

English Consonants & Vowels

- Phonetics: a study on speech

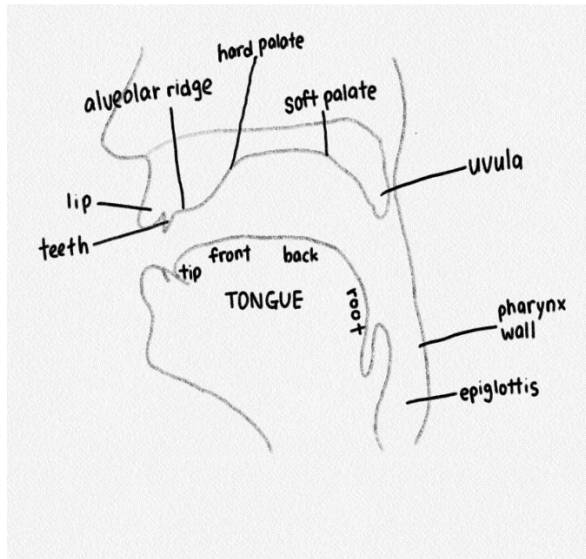
articulatory phonetics (from mouth) → how to produce speech

acoustic phonetics (through air) → how to transmit speech

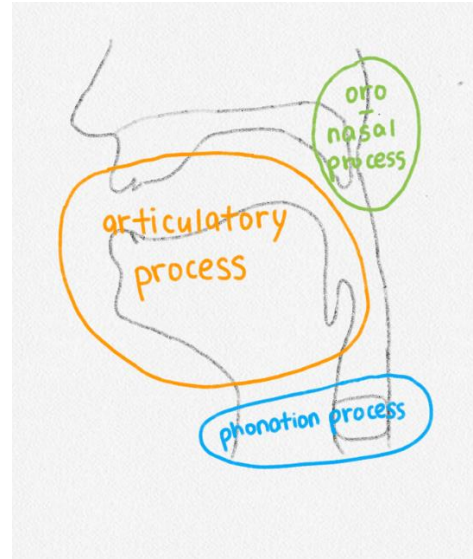
auditory phonetics (to ear) → how to hear speech

Articulation

- Vocal tract:



- 5 speech organs = constrictors = articulators



Phonation Process in Larynx

- larynx = voicebox: voiced → can feel vibration

ex. v, z, l, m, a, i

voiceless → can't feel vibration

ex. f, s, k, p, h

Oro-nasal Process in Velum

- nasal: when velum lowered

ex. m, n, ng

Articulatory Process

- lips / tongue tip / tongue body

Control of Constrictors(Articulators)

- Each constrictor needs to be more specific in geometry

constriction location(CL) / constriction degree(CD)

- Constriction location: Lips → bilabial / labiodental

Tongue body → palatal / velar

Tongue tip → dental / alveolar / retroflex / palate-alveolar

- Constriction degree: stops > fricatives > approximants (/r, l, w, j/) > vowels

How to Produce English Consonants and Vowels

- constrictors(lips, tongue tip, tongue body) / CD / CL / velum / larynx
 - ex) /p/: lips / bilabial / stop / velum raised / larynx open
 - /b/: lips / bilabial / stop / velum raised / larynx closed
 - /d/: tongue tip / alveolar / stop / velum raised / larynx closed
 - /z/: tongue tip / alveolar / fricative / velum raised / larynx closed
 - /n/: tongue tip / alveolar / stop / velum lowered / larynx closed
- * 모든 모음은 constrictor로서 tongue body 만 사용(constrictor location 정의 X)
- Phonemes: individual sounds that form words
 - a combination of speech organs' actions

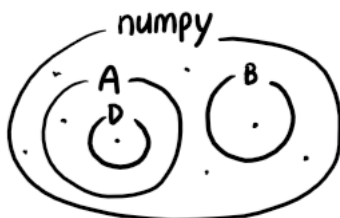
Acoustics

- Praat: duration > select(click and drag on waveform or spectrogram) →
 - read a value (sec.) on the top → zoom in (if not visible)
 - intensity > show intensity → click on green → read a value (dB) on the right
 - pitch > show pitch → pitch setting > pitch range > 65-200Hz male / 145-276Hz female
 - click on blue → read a value (Hz) on the right
 - formant(모음 구별 수치) > show formants → place the cursor on one of the trajectories
 - read a value (Hz) on the left
- the number of occurrences of a repeating event per second (frequency, Hz)
 - repeating event = vibration of vocal folds / repeating > sine wave = pure tone
 - * sine wave: frequency + magnitude(amplitude) (x 축 시간 / y 축 value, voltage)
- 모든 신호는 단순한 sine wave 들의 합으로 표현된다. (synthesis)
 - complex tone 이 반복하는 주기는 Fundamental Frequency 와 동일
 - spectrum: x 축 frequency / y 축 magnitude(amplitude)
 - spectrogram: spectrum 을 시간으로 visualize 한 것 (x 축 시간 / y 축 frequency)
 - sine wave(time-value graph)→spectrum: spectral analysis
- pure tone→spectral analysis: frequency 가 같은 sine wave 한 개
 - complex tone→spectral analysis: 일정한 간격의 sine wave 여러 개 (간격=pitch)
 - (Praat: Spectrum > View Spectrum Slice)
- source: 성대에서 나는 소리 (measured by EGG)
 - human voice source consists of harmonics
 - a complex tone = sum of pure tones at integer multiples of the lowest pure tone
 - the lowest pure tone = fundamental frequency(F0) = rate of vibration of the larynx
 - = the number of opening-closing cycles of the larynx per second
 - amplitude of pure tones gradually decreases
- filter: vocal tract 에 의해서 달라지는 소리
 - filter 의 spectrum → jiggjagging with peaks and valleys (amplitude 의 패턴이 사라짐)
 - peaks/mountains: frequencies VT likes (formants)
 - valleys: frequencies VT does not like

- Synthesize Source: New > Sound > Create Sound as Pure Tone
 - > Tone frequency 100~1000Hz / Amplitude 1.0~0.55Pa (10 개의 pure tone)
 - Combine > Combine to Stereo (10 개의 channel 을 가진 하나의 stereo)
 - Convert > Convert to Mono (10 개의 pure tone 이 합쳐진 complex tone)
 - 반복 주기: frequency 100Hz / Amplitude 1.0Pa 와 일치
 - 음: frequency 100Hz / Amplitude 1.0Pa 와 일치
 - Spectrum > View Spectral Slice: gradually decrease / 10 개 / 100Hz
- F1: 모음의 height / F2: 모음의 frontness / backness
F1 and F2 are enough to disambiguate vowels.
(Praat: New > Sound > Create Sound as VowelEditor)

Coding

- 코딩: 자동화 > 똑같은 과정을 쉽게 반복할 수 있기 위해서
- 모든 language 는 공통적으로 단어와 문법으로 이루어짐
단어: 정보를 담는 그릇
- Computer Language 의 단어: 변수(variable)
Computer Language 의 문법: 1. variable assigning
2. 'if' conditioning
3. 'for' loop
함수: 어떤 입력을 넣어야 자신이 원하는 출력이 나오는지
ex. Praat 입력: 마우스로 구간 설정 / Praat 출력: 소리
- Anaconda Prompt > 'Jupyter Notebook' 입력
원하는 디렉토리 > New > Python 3
- cell 생성: cell 선택 후 b(아래쪽에 생성) / a(위에 생성) / x(삭제)
- Run: Shift + Enter
- =: 오른쪽에 있는 정보를 왼쪽에 있는 variable 로 assign 한다
ex. a=1 > 정보: 1 / variable: a
Print: 어떤 변수를 넣으면 그 값을 출력함
- Comment 쓰기: cell 에 #을 쓴 후 내용 적기
Code 를 Markdown 으로 바꾼 후 내용 적기
- numpy: 라이브러리 중 하나 (수학적으로 계산 가능)



import numpy > numpy.A.D.f (numpy 라이브러리 안의 A 안의 D 안의 f 함수)
from numpy import A > A.D.f (numpy 라이브러리에서 A 를 불러옴)
from numpy import A.D > A.D.f (numpy 라이브러리에서 A 안의 D 를 불러옴)

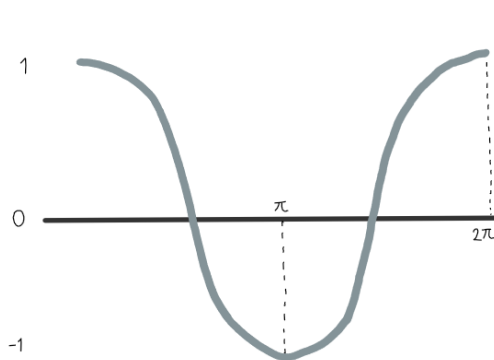
- float64: float 의 정밀함의 정도 (64 비트) / 메모리를 많이 차지함

행렬과 벡터

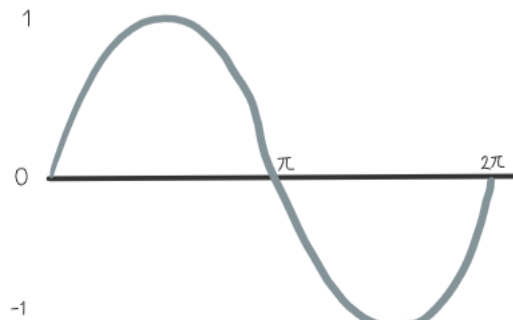
- 이미지, 소리, 텍스트 > 벡터
모든 데이터는 벡터로 나타낼 수 있음
- 흑백 이미지 > 2 차원 / 컬러 이미지 > 3 차원 / 영상 > 4 차원

Sound

- Sinusoidal: cos 이나 sin 과 같은 곡선
Phasor: sinusoidal 의 function 을 만드는 것
- π (무리수)
0 ~ π ~ 2π (radians - sin/cos 의 입력값 / θ)
0° ~ 180° ~ 360° (degree)
- Phasor:

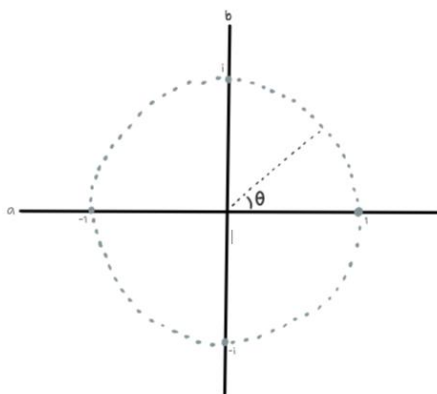


cos 함수



sin 함수

- 오일러 공식: $f(\theta) = e^{i\theta} = \cos(\theta) + \sin(\theta)i$
e = 2.71... (무리수) / i = imaginary (허수, $\sqrt{-1}$)
복소수 = $a + bi$ (모든 수 표현 가능) > $f(\theta)$
 $\theta = 0 > f(0) = e^0 = 1$
 $\theta = \frac{\pi}{2} > f(\frac{\pi}{2}) = \cos(\frac{\pi}{2}) + \sin(\frac{\pi}{2})i = i$
 $\theta = \pi > f(\pi) = \cos(\pi) + \sin(\pi)i = -1$
 $\theta = \frac{3\pi}{2} > f(\frac{3\pi}{2}) = \cos(\frac{3\pi}{2}) + \sin(\frac{3\pi}{2})i = -i$



복소평면 (complex plane): a 실수 / b 허수

Projection: x 축(a)에서 볼 때 > 허수(sin 그래프)

y 축(b)에서 볼 때 > 실수(cos 그래프)

- sin 과 cos 음의 같음 (sin 그래프와 cos 그래프는 $\frac{\pi}{2}$ 의 차이)

인간의 귀는 phasor shift 는 구별하지 못함 / frequency 의 차이만 구별

Linear Algebra

- 인공지능: 데이터(vector) > 기계(함수) > 데이터(vector) / 행렬의 곱

$$\begin{array}{ccccccc} & & -1 & 0 & 2 & & \\ -5 & 3 & 0 & 1 & \times & \begin{bmatrix} 0 & 1 & 3 \\ 3 & -5 & 7 \\ 2 & 3 & 4 \end{bmatrix} & = & \begin{bmatrix} -3 & 6 & 23 \end{bmatrix} \end{array}$$

1x4

4x3

1x3

행렬의 곱 계산법: $5*(-1) + 3*0 + 0*3 + 1*2 = -3$

$$5*0 + 3*-1 + 0*-5 + 1*3 = 6$$

$$5*2 + 3*3 + 0*7 + 1*4 = 23$$

- Matrices: $\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}$ (m 행 n 열 / m by n 행렬)

- Vector: a sequence of numbers

$$\begin{bmatrix} a_1 \\ \vdots \\ a_m \end{bmatrix} \text{ (m by 1 행렬 / column vector) / } [a_1 \dots a_n] \text{ (1 by n 행렬)}$$

- Vector Spaces: 여러 벡터들이 만들어내는 공간 / linear combinations stay in the space

Linear Combinations: $c*v + d*w$ (c, d: scalars / v, w: vectors)

R^n space consists of all vectors with n components (차원의 모든 공간 / 일부분 X)

- Column Space: $A = \begin{bmatrix} 2 & -1 \\ 1 & 3 \end{bmatrix}$ > (2, 1) (-1, 3)을 linear combination 하면 모든 공간을 채움
column vector 가 column space 를 채움

col1 & 2 not on a line > independent

: Whole Space(vector 가 속해있는 space) = Column Space = 2 차원

$$A = \begin{bmatrix} 2 & -1 \\ 1 & -0.5 \end{bmatrix} > (2, 1) (-1, -0.5)$$

col1 & 2 on a line > dependent

: Whole Space = 2 차원 / Column Space = 1 차원

dim(whole space) = n rows / dim(column space) = n of independent columns

$$A = \begin{bmatrix} 1 & 3 & -2 \\ 2 & 3 & 0 \\ 4 & 1 & 5 \\ 1 & 2 & -2 \end{bmatrix} > \text{Whole Space} = \text{Column Space} = 3 \text{ 차원}$$

$$A = \begin{bmatrix} 4 & 1 & 5 \\ 2 & 4 & 0 \\ 4 & 8 & 5 \end{bmatrix} > \text{Whole Space} = 3 \text{ 차원} / \text{Column Space} = 2 \text{ 차원}$$

$$(2*\text{Col1} = \text{Col2})$$

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 1 & 3 \\ 4 & 1 & 5 \end{bmatrix} > \text{Whole Space} = 3 \text{ 차원} / \text{Column Space} = 2 \text{ 차원}$$

$$(\text{Col1} + \text{Col2} = \text{Col3})$$

$$A = \begin{bmatrix} 1 & 2 & 4 \\ 2 & 4 & 8 \\ 4 & 8 & 16 \end{bmatrix} > \text{Whole Space} = 3 \text{ 차원} / \text{Column Space} = 1 \text{ 차원}$$

$$(2*\text{Col1} = \text{Col2} / 2*\text{Col2} = \text{Col3})$$

$$A = \begin{pmatrix} 1 & 3 \\ 2 & 3 \\ 4 & 1 \end{pmatrix} > \text{Whole Space} = 3 \text{ 차원} / \text{Column Space} = 2 \text{ 차원}$$

(independent 한 column 2 개)

$$A^T = \begin{pmatrix} 1 & 2 & 4 \\ 3 & 3 & 1 \end{pmatrix} \text{ (transpose)} > \text{Whole Space} = 2 \text{ 차원} / \text{Column Space} = 2 \text{ 차원}$$

(Column Space 는 Whole Space 보다 차원이 클 수 없음)

Column space 는 Column vector 보다 차원이 높을 수 없음

- Four spaces in a matrix: two whole spaces R^m / R^n

$$A = \begin{pmatrix} 1 & 2 \\ 2 & 4 \\ 3 & 6 \end{pmatrix} > \text{Whole Space(column)} = 3 \text{ 차원} / \text{Whole Space(row)} = 2 \text{ 차원}$$

Column Space = 1 차원 / Row Space = 1 차원

Independent 한 column/row 의 개수: rank (column 과 row 의 rank 는 같음)

Null Space: Whole Space 에서 사용하지 않는 차원

Column space 와 orthogonal 한 space (기하학적 해석)

$$A_x = \begin{pmatrix} 1 & 2 \\ 2 & 3 \\ 4 & 6 \end{pmatrix} * \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} > \text{null space} \begin{pmatrix} 0 & 2 & -2 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ -\frac{1}{2} \end{pmatrix} \dots \text{ (수학적 해석)}$$

- Linear Transformation: $A(\text{행렬}) * x = b(\text{벡터})$ (기계 * 입력 = 출력)

입력 벡터와 출력 벡터의 차원은 다를 수 있음

$$Av = b \text{ (A transforms v to b)}$$

- Detransformation: Inverse matrix

$$A^{-1} * b = x$$

dependent column > not invertible

$$\begin{bmatrix} a & c \\ b & d \end{bmatrix} > a*d - b*c = 0 \text{ 이 되는 경우 not invertible}$$

- Eigenvector: transformation 후 원점과 일직선 상에 있는 vector (기하학적 해석)

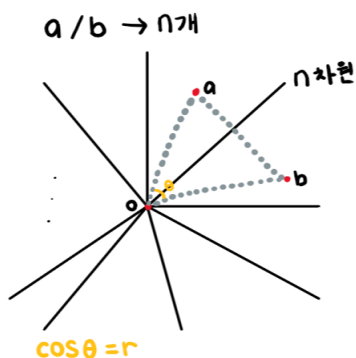
Given matrix 의 Eigenvector 는 transformation 후의 결과값이 전의 벡터와 원점과 같은 선상에 있음

among all v, some v is parallel to Av > That v is eigenvector

$$Av = \lambda v \text{ (transformed 된 후의 vector 가 상수를 곱한 vector 와 같다)}$$

Eigenvalue: 비율(λ)

- 상관 관계(correlation)



두 개의 변수가 비례하는 관계

$$-1 \leq r \leq 1$$

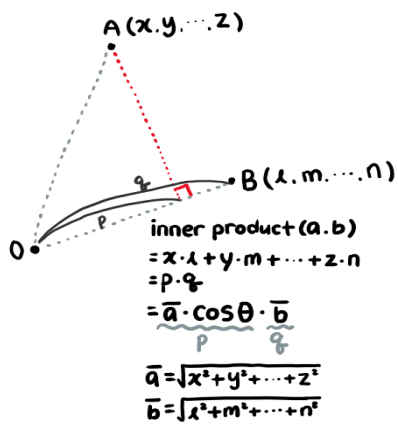
 $r=0$ 일 때 상관관계가 가장 낮음

일직선에 가까울수록 상관관계 높음

$$\cos \theta = r$$

$$\theta = 90 > r = 0 / \theta = 0 > r = 1$$

- inner product

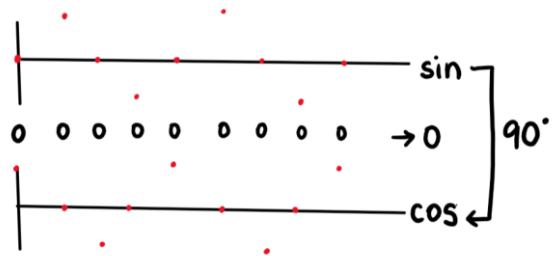
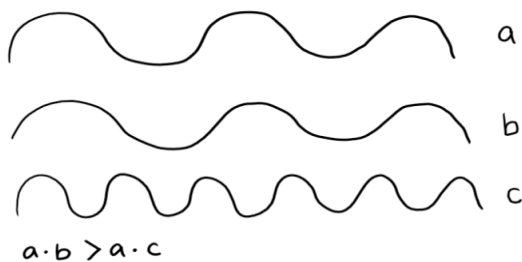


correlation 이 높을수록 inner product 가 큼

$A = [x, y, z]$ 1 by 3 / $B = [l, m, n]$ 1 by 3

$A \cdot B$ 불가능 > $A \cdot B^T = 1$ by 1 의 vector (Inner product)

$\cos \theta = \cos \text{similarity}$: A 와 B 가 얼마나 비슷한지



맹점: sin, cos phasor 는 같은 frequency 의 wave 를 shift 할 경우 값이 달라짐

> complex phasor($e^{i\theta}$)를 사용하여 inner product (complex 값)

> complex number plotting 불가능하므로 절댓값 ($a+bi$ 의 절댓값 = (a,b) 의 길이)