Project 2

Xiaonan Hu, Lee Stovall, Liqi Zhu

Part 1 ASJP Dataset

—— Liqi Zhu

1. Introduction

The purpose of Part 1 is to find the overall status for those languages we choose based on the population that use the language. (http://www.vistawide.com/languages/language_families_statistics1.htm)

Language Family	Speaker Percentage
Indo-European	44.78%
Sino-Tibetan	22.28%
Niger-Congo	6.26%
Afro-Asiatic	5.93%
Austronesian	5.45%
Dravidian	3.87%
Japanese	2.16%
Austro-Asiatic	1.77%
Tai-Kadai	1.37%
Total	**93.87%**

We managed to download the dataset for each language family from ASJP database.

2. Method & Processing Data

Our thought is to compare the Distance data within each language family. One dataset is with the numbers and the other is not.

```
#### Cygwin ####
egrep -v '\sone|\stwo' < Indo-European.txt > IEwonum.txt
egrep -v '\sone|\stwo' < Austro-Asiatic.txt > AuAwonum.txt
egrep -v '\sone|\stwo' < Austronesian.txt > Awonum.txt
egrep -v '\sone|\stwo' < Dravidian.txt > Dwounum.txt
egrep -v '\sone|\stwo' < Japanese.txt > Jwonum.txt
egrep -v '\sone|\stwo' < Niger-Congo.txt > NCwonum.txt
egrep -v '\sone|\stwo' < Sino-Tibetan.txt > STwnum.txt
egrep -v '\sone|\stwo' < Tai-Kadai.txt > TKwonum.txt
egrep -v '\sone|\stwo' < Afro-Asiatic.txt > AAwonum.txt
# Exclude numbers in dataset > Datasets without numbers.
```

With Programs for calculating ASJP distance matrices (http://asjp.clld.org/software). We managed to get the distance matrix for each language family.

```
#### Power Shell ####
 asjp62 < listss17.txt > output1.txt
 asjp62 < Indo-European.txt > IEall.txt
 asjp62 < Austro-Asiatic.txt > AuAall.txt
 asjp62 < Austronesian.txt > Aall.txt
 asjp62 < Dravidian.txt > Dall.txt
 asjp62 < Japanese.txt > Jall.txt
 asjp62 < Niger-Congo.txt > NCall.txt
 asjp62 < Tai-Kadai.txt > TKall.txt
 asjp62 < Sino-Tibetan.txt > STall.txt
 asjp62 < Afro-Asiatic.txt > AAall.txt
# Overall view of whole dataset and different language families with numbers.
 asjp62 < IEwonum.txt > IEall_won.txt
 asjp62 < AuAwonum.txt > AuAall_won.txt
 asjp62 < Awonum.txt > Aall_won.txt
 asjp62 < Dwounum.txt > Dall_won.txt
 asjp62 < Jwonum.txt > Jall_won.txt
 asjp62 < NCwonum.txt > NCall_won.txt
 asjp62 < TKwonum.txt > TKall_won.txt
 asjp62 < STwnum.txt > STall_won.txt
 asjp62 < AAwonum.txt > AAall_won.txt
# Overall view of whole dataset and different language families without numbers.
```

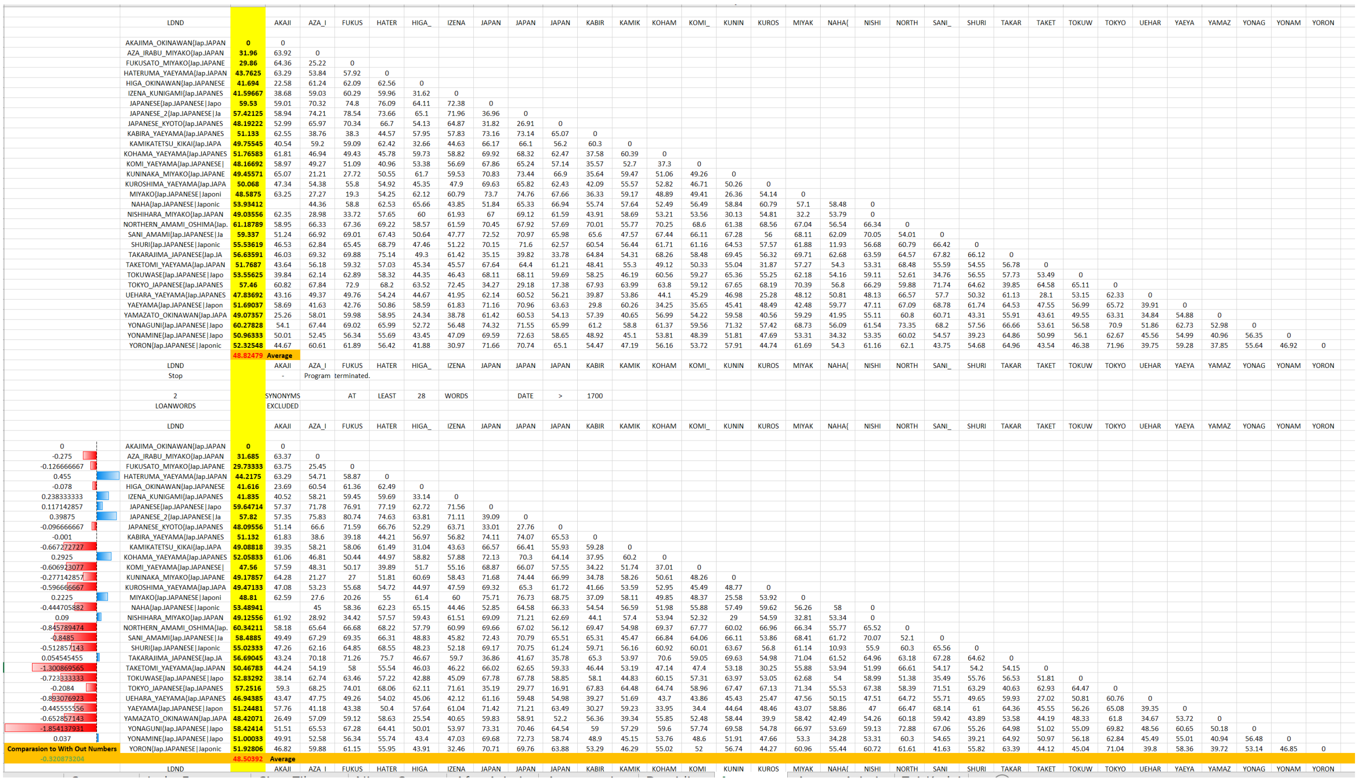
```
#### Cygwin ####
cat IEall*.txt > IE.txt
cat AuAall*.txt > AuA.txt
cat Aall*.txt > A.txt
cat Dall*.txt > D.txt
cat Jall*.txt > J.txt
cat NCall*.txt > NC.txt
cat TKall*.txt > TK.txt
cat STall*.txt > ST.txt
cat AAall*.txt > AA.txt

sed 's/ \+/,/g' IE.txt > IE.csv
sed 's/ \+/,/g' AuA.txt > AuA.csv
sed 's/ \+/,/g' A.txt > A.csv
sed 's/ \+/,/g' D.txt > D.csv
sed 's/ \+/,/g' J.txt > J.csv
sed 's/ \+/,/g' NC.txt > NC.csv
sed 's/ \+/,/g' TK.txt > TK.csv
sed 's/ \+/,/g' ST.txt > ST.csv
sed 's/ \+/,/g' AA.txt > AA.csv
# Arrange data into CSV file
```

With the combination and calculation of the .CSV files, we are able to compare the ASJP Distance of each language familt with and without numbers.

3. Results

Example of Japanese



As showed above, values are mesured ASJP Distance data. We calculated the average standard for each family language family and then compare the Distance matrix with numbers and the one without.
Then we repeat the process for the other language families.
The results of all language families are followed:

	Speaker Percentage	ASJP Distance With Numbers	ASJP Distance Without Numbers	Difference
Indo-European	44.78%	85.56	85.84	0.280495
Sino-Tibetan	22.28%	88.27	88.46	0.191502
Niger-Congo	6.26%	91.43	91.32	-0.1074
Afro-Asiatic	5.93%	91.32	91.16	-0.15255
Austronesian	5.45%	85.31	85.40	0.089109
Dravidian	3.87%	85.31	57.28	-28.0276
Japanese	2.16%	48.82	48.50	-0.32087
Austro-Asiatic	1.77%	88.80	81.34	-7.4635
Tai-Kadai	1.37%	75.99	75.61	-0.37409
Average	10.43%	82.31	78.32	-3.99
Total	93.87%	740.80	704.92	-35.8849

4. Conclusion

As shown above, the first thing we discovered is that language families have different features. Japanese was obviously the language family containing the most similar languages since ASJP Distance is the smallest. While all the other language families show different level of variance.
About the influence of numbers, Dravidian's Distance drops significantly, which means the number changes a lot in Dracvidian Languages. Thus to say all the language families with positive numbers of Difference means number changes less than other words in that family. And Negative number indicates number changes more.
As a conclusion, in Indo-European, Sina-Tibetan, Austronesian language families, numbers changes less. While in

other language families numbers may change more than average.

5. Comments

ASJP Database is useful, while the shortcomings are obvious.
The first problem is that ASJP translated the language into ASJP code, and the ASJP Distance has no defenition of calculation either on website or in software instruction. We can only campare the numbers but we don't know where the number itself was calculated.
The second problem is that the program on the website for calculation of ASJP Distance works only for ASJP Format txt file with a strict rule, which troubles a lot when considering subsets. And the program takes so long to output. Not to mention the program is not modifiable.
Additionally, ASJP Database is not complete. For example, Altaic family is missing. And some language has missing values.

Part 2 Words similarity calculated by *difflib.SequenceMatcher*

—— Xiaonan Hu

1. Introduction

The **difflib** module contains tools for comparing sequences. It is flexible for pairs of sequences of any type, so long as their elements are hashable. And it can produce reports using various formats.
The **SequenceMatcher** class works by finding subsequences common in both sequences, and the ratio returned as the result of *2*the number of matches/the total number of elements in both sequences* .

And in the programming work I have done, I intergrated the paring process with the comparing process, the defined function is shown below. With input of any list of 'words by languages', where n is number of two-language combinations, m is the number of words involved, the function will return the matrix of similarities comparing the same words of any two paring languges.

```
def list_S(datalist, n, m):
    result = np.zeros((m, n))
    for i in range(m):
        nums = datalist[i]
        com = list(combinations(nums, 2))
        for j in range(n):
```

```
            seq = difflib.SequenceMatcher(None, com[j][0], com[j][1])
            result[i][j] = seq.ratio()*100
    return result
```

2. Data

Here we used the data of *Counting to a thousand in 14 different languages* cited from *Sbiis Saibian's Large Number Site*.
The dataset contains the words of **1, 2, ...,19, 20, 30, ... , 90, 100, 200, ..., 900, 1000** from 14 languages, including **English, Spanish, Latin, Greek, Japanese, Chinese, Hebrew, Italain, French, German, Swahili, Sanskrit, Welsh and Thai**.

The completed dataset shown below.

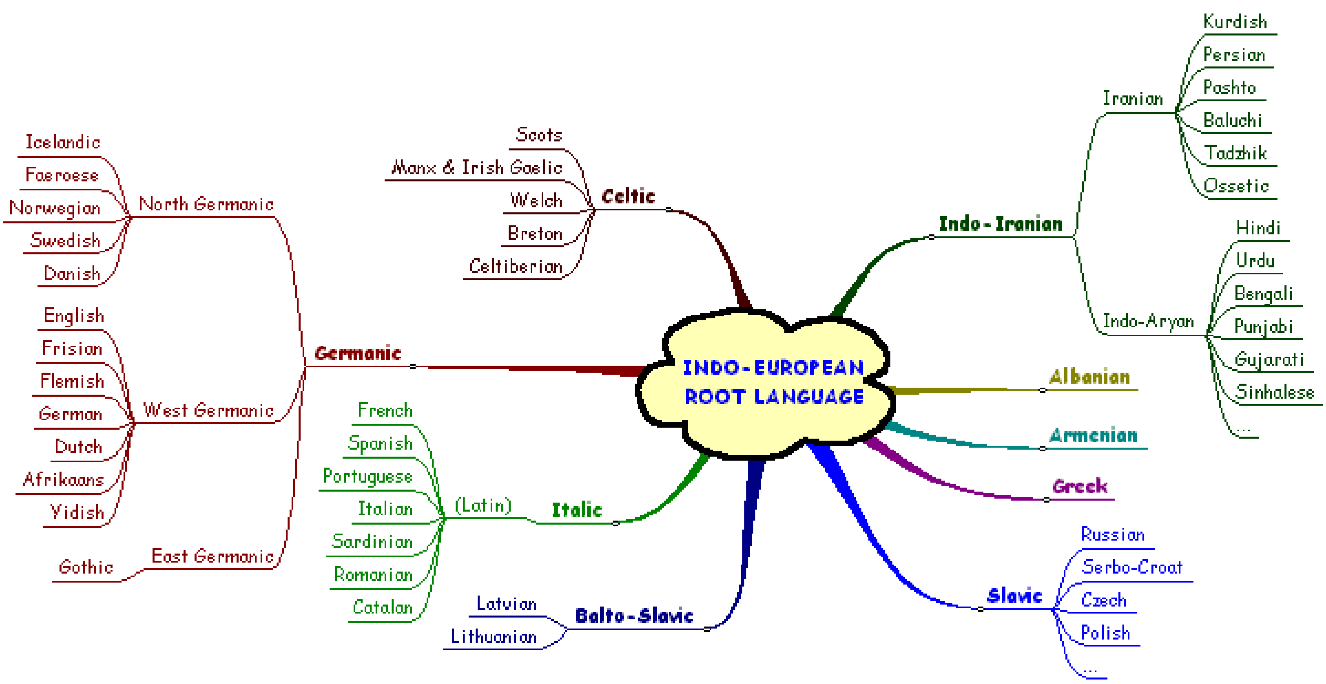
VALUE	ENGLISH	SPANISH	LATIN	GREEK	JAPANESE	CHINESE	HEBREW	ITALIAN	FRENCH	GERMAN	SWAHILI	SANSKRIT	WELSH	THAI
1	one	uno	unus	enas	iti	yi	echad	uno	un	eins	moja	eka	un	nueng
2	two	dos	duo	duo	ni	er	shnayim	due	deux	zwei	mbili	dvi	dau	song
3	three	tres	tres	treis	san	san	shlosha	tre	trois	drei	tatu	tri	tri	sam
4	four	cuatro	quattuor	tessera	si	si	arba'a	quattro	quatre	vier	nne	chatur	pedwar	see
5	five	cinco	quinque	pente	go	wu	chamisha	cinque	cinq	funf	tano	pancha	pump	har
6	six	seis	sex	exi	roku	liu	shisha	sei	six	sechs	sita	shash	chwech	hok
7	seven	siete	septem	epta	siti	qi	shiv'a	sette	sept	sieben	saba	sapta	saith	jed
8	eight	ocho	octo	okto	hati	ba	shmonah	otto	huit	acht	nane	ashta	wyth	bad
9	nine	nueve	novem	ennea	kyuu	jiu	tish'a	nove	neuf	neun	tisa	nava	naw	gao
10	ten	diez	decem	deka	zyuu	shi	assara	dieci	dix	zehn	kumi	dasha	deg	sib
11	eleven	once	undecim	endeka	zyuu-iti	shi-yi	achad asar	undici	onze	elf	kumi na moja	ekadashan	un-deg-un	sib-et
12	twelve	doce	duodecim	dodeka	zyuu-ni	shi-er	shneim asar	dodici	douze	zwolf	kumi na mbili	dvadashan	un-deg-dau	sib-song
13	thirteen	trece	tredecim	dekatreis	zyuu-san	shi-san	shlosha asar	tredici	treize	dreizehn	kumi na tatu	tridashan	un-deg-tri	sib-sam
14	fourteen	catorce	quattuordecim	dekatessera	zyuu-si	shi-si	arba'a asar	quattordici	quatorze	vierzehn	kumi na nne	chaturdashan	un-deg-pedwar	sib-see
15	fifteen	quince	quindecim	dekapente	zyuu-go	shi-wu	chamisha asar	quindici	quinze	funfzehn	kumi na tano	panchadashan	un-deg-pedwar	sib-har
16	sixteen	dieciseis	sedecim	dekaexi	zyuu-roku	shi-liu	shisha asar	sedici	seize	sechzehn	kumi na sita	shashdashan	un-deg-chwech	sib-hok
17	seventeen	diecisiete	septendecim	dekaepta	zyuu-siti	shi-qi	shiv'a asar	dicissette	dix-sept	siebzehn	kumi na saba	saptadashan	un-deg-saith	sib-jed
18	eighteen	dieciocho	duodeviginti	dekaokto	zyuu-hati	shi-ba	shmona asar	diciotto	dix-huit	achtzehn	kumi na nane	ashtadashan	un-deg-wyth	sib-bad
19	nineteen	diecinueve	undeviginti	dekaennea	zyuu-kyuu	shi-jiu	tish'a asar	diciannove	dix-neuf	neunzehn	kumi na tisa	navadashan	un-deg-naw	sib-gao
20	twenty	veinte	viginti	eikosi	ni-zyuu	er-shi	esrim	venti	vingt	zwanzig	ishirini	vinshat	dau-ddeg	yee-sib
30	thirty	treinta	triginta	trianta	san-zyuu	san-shi	shloshim	trenta	trente	dreiBig	thelathini	trinshat	tri-deg	sam-sib
40	forty	cuarenta	quadraginta	saranta	si-zyuu	si-shi	arba'im	quaranta	quarante	vierzig	arobaini	catvarinshat	pedwar-deg	see-sib
50	fifty	cincuenta	quinquaginta	penenta	go-zyuu	wu-shi	chamishim	cinquanta	cinquante	funfzig	hamsini	panchashat	pum-deg	har-sib
60	sixty	sesenta	sexaginta	exenta	roku-zyuu	liu-shi	shishim	sessanta	soixante	sechzig	sitini	shashti	chew-deg	hok-sib
70	seventy	setenta	septuaginta	ebdomenta	siti-zyuu	qi-shi	shiv'im	settanta	soixante-dix	siebzig	sabini	saptati	saith-deg	jed-sib
80	eighty	ochenta	octoginta	ogdoenta	hati-zyuu	ba-shi	shmonim	ottanta	quatre-vingts	achtzig	themanini	ashiti	wyth-deg	bad-sib
90	ninety	noventa	nonaginta	enenenta	kyuu-zyuu	jiu-shi	tish'im	novanta	quatre-vingt-dix	neunzig	tisini	navati	naw-deg	gao-sib
100	hundred	cien(ciento)	centum	ekato	hyaku	bai	me'a	cento	cent	hundert	mia	shata	cant	nueng-roi
200	two hundred	doscientos	ducenti	diakosia	ni-hyaku	er-bai	matayim	duecento	deux cents	zweihundert	mia mbili	dvashatam	dau gant	song-roi
300	three hundred	trescientos	trecenti	triakosia	san-hyaku	san-bai	shlosh meot	trecento	trois cents	dreihundert	mia tatu	trishatam	tri chant	sam-roi
400	four hundred	cuatrocientos	quadringenti	tetrakosia	si-hyaku	si-bai	arba meot	quattrocento	quatre cents	vierhundert	mia nne	chaturshatam	pedwar cant	see-roi
500	five hundred	quinientos	quingenti	pentekosia	go-hyaku	wu-bai	chamesh meot	cinquecento	cinq cents	funfhundert	mia tano	panchashatam	pum cant	har-roi
600	six hundred	seiscientos	sescenti	exakosia	roku-hyaku	liu-bai	shesh meot	seicento	six cents	sechshundert	mia sita	shashshatam	chwe chant	hok-roi
700	seven hundred	setecientos	septingenti	eptakosia	siti-hyaku	qi-bai	shva meot	settecento	sept cents	siebenhundert	mia saba	saptashatam	saith cant	jed-roi
800	eight hundred	ochocientos	octingenti	oktakosia	hati-hyaku	ba-bai	shmone meot	ottocento	huit cents	achthundert	mia nane	ashtashatam	wyth cant	bad-roi
900	nine hundred	novecientos	nongenti	enniakosia	kyuu-hyaku	jiu-bai	tsha meot	novecento	neuf cents	neunhundert	mia tisa	navashatam	naw cant	gao-roi
1000	thousand	mil	mille	chilia	sen	qian	elef	mille	mille	tausend	elfu moja	sahasra	mil	nueng-pun

3. Result

By pairing any two languages from the dataset, we have 91 pairs of languages in total. Then we run the *difflib.SequenceMatcher* function on the same words from each pair of languages. By doing so, we have 91×37 measures of similarity, and then we calculated the mean similarity of each pair of languages. The results shown in the table below.

No.	1	2	3	4	5	6	7	8	9	10	11	12	13
Pair	('SPANISH', 'ITALIAN')	('LATIN', 'ITALIAN')	('ITALIAN', 'FRENCH')	('SPANISH', 'LATIN')	('SPANISH', 'FRENCH')	('LATIN', 'FRENCH')	('SPANISH', 'GREEK')	('LATIN', 'GREEK')	('GREEK', 'ITALIAN')	('ENGLISH', 'GERMAN')	('FRENCH', 'WELSH')	('LATIN', 'WELSH')	('CHINESE', 'THAI')
Mean	71.06957862	69.19623864	59.96061443	59.12616873	57.79440741	54.97195778	47.35608302	45.20061652	43.66586342	43.60707971	40.43680301	38.83859508	38.05167805
No.	14	15	16	17	18	19	20	21	22	23	24	25	26
Pair	('GREEK', 'FRENCH')	('GREEK', 'SANSKRIT')	('ITALIAN', 'WELSH')	('HEBREW', 'SANSKRIT')	('HEBREW', 'SWAHILI')	('SPANISH', 'WELSH')	('LATIN', 'GERMAN')	('ENGLISH', 'SPANISH')	('ENGLISH', 'FRENCH')	('SPANISH', 'GERMAN')	('FRENCH', 'GERMAN')	('GREEK', 'WELSH')	('ENGLISH', 'LATIN')
Mean	37.09699576	37.07622537	36.72512772	36.71144572	35.95238371	35.42661108	34.8212935	34.39640111	33.85809776	33.35179081	32.99116835	32.80363629	32.35999498
No.	27	28	29	30	31	32	33	34	35	36	37	38	39
Pair	('SPANISH', 'SANSKRIT')	('JAPANESE', 'CHINESE')	('GERMAN', 'WELSH')	('GERMAN', 'SANSKRIT')	('ENGLISH', 'ITALIAN')	('LATIN', 'SANSKRIT')	('SANSKRIT', 'WELSH')	('FRENCH', 'SANSKRIT')	('ITALIAN', 'SANSKRIT')	('ITALIAN', 'GERMAN')	('SWAHILI', 'SANSKRIT')	('ENGLISH', 'GREEK')	('GREEK', 'SWAHILI')
Mean	32.26735434	31.99243523	31.62983268	31.37686782	31.18580003	30.97263707	30.52366571	30.03334859	29.91418233	29.71651185	27.09507169	26.90443297	25.30609899
No.	40	41	42	43	44	45	46	47	48	49	50	51	52
Pair	('CHINESE', 'HEBREW')	('GREEK', 'GERMAN')	('JAPANESE', 'THAI')	('HEBREW', 'THAI')	('LATIN', 'SWAHILI')	('ENGLISH', 'SANSKRIT')	('CHINESE', 'SWAHILI')	('FRENCH', 'SWAHILI')	('JAPANESE', 'GERMAN')	('SPANISH', 'HEBREW')	('SPANISH', 'SWAHILI')	('SWAHILI', 'WELSH')	('ITALIAN', 'SWAHILI')
Mean	24.43643462	24.33114369	23.82863956	23.64774595	23.23802146	23.0197805	23.00127025	22.96597939	22.07838341	22.01927367	21.93836981	21.76431479	21.35216616
No.	53	54	55	56	57	58	59	60	61	62	63	64	65
Pair	('CHINESE', 'SANSKRIT')	('HEBREW', 'ITALIAN')	('ENGLISH', 'SWAHILI')	('LATIN', 'HEBREW')	('ENGLISH', 'WELSH')	('HEBREW', 'FRENCH')	('CHINESE', 'GERMAN')	('HEBREW', 'GERMAN')	('SWAHILI', 'THAI')	('GREEK', 'HEBREW')	('GREEK', 'THAI')	('HEBREW', 'WELSH')	('GERMAN', 'THAI')
Mean	21.2388755	20.2986299	20.16650339	19.95768281	19.90144831	19.80612945	19.79990254	19.73766204	19.65622088	19.63875747	19.61966006	19.55501964	19.4889337
No.	66	67	68	69	70	71	72	73	74	75	76	77	78
Pair	('GERMAN', 'SWAHILI')	('JAPANESE', 'SWAHILI')	('SANSKRIT', 'THAI')	('LATIN', 'THAI')	('JAPANESE', 'WELSH')	('WELSH', 'THAI')	('JAPANESE', 'SANSKRIT')	('ENGLISH', 'HEBREW')	('FRENCH', 'THAI')	('ITALIAN', 'THAI')	('SPANISH', 'THAI')	('JAPANESE', 'FRENCH')	('CHINESE', 'FRENCH')
Mean	19.25742851	18.57558908	18.33340664	18.05629142	17.71399128	17.59328971	17.49050023	17.22631588	16.93377744	16.7893895	16.74863778	15.91905807	15.906692
No.	79	80	81	82	83	84	85	86	87	88	89	90	91
Pair	('SPANISH', 'CHINESE')	('ENGLISH', 'THAI')	('JAPANESE', 'HEBREW')	('CHINESE', 'WELSH')	('LATIN', 'CHINESE')	('CHINESE', 'ITALIAN')	('GREEK', 'CHINESE')	('LATIN', 'JAPANESE')	('ENGLISH', 'JAPANESE')	('SPANISH', 'JAPANESE')	('GREEK', 'JAPANESE')	('ENGLISH', 'CHINESE')	('JAPANESE', 'ITALIAN')
Mean	15.13863564	15.02390536	14.78752344	14.59301483	14.06191174	13.94699498	13.79670233	13.72334761	13.58847212	12.92088937	12.88247983	12.69938162	11.60727938

We can draw a conclusion that **Spanish and Italian** are the most similar two languages, among all the 14 languages involved in the comparison of number words. However, **Japanese and Italian** have the least similarity among the 14 languages.
And We can briefly draft out a network of strong relations among the languages, including **Spanish, Italin, Latin, French, Greek and Welsh**, because the similarity between any two of them is in the top 25. Furthermore, the relations among **Spanish, Italian, Latin and French** are especially strong, it consistent with one of the branches of the widely accepted Indo-European language tree, as the image cited below.



Source:http://www.viking.no/e/heritage/indo.htm

4. Comments

The advantage of the programming work I deveeloped, is defining the pairing and comparing works together as a function. So that the function can be implemented in other dataset with ease. And so long as we have dataset including more languages and words, more advanced research and conclusion can be achieved.

This algorithm is only based on the comparison of letter-sequences, so that it doesn't include the measurement of the pronouncement, which is also an important but tough task for comparing languages. And because the languages, like Chinese and Japanese, are not originaly in the form of letters, more bias may be involved by such conversion. So the results invoved a translated language should be less precise.

Part 3 Degrees of variance in language families

—— Lee Stovall

1. Introduction

For the last part of our project, we wanted to see if the variance between number words and baseline words of the same language family could differ in degree compared to the degree of variance in other language families. Results yielded from this experiment could help us understand which language families hold the greatest diversity in dialect as well as the ones that don’ t. In turn, these results can lead to further exploration as to what these differences are and why some degrees of variance are greater than others.

2. Method

We refined our method for the difflib.SequenceMatcher to focus on the differences between languages within the same family and then compared these degrees of variance amongst others. Keep in mind that we are using the same families used from part 1.
Additionally, we use the data provided by part 1 to aid in comparing the differences in variance of language families. As concluded by the end of part 1, we will not be looking at the results of the Japanese language family (since that has already been covered. Instead, we shall focus on Indo-European and Sino-Tibetan language families and their differences in language variance.

3. Data

The results from the differences in number words were compared with the differences in baseline words amongst languages of the same family. We then took the mean of variance between all possible language combinations of the same family and compared them to the mean of variance of other language families.We were able to look at the results that we gathered through part 2 to find an average variance for language families. This made getting the results easier since we already collected the relevant information.

Example

Variance of languages in word numbers between Sino-Tibetan and Indo-European
language families.

Indo-European	Sino-Tibetan

	Mean
('SPANISH', 'ITALIAN')	71.06957862
('LATIN', 'ITALIAN')	69.19623864
('ITALIAN', 'FRENCH')	59.96061443
('SPANISH', 'LATIN')	59.12616873
('SPANISH', 'FRENCH')	57.79440741
('LATIN', 'FRENCH')	54.97195778
('SPANISH', 'GREEK')	47.35608302
('LATIN', 'GREEK')	45.20061652
('GREEK', 'ITALIAN')	43.66586342
('ENGLISH', 'GERMAN')	43.60707971
('FRENCH', 'WELSH')	40.43680301
('LATIN', 'WELSH')	38.83859508
('GREEK', 'FRENCH')	37.09699576
('ITALIAN', 'WELSH')	36.72512772
('SPANISH', 'WELSH')	35.42661108
('LATIN', 'GERMAN')	34.8212935
('ENGLISH', 'SPANISH')	34.39640111
('ENGLISH', 'FRENCH')	33.85809776
('SPANISH', 'GERMAN')	33.35179081
('FRENCH', 'GERMAN')	32.99116835
('GREEK', 'WELSH')	32.80363629
('ENGLISH', 'LATIN')	32.35999498
('GERMAN', 'WELSH')	31.62983268
('ENGLISH', 'ITALIAN')	31.18580003
('ITALIAN', 'GERMAN')	29.71651185
('ENGLISH', 'GREEK')	26.90443297
('GREEK', 'GERMAN')	24.33114369
('ENGLISH', 'WELSH')	19.90144831
Avg. Variance	40.67

	Mean
('CHINESE', 'THAI')	38.05167805
('CHINESE', 'SANSKRIT')	21.2388755
('SANSKRIT', 'THAI')	18.33340664
Avg. Variance	25.87

We can conclude from these results that the variance between **Indo-European** word numbers are less than the variance between **Sino-Tibetan** languages. As for baseline words, the results from part 1 already tell us about the commonalities of dialects between languages of a given family. Based from the results from average variance aswell as the data retrieved from part 1, we are able to conclude that **Sino-Tibetan** family is greater in language diversity.

Conclusion

Like before, our calculations are completely based on the lexical difference between languages. This may neglect dialect differences in pronunciation. However, it can be argued that our margin of error is smaller with this particular part of the experiment. Since there is more likely to be a common pronunciation between two dialects that are from the same family (rather than from two distinct families), the degree of variance between dialects are likely to be more accurate. Additionally, the results acquired from this are meant to provoke further questioning to learn more about the differences in the languages of a given language family as well as raise some questions about why they have such differences. They are not meant to give us anything concrete.