utual **I**nformation (PMI, a.k.a. *log-odds*):

$$s(a,b) = \log \frac{p(a,b)}{q(a)q(b)}$$

  - $p(a,b)$: probability of sound a being etymologically related to sound b in a pair of cognates
  - $q(a)$: relative frequency of sound a
- **Needleman-Wunsch algorithm:** given a matrix of pairwise PMI scores between individual symbols and two strings, it returns the alignment that maximizes the aggregate PMI score
- but first we need to estimate $p(a,b)$ and $q(a), q(b)$ for all soundclasses a and b
- $q(a)$: relative frequency of occurence of segment a in all words in ASJP
- $p(a,b)$: that's a bit more complicated...

Substitution matrix for the ASJP data

1. identify large sample of pairs of closely related languages (using expert information or heuristics based on aggregated Levenshtein distance)

```
An.NORTHERN_PHILIPPINES.CENTRAL_BONTOC
An.MESO-PHILIPPINE.NORTHERN_SORSOGON

WF.WESTERN_FLY.IAMEGA
WF.WESTERN_FLY.GAMAEWE

Pan.PANOAN.KASHIBO_BAJO_AGUAYTIA
Pan.PANOAN.KASHIBO_SAN_ALEJANDRO

AA.EASTERN_CUSHITIC.KAMBAATA_2
AA.EASTERN_CUSHITIC.HADIYYA_2

ST.BAI.QILIQIAO_BAI_2
ST.BAI.YUNLONG_BAI

An.SULAWESI.MANDAR
An.OCEANIC.RAGA

An.SULAWESI.TANETE
An.SAMA-BAJAW.BOEPINANG_BAJAU

An.SOUTHERN_PHILIPPINES.KAGAYANEN
An.NORTHERN_PHILIPPINES.LIMOS_KALINGA

An.MESO-PHILIPPINE.CANIPAAN_PALAWAN
An.NORTHWEST_MALAYO-POLYNESIAN.LAHANAN

NC.BANTOID.LIFONGA
NC.BANTOID.BOMBOMA_2

IE.INDIC.WAD_PAGGA
IE.INDIC.TALAGANG_HINDKO

NC.BANTOID.LINGALA
NC.BANTOID.LIFONGA

An.CENTRAL_MALAYO-POLYNESIAN.BALILEDO
An.CENTRAL_MALAYO-POLYNESIAN.PALUE

AuA.MUNDA.HO
AuA.MUNDA.KORKU
```

Substitution matrix for the ASJP data

2. pick a concept and a pair of related languages at random
   - languages: `Pen.MAIDUAN.MAIDU_KONKAU`, `Pen.MAIDUAN.NE_MAIDU`
   - concept: *one*
3. find corresponding words from the two languages:
   - `nisam`, `niSem`
4. do Levenshtein alignment

n	i	s	a	m
n	i	S	e	m

5. for each sound pair, count number of correspondences
   - nn: 1; ii: 1; sS; 1; ae: 1; mm: 1

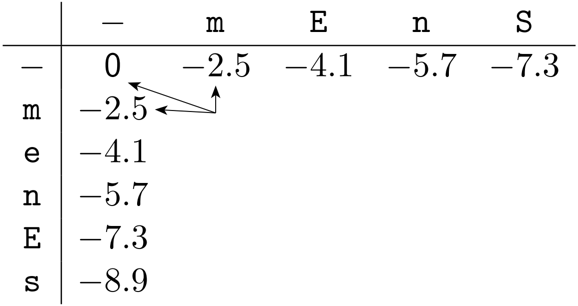
Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5				
e	−4.1				
n	−5.7				
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming



	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5				
e	−4.1				
n	−5.7				
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	-2.5	-4.1	-5.7	-7.3
m	-2.5				
e	-4.1				
n	-5.7				
E	-7.3				
s	-8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	-2.5	-4.1	-5.7	-7.3
m	-2.5	4.13			
e	-4.1				
n	-5.7				
E	-7.3				
s	-8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	-2.5	-4.1	-5.7	-7.3
m	-2.5	4.13			
e	-4.1				
n	-5.7				
E	-7.3				
s	-8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	-2.5	-4.1	-5.7	-7.3
m	-2.5	4.13			
e	-4.1				
n	-5.7				
E	-7.3				
s	-8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53		
e	−4.1				
n	−5.7				
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	-2.5	-4.1	-5.7	-7.3
m	-2.5	4.13	1.53	0.03	
e	-4.1				
n	-5.7				
E	-7.3				
s	-8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1				
n	−5.7				
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53			
n	−5.7				
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65		
n	−5.7				
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	-2.5	-4.1	-5.7	-7.3
m	-2.5	4.13	1.53	0.03	-1.47
e	-4.1	1.53	5.65	3.05	
n	-5.7				
E	-7.3				
s	-8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7				
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03			
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05		
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	−	m	E	n	S
−	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3				
s	−8.9				

Finding the best alignment

- Dynamic Programming

	—	m	E	n	S
—	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3	−1.47			
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	-2.5	-4.1	-5.7	-7.3
m	-2.5	4.13	1.53	0.03	-1.47
e	-4.1	1.53	5.65	3.05	1.55
n	-5.7	0.03	3.05	9.2	6.6
E	-7.3	-1.47	4.75		
s	-8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3	−1.47	4.75	6.6	
s	−8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	-2.5	-4.1	-5.7	-7.3
m	-2.5	4.13	1.53	0.03	-1.47
e	-4.1	1.53	5.65	3.05	1.55
n	-5.7	0.03	3.05	9.2	6.6
E	-7.3	-1.47	4.75	6.6	7.62
s	-8.9				

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3	−1.47	4.75	6.6	7.62
s	−8.9	−2.97			

Finding the best alignment

- Dynamic Programming

	—	m	E	n	S
—	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3	−1.47	4.75	6.6	7.62
s	−8.9	−2.97	2.15		

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	-2.5	-4.1	-5.7	-7.3
m	-2.5	4.13	1.53	0.03	-1.47
e	-4.1	1.53	5.65	3.05	1.55
n	-5.7	0.03	3.05	9.2	6.6
E	-7.3	-1.47	4.75	6.6	7.62
s	-8.9	-2.97	2.15	5.1	

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3	−1.47	4.75	6.6	7.62
s	−8.9	−2.97	2.15	5.1	8.84

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3	−1.47	4.75	6.6	7.62
s	−8.9	−2.97	2.15	5.1	↖ 8.84

- memorizing in each step which of the three cells to the left and above gave rise to the current entry lets us recover the corresponing optimal alignment

Finding the best alignment

- Dynamic Programming

	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3	−1.47	4.75	6.6	7.62
s	−8.9	−2.97	2.15	5.1	8.84

- memorizing in each step which of the three cells to the left and above gave rise to the current entry lets us recover the corresponing optimal alignment

Finding the best alignment

- Dynamic Programming

	—	m	E	n	S
—	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	↖ 9.2	6.6
E	−7.3	−1.47	4.75	6.6 ↑	7.62
s	−8.9	−2.97	2.15	5.1	↖ 8.84

- memorizing in each step which of the three cells to the left and above gave rise to the current entry lets us recover the corresponing optimal alignment

Finding the best alignment

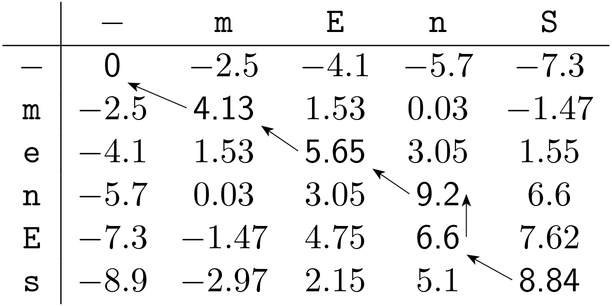
- Dynamic Programming

	—	m	E	n	S
—	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3	−1.47	4.75	6.6	7.62
s	−8.9	−2.97	2.15	5.1	8.84

- memorizing in each step which of the three cells to the left and above gave rise to the current entry lets us recover the corresponing optimal alignment

Finding the best alignment

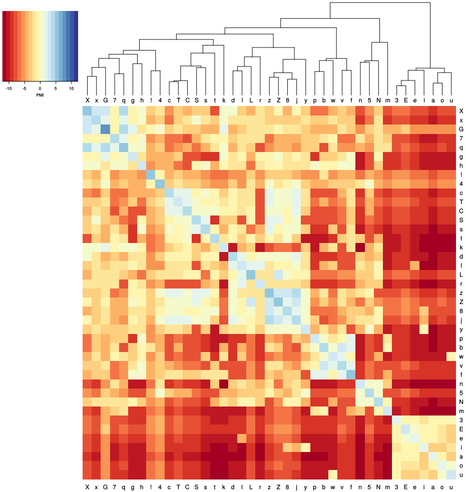
- Dynamic Programming



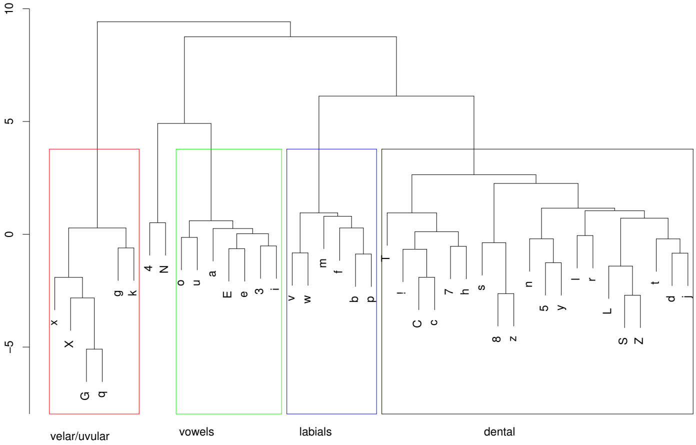
	–	m	E	n	S
–	0	−2.5	−4.1	−5.7	−7.3
m	−2.5	4.13	1.53	0.03	−1.47
e	−4.1	1.53	5.65	3.05	1.55
n	−5.7	0.03	3.05	9.2	6.6
E	−7.3	−1.47	4.75	6.6	7.62
s	−8.9	−2.97	2.15	5.1	8.84

- memorizing in each step which of the three cells to the left and above gave rise to the current entry lets us recover the corresponing optimal alignment

Evaluation

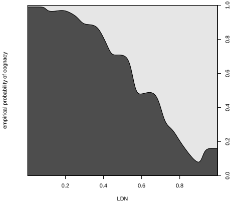

-10
-5
0
5
10
PMI
X x G 7 q g h ! 4 c T C S s t k d l L r z Z 8 j y p b w v f n 5 N m 3 E e i a o u
X x G 7 q g h ! 4 c T C S s t k d l L r z Z 8 j y p b w v f n 5 N m 3 E e i a o u
X x G 7 q g h ! 4 c T C S s t k d l L r z Z 8 j y p b w v f n 5 N m 3 E e i a o u

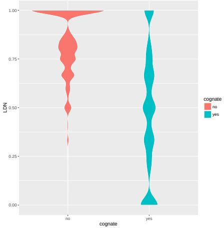
Evaluation

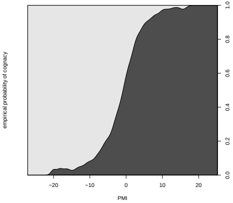

10
5
0
−5
x
X
G
q
g
k
4
N
o
u
a
E
e
3
i
v
w
m
f
b
p
T
!
C
c
7
h
s
8
z
n
5
y
l
r
L
S
Z
t
d
j
velar/uvular
vowels
labials
dental

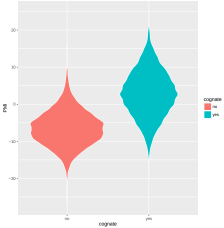
How well does PMI similarity predict cognacy?

expert cognacy judgments used as gold standard









Calibrated PMI similarity

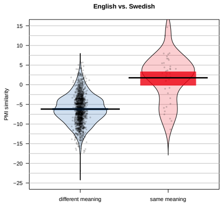
English / Swedish

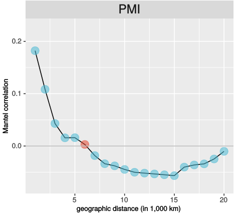
	Ei	**yu**	**wi**	**w3n**	**tu**	**fiS**	. . .
yog	**−7.77**	0.75	−7.68	−7.90	−8.57	−10.50	
du	−7.62	**0.33**	−5.71	−7.41	2.66	−8.57	
vi	−2.72	−2.83	**4.04**	−1.34	−6.45	0.70	
et	−5.47	−7.87	−5.47	**−6.43**	−1.83	−4.70	
tvo	−7.91	−4.27	−3.64	−4.57	**0.39**	−6.98	
fisk	−7.45	−11.2	−3.07	−9.97	−8.66	**7.58**	
⋮							

- values along diagonal give similarity between candidates for cognacy (possibility of meaning change is disregarded)
- values off diagonal provide sample of similarity distribution between non-cognates

Calibrated PMI similarity

- let s be the PMI-similarity between the English and Swedish word for concept c
- **calibrated string similarity**: $-\log($probability that random word pairs are more similar than $s)$
- **language similarity:** average word similarity for all concepts





Cognate clustering

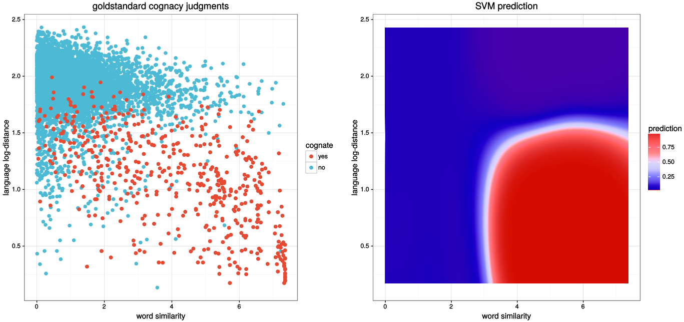
Cognate clustering

- clustering of ASJP strings into *automatically inferred cognate classes* (Jäger and Sofroniev, 2016; Jäger et al., 2017) (take “cognate” with a grain of salt)
- supervised learning, based on expert cognacy judgments as goldstandard
- sources (only the 40 ASJP concepts were used)

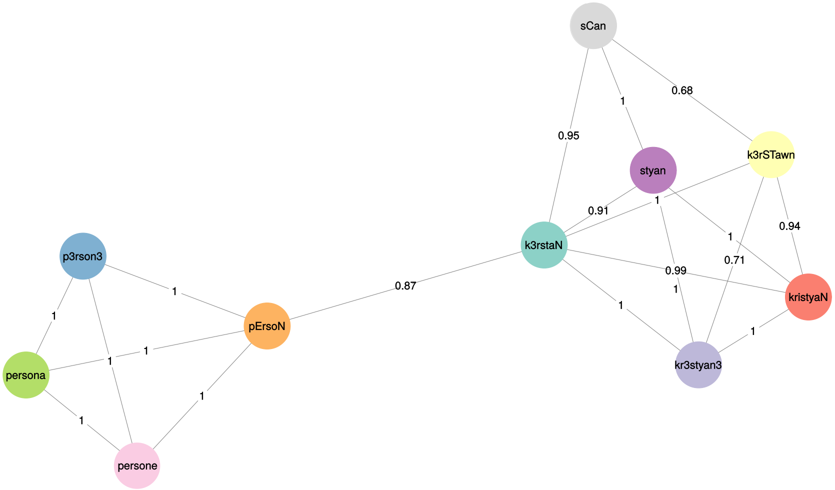
Dataset	Source	Words	Concepts	Languages	Families	Cognate classes
ABVD	Greenhill et al. (2008)	2,306	34	100	Austronesian	409
Afrasian	Militarev (2000)	770	39	21	Afro-Asiatic	351
Chinese	Běijng Dàxué (1964)	422	20	18	Sino-Tibetan	126
Huon	McElhanon (1967)	441	32	14	Trans-New Guinea	183
IELex	Dunn (2012)	2,089	40	52	Indo-European	318
Japanese	Hattori (1973)	387	39	10	Japonic	74
Kadai	Peiros (1998)	399	40	12	Tai-Kadai	102
Kamasau	Sanders and Sanders (1980)	270	36	8	Torricelli	59
Mayan	Brown et al. (2008)	1,113	40	30	Mayan	241
Miao-Yao	Peiros (1998)	206	36	6	Hmong-Mien	69
Mixe-Zoque	Cysouw et al. (2006)	355	39	10	Mixe-Zoque	79
Mon-Khmer	Peiros (1998)	579	40	16	Austroasiatic	232
ObUgrian	Zhivlov (2011)	769	39	21	Uralic	68
total		10,106	40	318	13	2,311

Cognate clustering

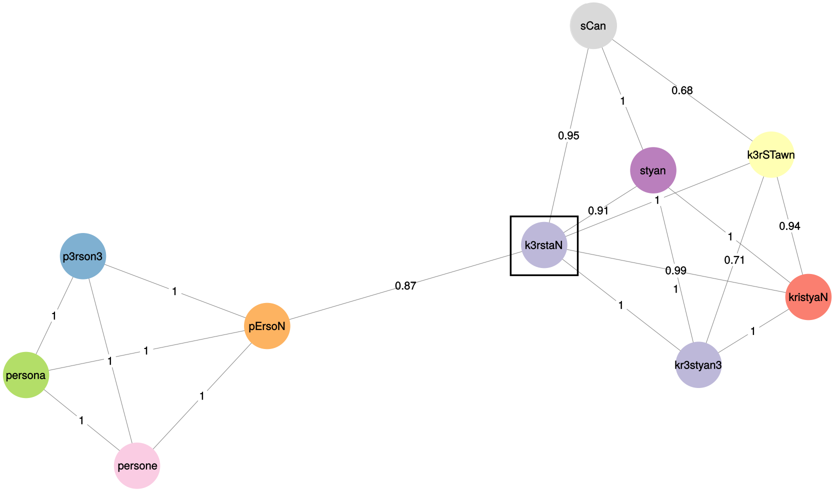
- calibrated word similarity and language similarity were used as predictors to train a *Support Vector Machine* → probability of being cognate for each pair of synonymous ASJP entries
- *Label Propagation* (Raghavan et al., 2007) for clustering
- 0.84 B-cubed F-score with cross-validation on goldstandard data



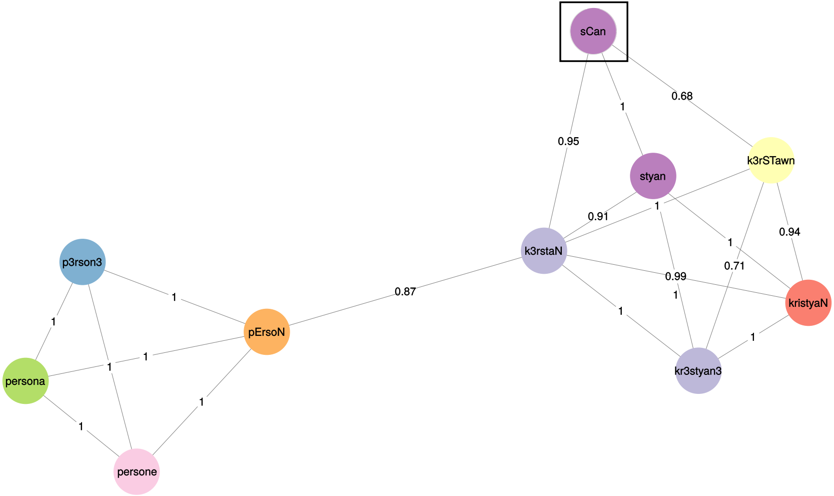
Clustering via Label Propagation



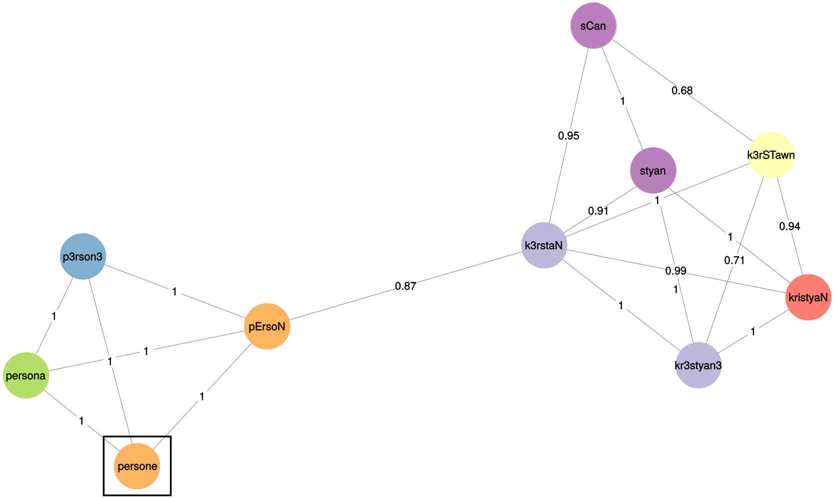
Clustering via Label Propagation



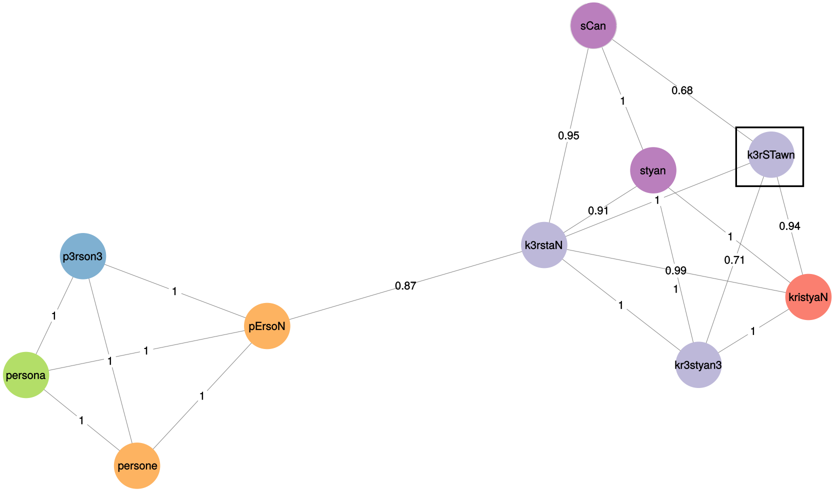
Clustering via Label Propagation



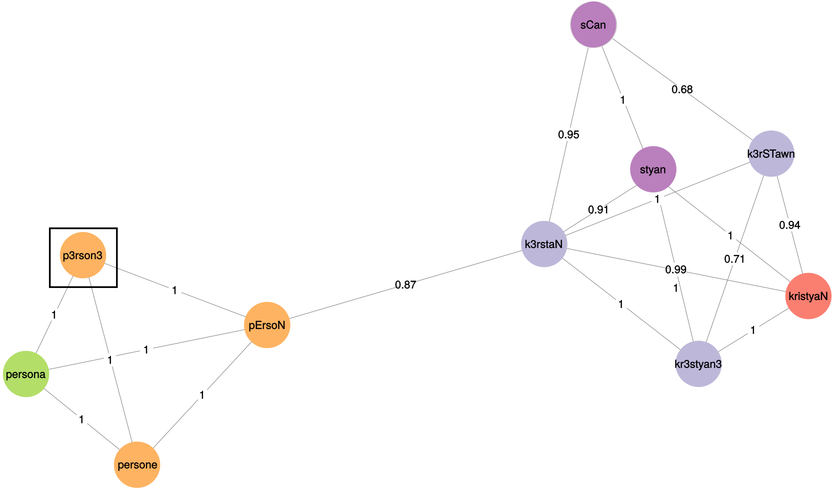
Clustering via Label Propagation



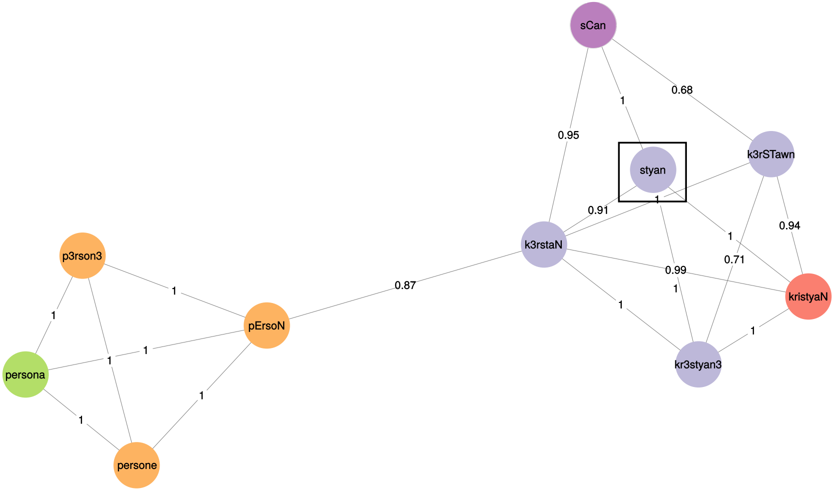
Clustering via Label Propagation



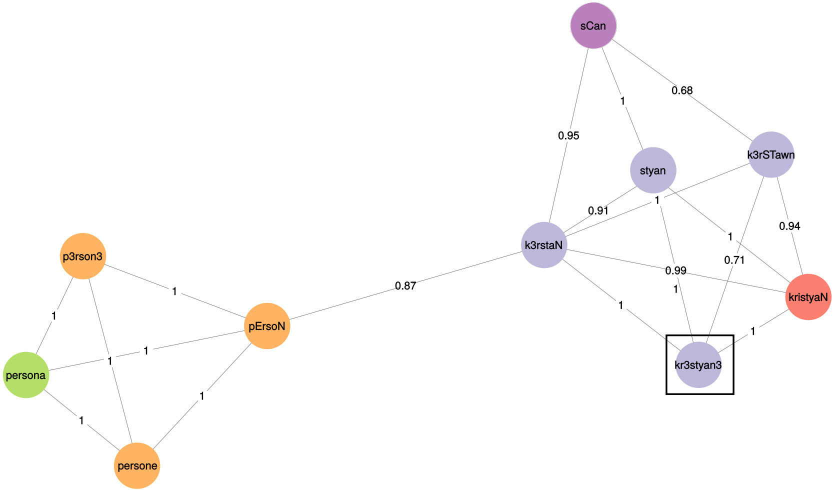
Clustering via Label Propagation



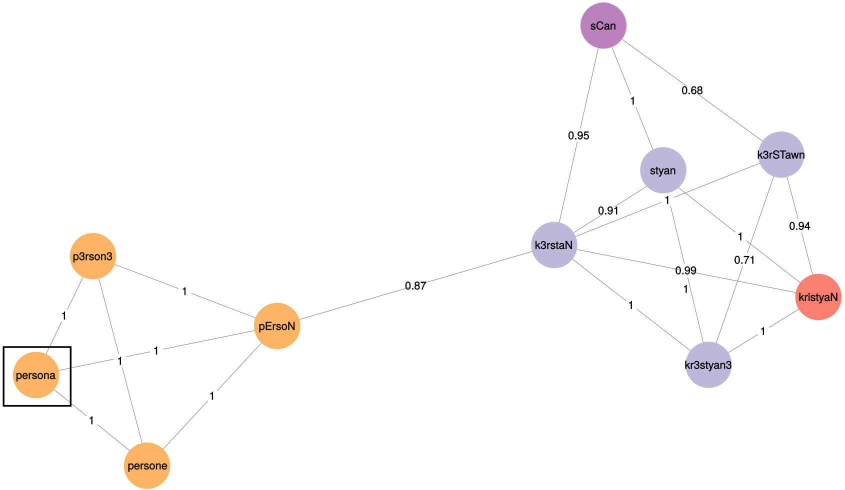
Clustering via Label Propagation



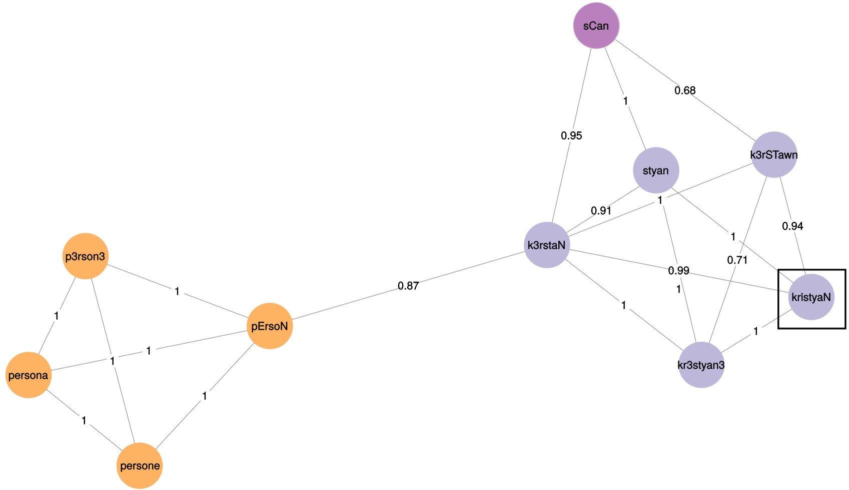
Clustering via Label Propagation



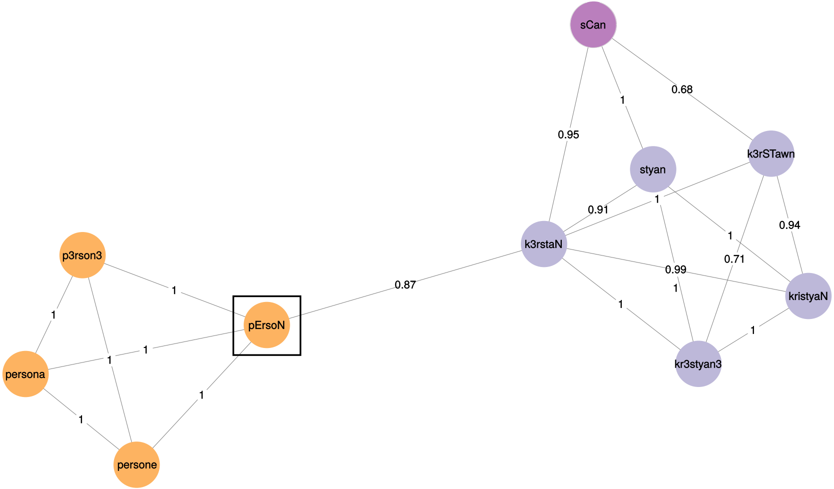
Clustering via Label Propagation



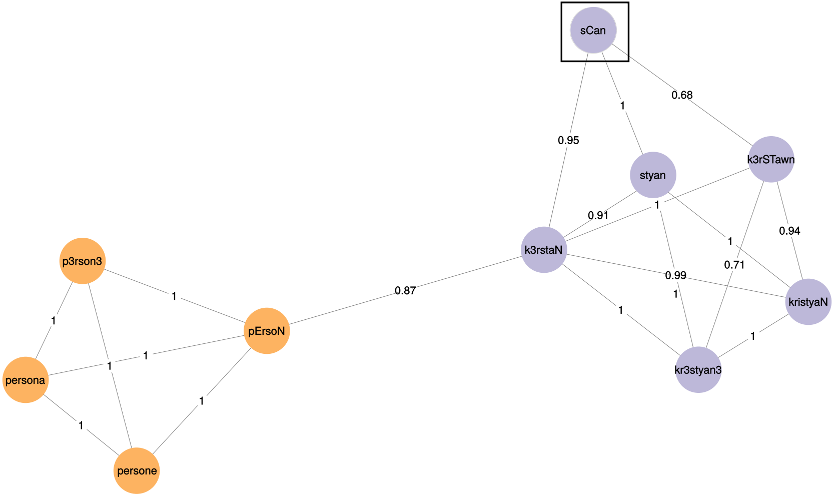
Clustering via Label Propagation



Clustering via Label Propagation



Clustering via Label Propagation



Cognate clustering

doculect	word	class label
ALBANIAN	vet3	0
ALBANIAN_TOSK	vEt3	0
ARAGONESE	ombre	1
ITALIAN_GROSSETO_TUSCAN	omo	2
ROMANIAN_MEGLENO	wom	2
VLACH	omu	2
ASTURIAN	persona	3
BALEAR_CATALAN	p3rson3	3
CATALAN	p3rson3	3
FRIULIAN	pErsoN	3
ITALIAN	persona	3
SPANISH	persona	3
VALENCIAN	persone	3
CORSICAN	nimu	4
DALMATIAN	om	5
EMILIANO_CARPIGIANO	om	5
ROMANIAN_2	om	5
TURIA_AROMANIAN	om	5
EMILIANO_FERRARESE	styan	6
LIGURIAN_STELLA	kristyaN	6
NEAPOLITAN_CALABRESE	kr3styan3	6
ROMAGNOL_RAVENNATE	sCan	6
ROMANSH_GRISHUN	k3rSTawn	6
ROMANSH_SURMIRAN	k3rstaN	6
GALICIAN	ome	7
GASCON	omi	7
PIEMONTESE_VERCELLESE	omaN	8
ROMANSH_VALLADER	uman	8
ALBANIAN_GHEG	5eri	9
SARDINIAN_CAMPIDANESE	omini	9
SARDINIAN_LOGUDARESE	omine	9

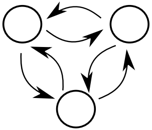
Cognate clustering

concept	doculect	glot_fam	transcription
eye	DORASQUE	Chibchan	oko
eye	NORTHERN_LOW_SAXON	Indo-European	ok
eye	NORTH_FRISIAN_AMRUM	Indo-European	uk
eye	STELLINGWERFS	Indo-European	ok
eye	ASSAMESE	Indo-European	soku
eye	CHAKMA_UnnamedInSource	Indo-European	sog
eye	DALMATIAN	Indo-European	vaklo
eye	FRIULIAN	Indo-European	voli
eye	ITALIAN	Indo-European	okkyo
eye	ITALIAN_GROSSETO_TUSCAN	Indo-European	okyo
eye	JUDEO_ESPAGNOL	Indo-European	oxo
eye	LATIN	Indo-European	okulus
eye	NEAPOLITAN_CALABRESE	Indo-European	woky3
eye	ROMANIAN_2	Indo-European	oky
eye	ROMANIAN_MEGLENO	Indo-European	wokLu
eye	SARDINIAN	Indo-European	ogu
eye	SARDINIAN_CAMPIDANESE	Indo-European	oxu
eye	SARDINIAN_LOGUDARESE	Indo-European	okru
eye	SICILIAN_UnnamedInSource	Indo-European	okiu
eye	SPANISH	Indo-European	oho
eye	TURIA_AROMANIAN	Indo-European	okLu
eye	VLACH	Indo-European	okklu
eye	BELARUSIAN	Indo-European	voka
eye	BOSNIAN	Indo-European	oko
eye	BULGARIAN	Indo-European	oko
eye	CROATIAN	Indo-European	oko
eye	CZECH	Indo-European	oko
eye	KASHUBIAN	Indo-European	wokwo
eye	LOWER_SORBIAN	Indo-European	voko
eye	LOWER_SORBIAN_2	Indo-European	woko
eye	MACEDONIAN	Indo-European	oko
eye	OLD_CHURCH_SLAVONIC	Indo-European	oko
eye	POLISH	Indo-European	oko
eye	SERBOCROATIAN	Indo-European	oko
eye	SLOVAK	Indo-European	oko
eye	SLOVENIAN	Indo-European	oko
eye	UKRAINIAN	Indo-European	oko
eye	UPPER_SORBIAN	Indo-European	voCko
eye	UPPER_SORBIAN	Indo-European	voko
eye	BAINOUK_GUNYAAMOLO	Atlantic-Congo	g3li
eye	USINO	Nuclear_Trans_New_Guinea	ogo

Phylogenetic inference

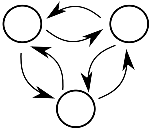
Modeling language change

Markov process

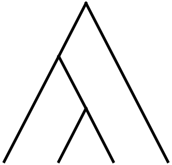


Modeling language change

Markov process

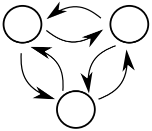


Phylogeny

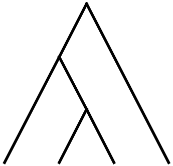


Modeling language change

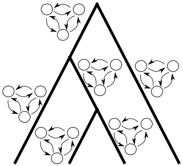
Markov process



Phylogeny

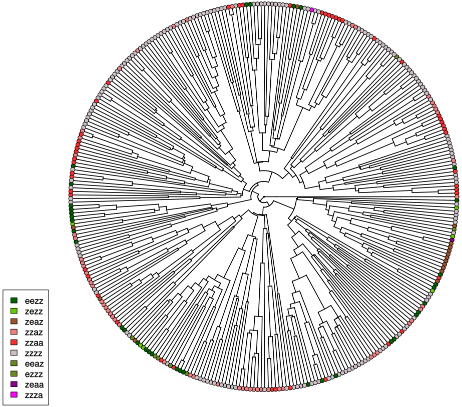


Branching process



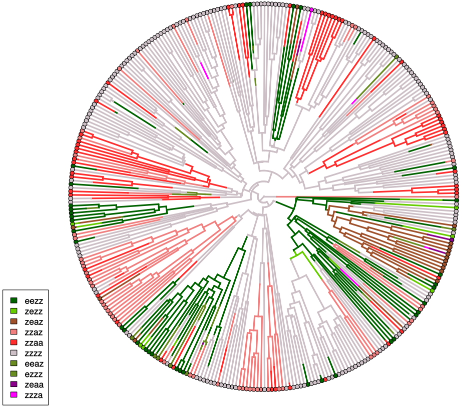
Estimating rates of change

- if phylogeny and states of extant languages are known...



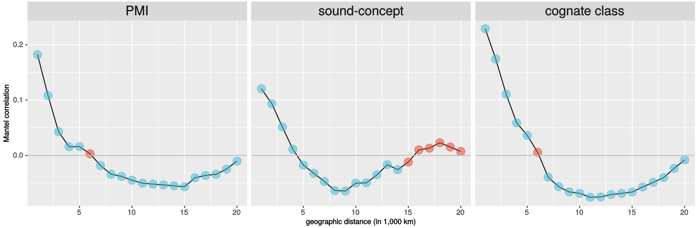
Estimating rates of change

- if phylogeny and states of extant languages are known...
- ... transition rates and ancestral states can be estimated based on Markov model



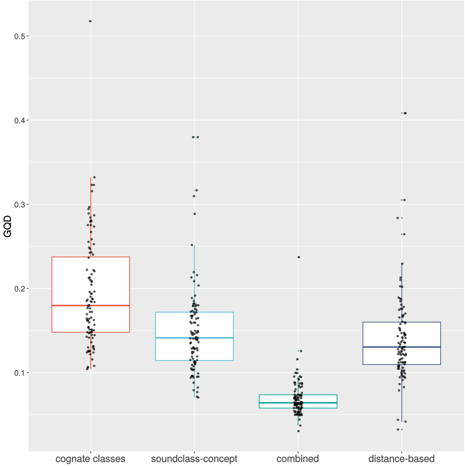
ASJP word lists → character matrix

1. **Automatically inferred cognate classes**
   - each cluster cc defines one character
   - doculect l has value 1 if its word list contains an element of cc, undefined if the slot of the corresponding concept is undefined, and 0 else
2. **Soundclass-concept characters**
   - each combination (c, s) of an ASJP concept c and an ASJP sound class s is a character
   - doculect l has value 1 if one of its entries for c contains s, 0 if not, and undefined if there is no entry for c

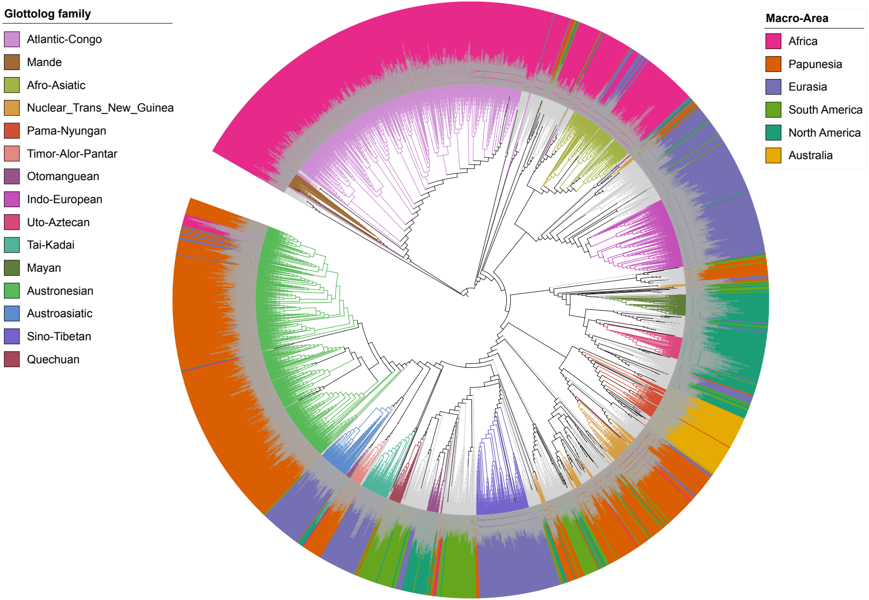


Character matrix → trees

- validation
  - correlation with geographic distance
  - phylogenetic inference (Maximum Likelihood) + comparison to Glottolog expert tree on 100 random sample of ASJP doculects, containing between 20 and 400 doculects
  - using Stamatakis' **RAxML** (which is great)
- partitioned character-based inference seems to work best



The world tree



Cecil H. Brown, Eric W. Holman, Søren Wichmann, and Viveka Velupillai. Automated classification of the world's languages: A description of the method and preliminary results. *STUF — Language Typology and Universals*, 4:285–308, 2008.

Běijng Dàxué. *Hànyǔ fngyán cíhuì* [Chinese dialect vocabularies]. Wénzì Gǎigé, 1964.

Michael Cysouw, Søren Wichmann, and David Kamholz. A critique of the separation base method for genealogical subgrouping. *Journal of Quantitative Linguistics*, 13(2-3):225–264, 2006.

Michael Dunn. Indo-European lexical cognacy database (IELex). URL: http://ielex.mpi.nl/, 2012.

Warren Ewens and Gregory Grant. *Statistical Methods in Bioinformatics: An Introduction*. Springer, New York, 2005.

Joseph Greenberg. Some universals of grammar with special reference to the order of meaningful elements. In *Universals of Language*, pages 73–113. MIT Press, Cambridge, MA, 1963.

Simon J. Greenhill, Robert Blust, and Russell D. Gray. The Austronesian Basic Vocabulary Database: From bioinformatics to lexomics. *Evolutionary Bioinformatics*, 4:271–283, 2008.

Shirō Hattori. Japanese dialects. In Henry M. Hoenigswald and Robert H. Langacre, editors, *Diachronic, areal and typological linguistics*, pages 368–400. Mouton, The Hague and Paris, 1973.

Gerhard Jäger. Global-scale phylogenetic linguistic inference from lexical resources. *Scientific Data*, 5, 2018. doi: 10.1038/sdata.2018.189.

Gerhard Jäger and Pavel Sofroniev. Automatic cognate classification with a Support Vector Machine. In Stefanie Dipper, Friedrich Neubarth, and Heike Zinsmeister, editors, *Proceedings of the 13th Conference on Natural Language Processing*, volume 16 of *Bochumer Linguistische Arbeitsberichte*, pages 128–134. Ruhr Universität Bochum, 2016.

Gerhard Jäger, Johann-Mattis List, and Pavel Sofroniev. Using support vector machines and state-of-the-art algorithms for phonetic alignment to identify cognates in multi-lingual wordlists. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*. ACL, 2017.

Elena Maslova. A dynamic approach to the verification of distributional universals. *Linguistic Typology*, 4(3):307–333, 2000.

Kenneth A. McElhanon. Preliminary observations on Huon Peninsula languages. *Oceanic Linguistics*, 6(1):1–45, 1967. ISSN 00298115, 15279421. URL http://www.jstor.org/stable/3622923.

A IU Militarev. *Towards the chronology of Afrasian (Afroasiatic) and its daughter families*. McDonald Institute for Archaelogical Research, Cambridge, 2000.

Ilia Peiros. Comparative linguistics in Southeast Asia. *Pacific Linguistics*, 142, 1998.

Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, 76(3):036106, 2007.

Joy Sanders and Arden G Sanders. Dialect survey of the Kamasau language. *Pacific Linguistics. Series A. Occasional Papers*, 56:137, 1980.

Alexandros Stamatakis. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 2014.

Søren Wichmann, Eric W. Holman, and Cecil H. Brown. The ASJP database (version 17). http://asjp.clld.org/, 2016.

Mikhail Zhivlov. Annotated Swadesh wordlists for the Ob-Ugrian group. In George S. Starostin, editor, *The Global Lexicostatistical Database*. RGGU, Moscow, 2011. URL: http://starling.rinet.ru.