2016/10/14
황준식

[딥러닝없음주의]

알파고의 시간

발표자 소개

황준식 @ NEXON KOREA
JUNIOR DATA ANALYST
JSIDEAS.NET

오늘의 주제

Confusion Matrix	y_pred: True	y_pred: False
y_true: True	알파고에 관한 개인프로젝트 소개	흥미로울수도 있는 약간의 이야기
y_true: False	알파고에 관한 이론적 배경이나 구현에 관한 정리	5분만에 배우는 텐서플로우

source: New Yorker

새로운 주제는 아닙니다

알파고와 딥러닝

nature

THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE

At last — a computer program that can beat a champion Go player PAGE 484

ALL SYSTEMS GO

CONSERVATION

SONGBIRDS À LA CARTE

gal harvest of millions f Mediterranean birds

PAGE 452

RESEARCH ETHICS

SAFEGUARD TRANSPARENCY

Don't let openness backfire on individuals

PAGE 459

POPULAR SCIENCE

WHEN GENES GOT 'SELFISH'

Dawkins's calling card forty years on

PAGE 462

NATURE.COM/NATURE

28 January 2016 £10

Vol. 529, No. 7587

9 770028 083095

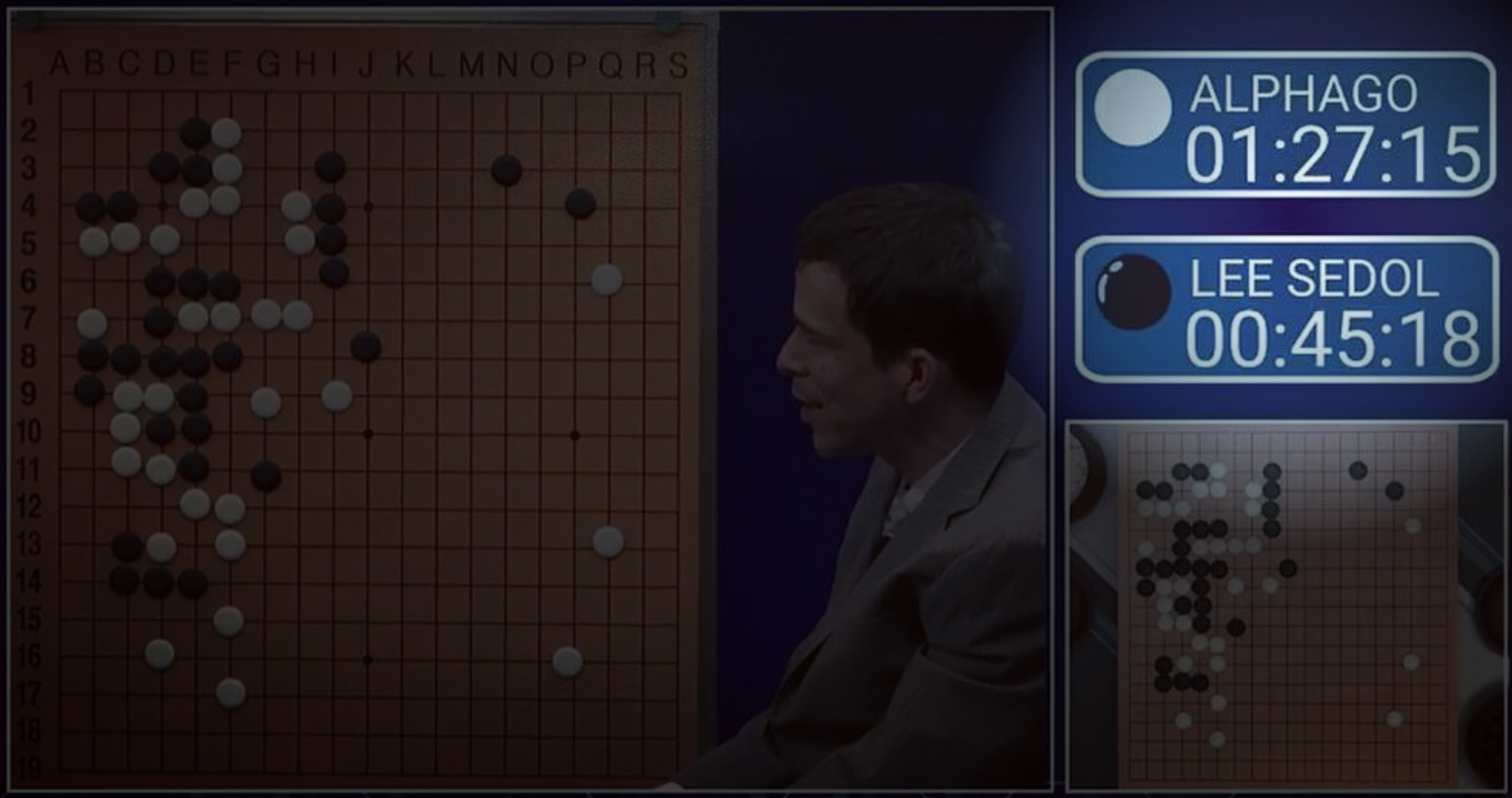
Machines 3, Man 0

QUESTION 1.

인간과 기계는 어떻게 다른가?

ALPHAGO
01:27:15

LEE SEDOL
00:45:18


A B C D E F G H I J K L M N O P Q R S
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
ALPHAGO
01:27:15
LEE SEDOL
00:45:18

alphago lee sedol time spent - ×

Junsik

https://www.google.com/search?q=alphago+lee+sedol+time+spent&espv=2&biw=1440&bih=733&source=lnms&sa=X&ved=0ahUKEwjDoJLtjLfPAhVC3GMKHeLkB9kQ_A...

Apps Gists GA ColorBrewer CodeFights ŷhat | Random Fore... TuneIn Loading Data and B... Tidy Data 07-tidy-data.pdf Quantifying the Invi... data-wrangling-che... Other Bookmarks

Google alphago lee sedol time spent

All News Images Videos More Search tools

About 34,600 results (0.29 seconds)

Lee Sedol vs. AlphaGo: 4th game - Thinking Time in minutes [OC ...
https://www.reddit.com/r/.../**lee_sedol**_vs_**alphago**_4th_game_thinking_**time**_in/
Mar 13, 2016 - **AlphaGo**: 4th game - Thinking **Time** in minutes [OC] (i.imgur.com) .... Nice, can you add the point at which **Lee Sedol** entered overtime? ... to go over that and **spend** an extra minute if a particularly complex situation arose.

Fascinating Insight into Alpha Go's from Match 4 : baduk - Reddit
https://www.reddit.com/r/baduk/.../fascinating_insight_into_**alpha_go**s_from_match_...
Mar 13, 2016 - In the face of defeat, **Lee Sedol spent** a good 40 minutes to come up with ..... low probability move it needs to **spend** a lot more **time** analyzing it.

CB News - AlphaGo vs Lee Sedol: history in the making | Chess News
https://en.chessbase.com/post/**alphago**-vs-**lee-sedol**-history-in-the-making
Mar 13, 2016 - An overview of the third game from **AlphaGo** vs **Lee Sedol**. ..... article in a field (neuroscience, neural nets, AI) that I **spent** some **time** once upon ...

Lee Sedol defeats AlphaGo in masterful comeback - Game 4
https://gogameguru.com/**lee-sedol**-defeats-**alphago**-masterful-comeback-game-4/
Mar 13, 2016 - **Lee Sedol** plays his first move as White, in his fourth game against ..... and the search starts turning to **spend** more **time** to different moves, not ...

Thinking time charts for AlphaGo vs Lee Sedol • Life In 19x19
lifein19x19.com › Board index › Go Gear › Computer Go
Mar 11, 2016 - 6 posts - 5 authors
The spike for **Lee's** 19th move is him thinking how to respond to **AlphaGo's** unusual ... **AlphaGo spent** the most **time** on White 61 aka White 122 ...

Why the Final Game Between AlphaGo and Lee Sedol Is Such a Big ...
https://www.wired.com/2016/.../final-game-**alphago-lee-sedol**-big-deal-humanit... Wired

착점시간 데이터 검색

▼

구글에 없어...?

▼

PROFIT!!

Table 1

turn_index	Lee_Sedol	AlphaGo
0	02:00:00	02:00:00
1	01:59:41	01:58:30
2	01:58:15	01:57:24
3	01:57:00	01:56:05
4	01:55:41	01:55:08
5	01:54:30	01:54:08
6	01:54:19	01:53:16
7	01:43:08	01:44:08
…	…	…

Table 1-1

turn_index	Lee_Sedol	AlphaGo
1	19	90
2	86	56
3	75	79
…	…	…

ALPHAGO 01:27:15

LEE SEDOL 00:45:18

원하는 데이터셋만

명확하면 노가다는 금방

Table 1

turn_index	AlphaGo	Lee_Sedol	countdown	AlphaGo_ott	Lee_Sedol_ott
41	01:12:29	00:03:23			
42	01:11:09	00:02:18			
43	01:09:59	00:00:53			
44	01:08:15	00:00:38			
45	01:07:04	00:00:02	lee -1		00:01:36
46	01:05:02	00:00:01	lee -1		00:01:59

```
def time_func(x):
    k = x.split(":")
    hour = int(k[0])
    minute = int(k[1])
    second = int(k[2])
    ts = hour * 3600 + minute * 60 + second
    return ts

df_alpha['AlphaGo_ts'] = df_alpha.AlphaGo.apply(lambda x: time_func(x))
df_lee['Lee_Sedol_ts'] = df_lee.Lee_Sedol.apply(lambda x: time_func(x))
```

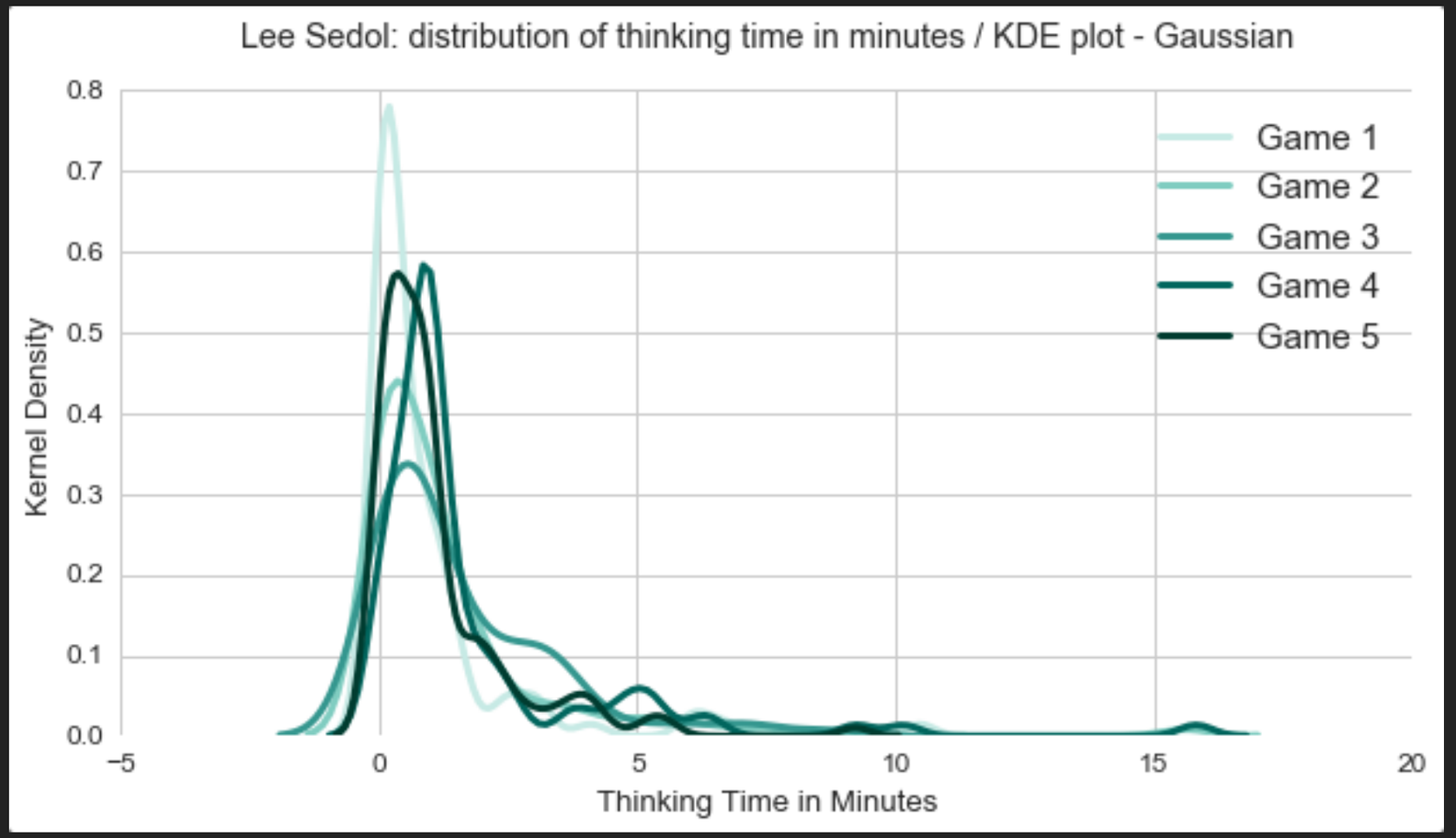
분:초 STRING -> 초 INTEGER

```
result_df_2.columns = ['turn_index', 'player', 'thinking_time']
result_df_2.head()
```

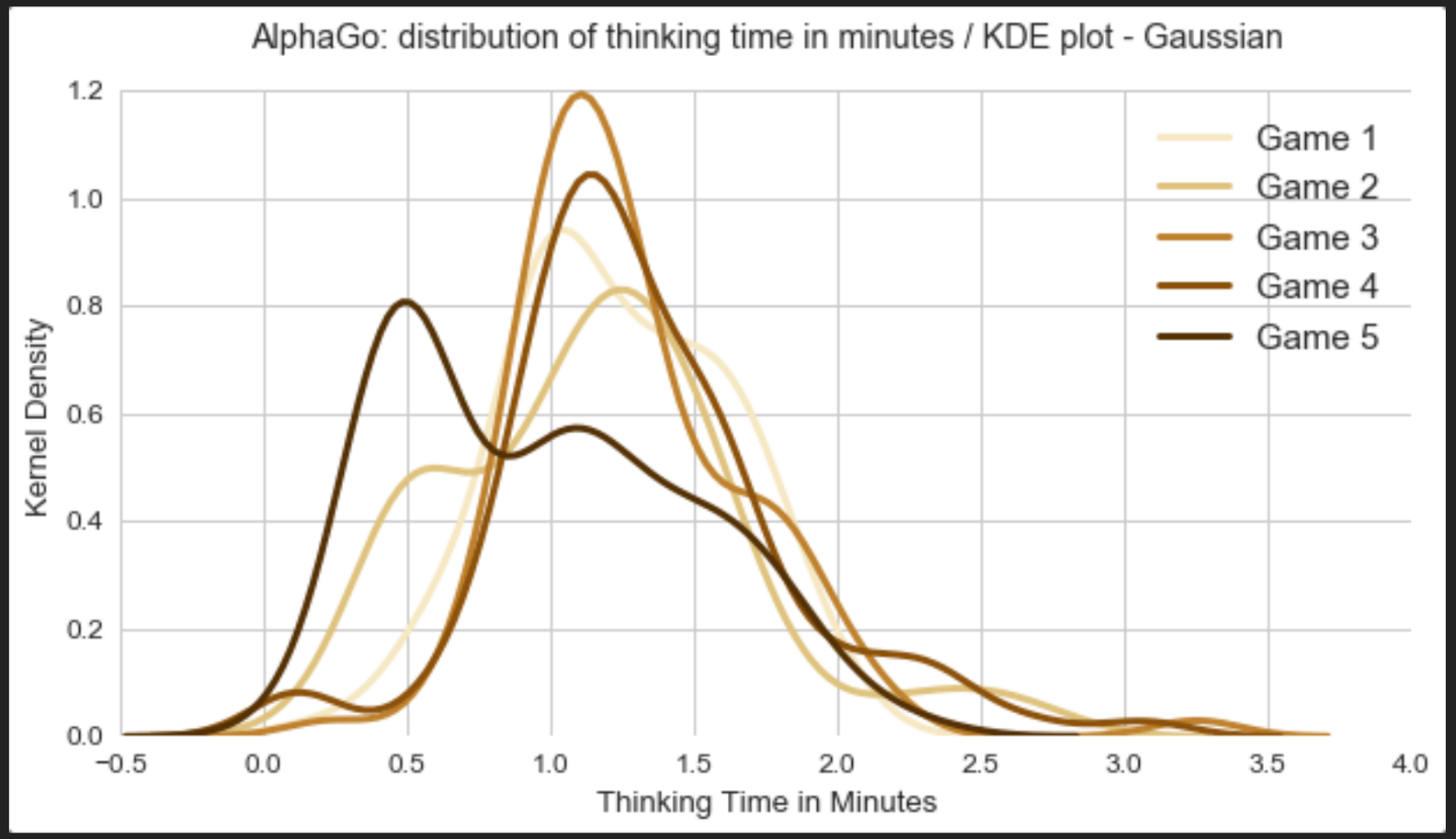
	turn_index	player	thinking_time
0	1	AlphaGo_WHITE	85
1	2	AlphaGo_WHITE	68
2	3	AlphaGo_WHITE	53
3	4	AlphaGo_WHITE	62
4	5	AlphaGo_WHITE	73

최종 데이터셋의 형태

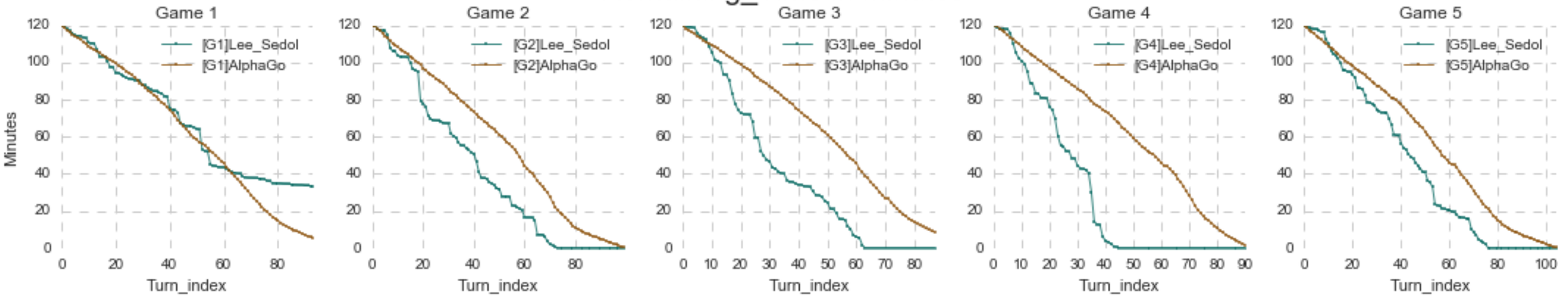
LEE SEDOL
MEAN: 79 SEC
STD: 111 SEC
MAX: 949 SEC



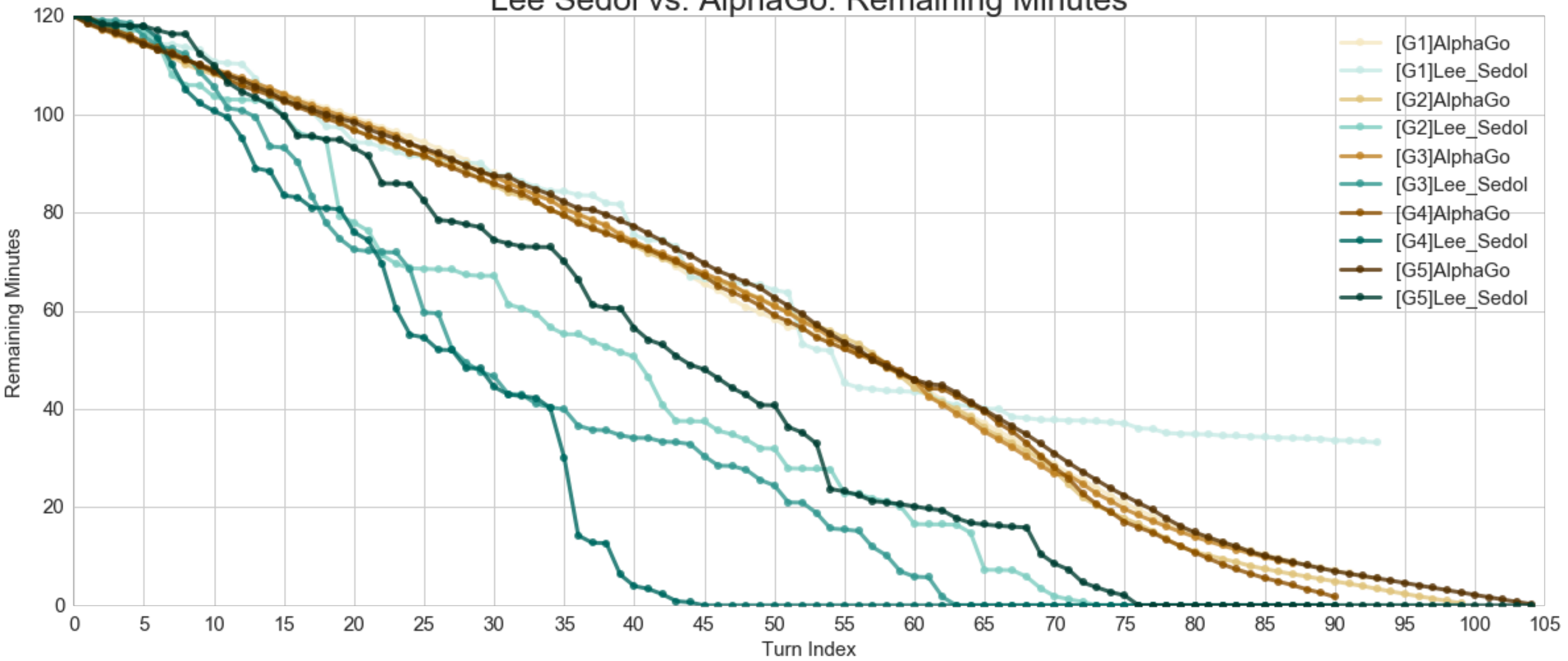
ALPHAGO
MEAN: 70 SEC
STD: 29 SEC
MAX: 195 SEC



Remaining_Time in Minutes

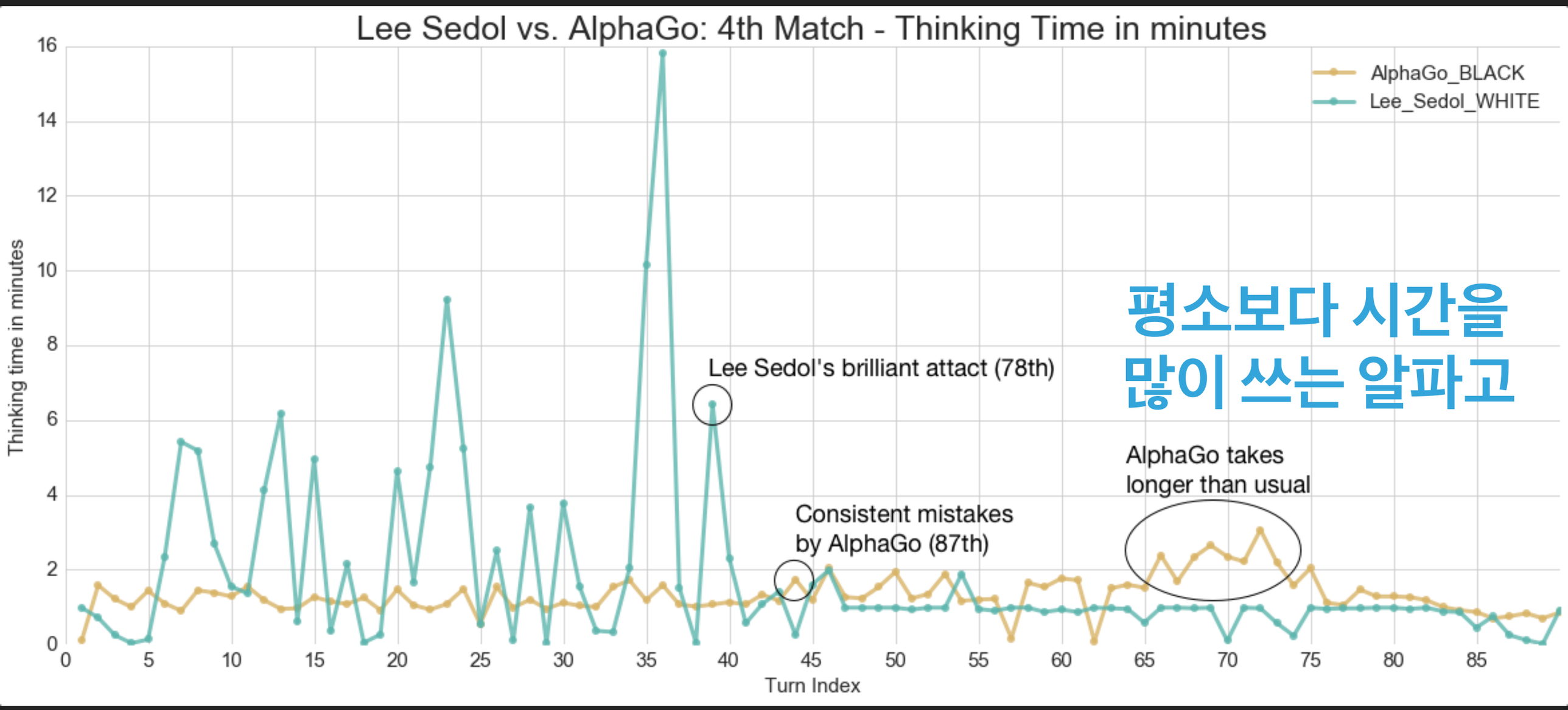


Lee Sedol vs. AlphaGo: Remaining Minutes



http://jsideas.net/python/2016/03/16/Google_DeepMind.html

착점시간과 이세돌의 승리는 무슨 관계?



착점시간과 이세돌의 승리는 무슨 관계?

Q. 알파고가 수에 따라 두는 시간이 다른가? 또 타이젬을 통해 훈련을 했는가

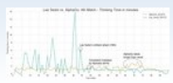
하사비스 : 알파고는 시간을 계산한다. 어려운 수는 더 많은 시간을 할애한다. 타이젬은 개발자의 개인적인 계정이다.

출처: 이세돌 "알파고에 지더라도 바둑의 아름다움은 계속될 것”(일문일답) (스포츠투데이)

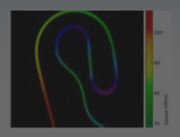
뜻밖의 관심과 트래픽

REDDIT

3477 
Half of the US Population [OC] imgur.com
submitted 9 months ago by pf_throwaway811
642 comments share save hide report pocket

3474 
Lee Sedol vs. AlphaGo: 4th game - Thinking Time in minutes [OC] i.imgur.com
submitted 7 months ago by junkwhinger
365 comments share save hide delete nsfw pocket

3471
Mathematicians created a Social Network map to decide who the Main Character was in the A Song of Ice and Fire (Game of Thrones) series (PDF) maa.org
submitted 6 months ago by Hyperdrunk
634 comments share save hide report pocket

3463 
Plot of my speed on track during a race on Gateway Motorsports Park's road course [OC] i.imgur.com
submitted 6 months ago by gamerdonkey
242 comments share save hide report pocket

3463 
TV shows that got better over time and TV shows that remained most consistent [OC] imgur.com
submitted 3 months ago by vedangmehta
1580 comments share save hide report pocket

3452 
An eye opening video about the distribution of wealth in the US youtu.be
submitted 11 months ago by smoke_memes_everyday
1552 comments share save hide report pocket

3444 
1100 declassified U.S. nuclear targets futureoflife.org
submitted 4 months ago by MrBeanie88
555 comments share save hide report pocket

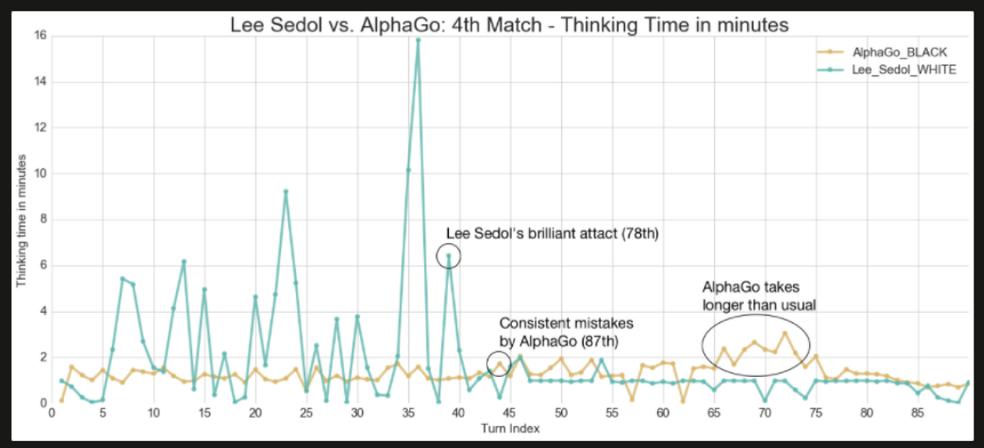
IMGUR

image 4 of 59

Name: fourth_gam... Views: 702,342 Submitted: 7 months ago Bandwidth: 78.18 GB

image

VIEWS: 702,342



TOTAL: 900,000

edit title/description | edit image

View and edit post

Image Link

http://imgur.com/3HcJKbk

Direct Link

http://i.imgur.com/3HcJKbk.png

Markdown Link (reddit comments)

[Imgur](http://i.imgur.com/3HcJKbk.png)

HTML (website / blogs)

<a href="http://imgur.com/3HcJKbk"><in

BBCode (message boards & forums)

[img]http://i.imgur.com/3HcJKbk.png[/in

Linked BBCode (message boards)

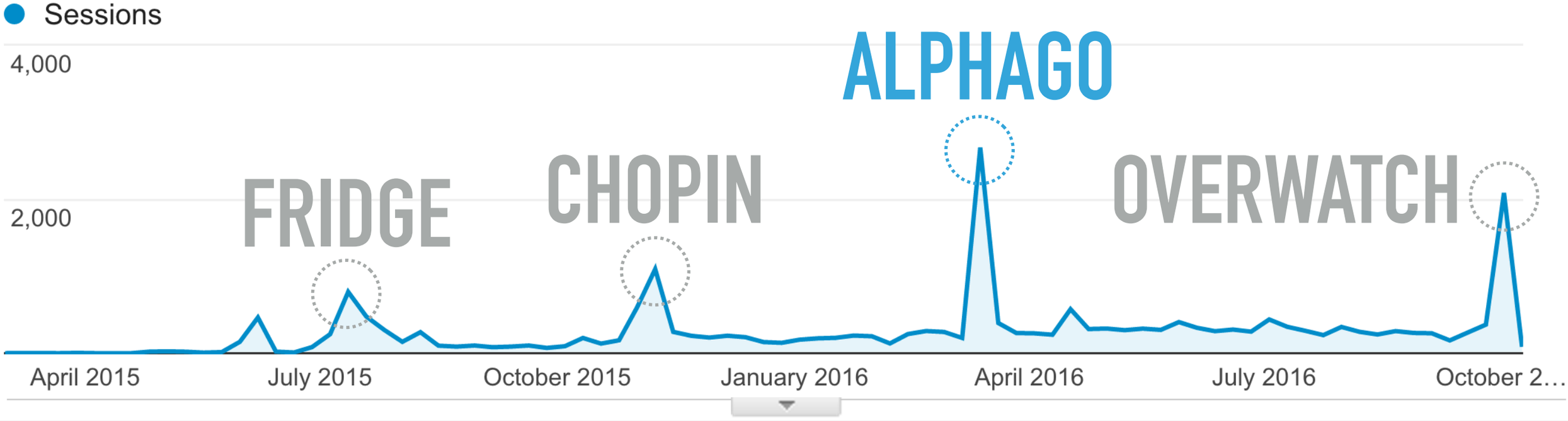
[url=http://imgur.com/3HcJKbk][img]http

Sizes: Original · Small Square · Big Square · Small Thumbnail · Medium Thumbnail · Large Thumbnail · Huge Thumbnail

JSIDEAS.NET

Sessions vs Select a metric

Hourly | Day | Week | Month



EXTRA QUESTION

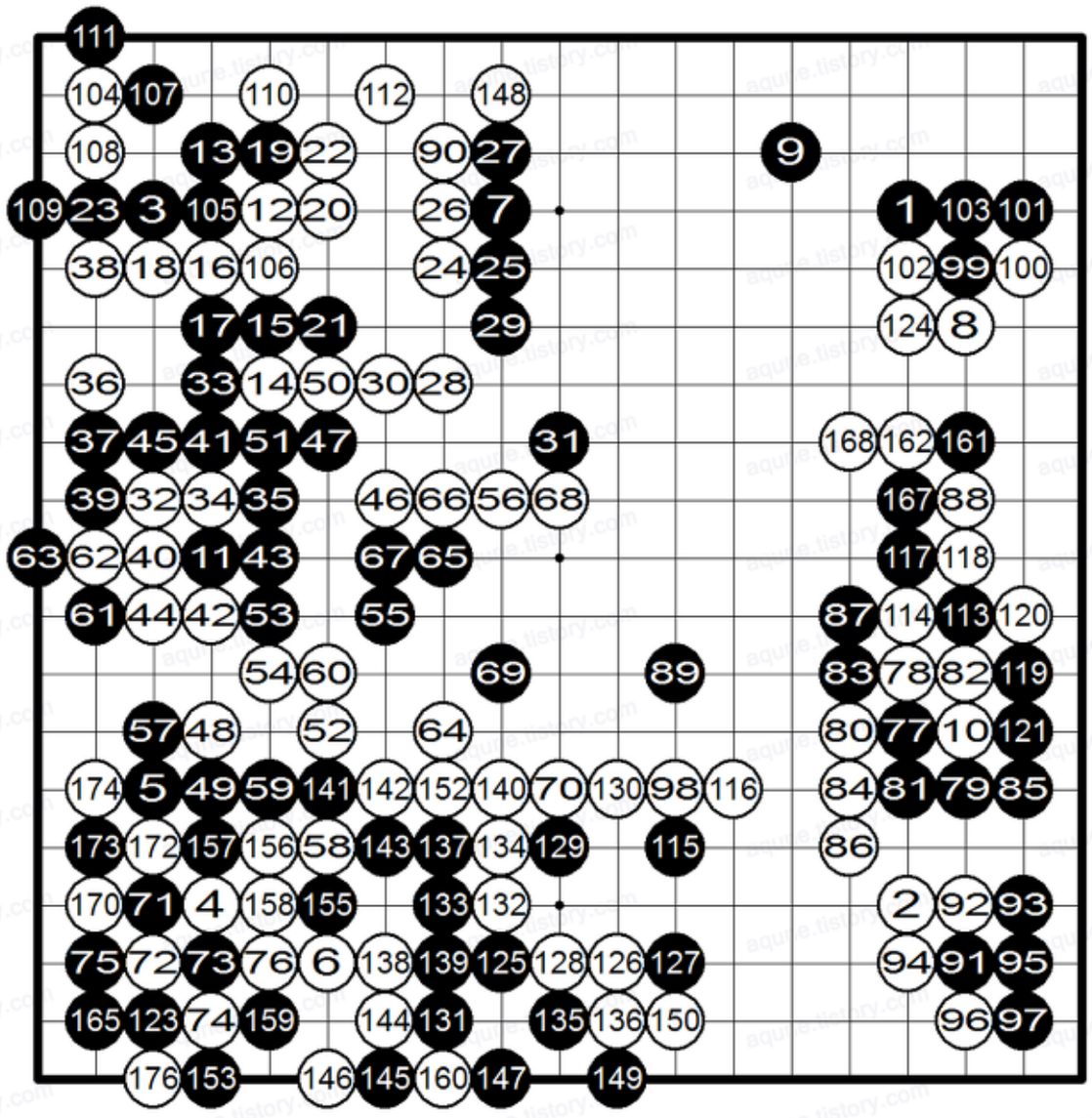
바둑의 흐름도 **정량화**할 수 있나?

Google DeepMind Challenge Match

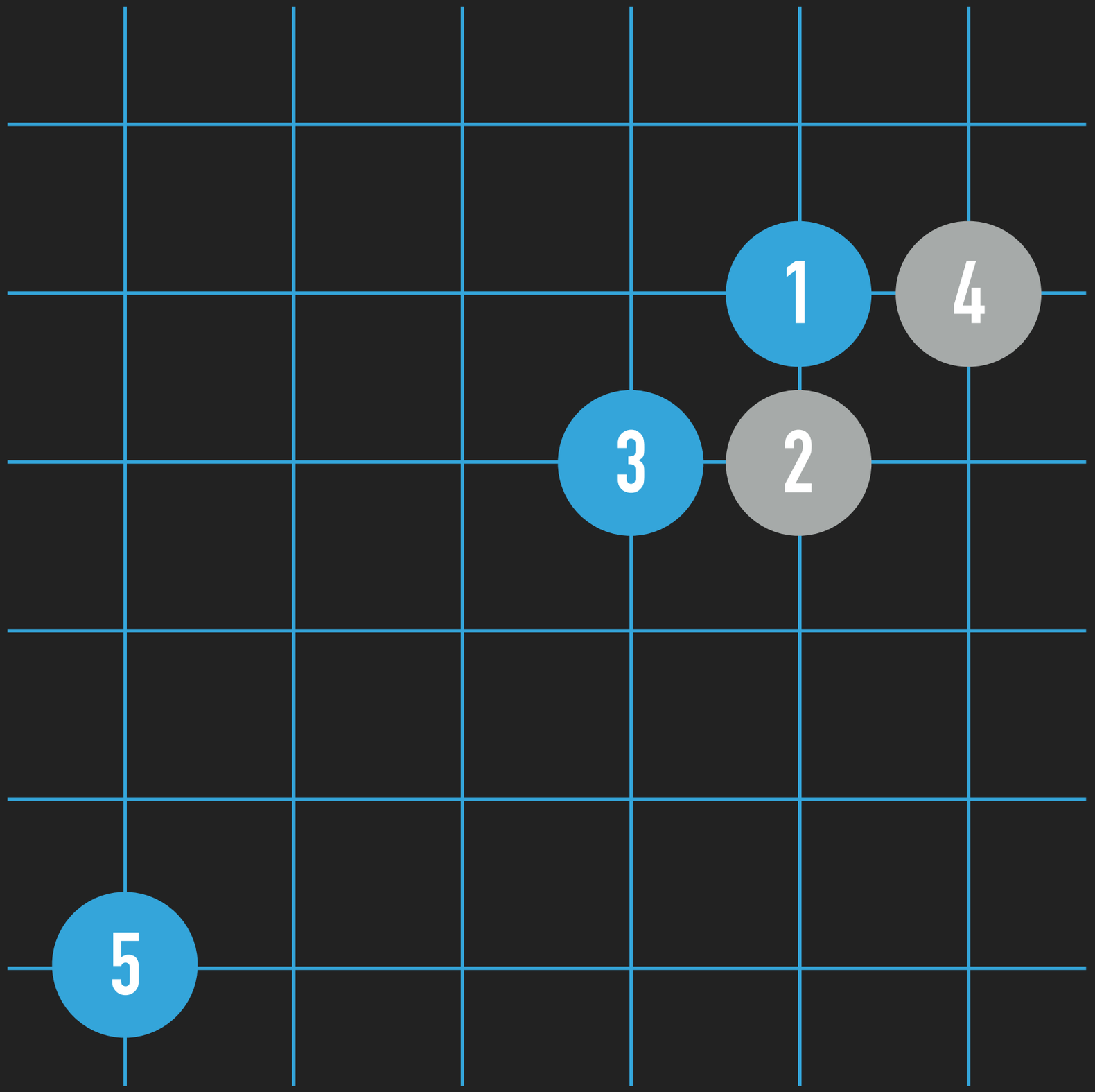
8 - 15 March 2016

AlphaGo vs Lee Sedol

제3국 : 2016년 3월 12일 / 흑-이세돌 vs. 백-알파고 / 176수 백 불계승



C16: 71 ▶ 175 C17: (72) ▶ 154 D17: 73 ▶ 151 164 Q11: 113 ▶ (122) G19: 145 ▶ 163 169 H19: (160) ▶ 166 171


1
4
3
2
5

19X19 바둑판

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
r																			
1	NaN	111.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	NaN	104.0	107.0	NaN	110.0	NaN	112.0	NaN	148.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	NaN	108.0	NaN	13.0	19.0	22.0	NaN	90.0	27.0	NaN	NaN	NaN	NaN	9.0	NaN	NaN	NaN	NaN	NaN
4	109.0	23.0	3.0	105.0	12.0	20.0	NaN	26.0	7.0	NaN	NaN	NaN	NaN	NaN	NaN	1.0	103.0	101.0	NaN
5	NaN	38.0	18.0	16.0	106.0	NaN	NaN	24.0	25.0	NaN	NaN	NaN	NaN	NaN	NaN	102.0	99.0	100.0	NaN
6	NaN	NaN	NaN	17.0	15.0	21.0	NaN	NaN	29.0	NaN	NaN	NaN	NaN	NaN	NaN	124.0	8.0	NaN	NaN
7	NaN	36.0	NaN	33.0	14.0	50.0	30.0	28.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
8	NaN	37.0	45.0	41.0	51.0	47.0	NaN	NaN	NaN	31.0	NaN	NaN	NaN	NaN	168.0	162.0	161.0	NaN	NaN
9	NaN	39.0	32.0	34.0	35.0	NaN	46.0	66.0	56.0	68.0	NaN	NaN	NaN	NaN	NaN	167.0	88.0	NaN	NaN
10	63.0	62.0	40.0	11.0	43.0	NaN	67.0	65.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	117.0	118.0	NaN	NaN
11	NaN	61.0	44.0	42.0	53.0	NaN	55.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	87.0	114.0	113.0	120.0	NaN
12	NaN	NaN	NaN	NaN	54.0	60.0	NaN	NaN	69.0	NaN	NaN	89.0	NaN	NaN	83.0	78.0	82.0	119.0	NaN
13	NaN	NaN	57.0	48.0	NaN	52.0	NaN	64.0	NaN	NaN	NaN	NaN	NaN	NaN	80.0	77.0	10.0	121.0	NaN
14	NaN	174.0	5.0	49.0	59.0	141.0	142.0	152.0	140.0	70.0	130.0	98.0	116.0	NaN	84.0	81.0	79.0	85.0	NaN
15	NaN	173.0	172.0	157.0	156.0	58.0	143.0	137.0	134.0	129.0	NaN	115.0	NaN	NaN	86.0	NaN	NaN	NaN	NaN
16	NaN	170.0	71.0	4.0	158.0	155.0	NaN	133.0	132.0	NaN	NaN	NaN	NaN	NaN	NaN	2.0	92.0	93.0	NaN
17	NaN	75.0	72.0	73.0	76.0	6.0	138.0	139.0	125.0	128.0	126.0	127.0	NaN	NaN	NaN	94.0	91.0	95.0	NaN
18	NaN	165.0	123.0	74.0	159.0	NaN	144.0	131.0	NaN	135.0	136.0	150.0	NaN	NaN	NaN	NaN	96.0	97.0	NaN
19	NaN	NaN	176.0	153.0	NaN	146.0	145.0	160.0	147.0	NaN	149.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

수 / 위치 정보 추출

```
stone_list = []
for i in range(1, 20):
    col_num = str(i)
    for j in range(0, 19):
        stone_dict = {}
        temp = j
        row_num = j + 1
        stone_value = data[col_num].iloc[temp]
        stone_dict['value'] = stone_value.astype(int)
        stone_dict['col'] = int(col_num)
        stone_dict['row'] = int(row_num)
        if np.isnan(stone_value):
            pass
        else:
            stone_list.append(stone_dict)
```

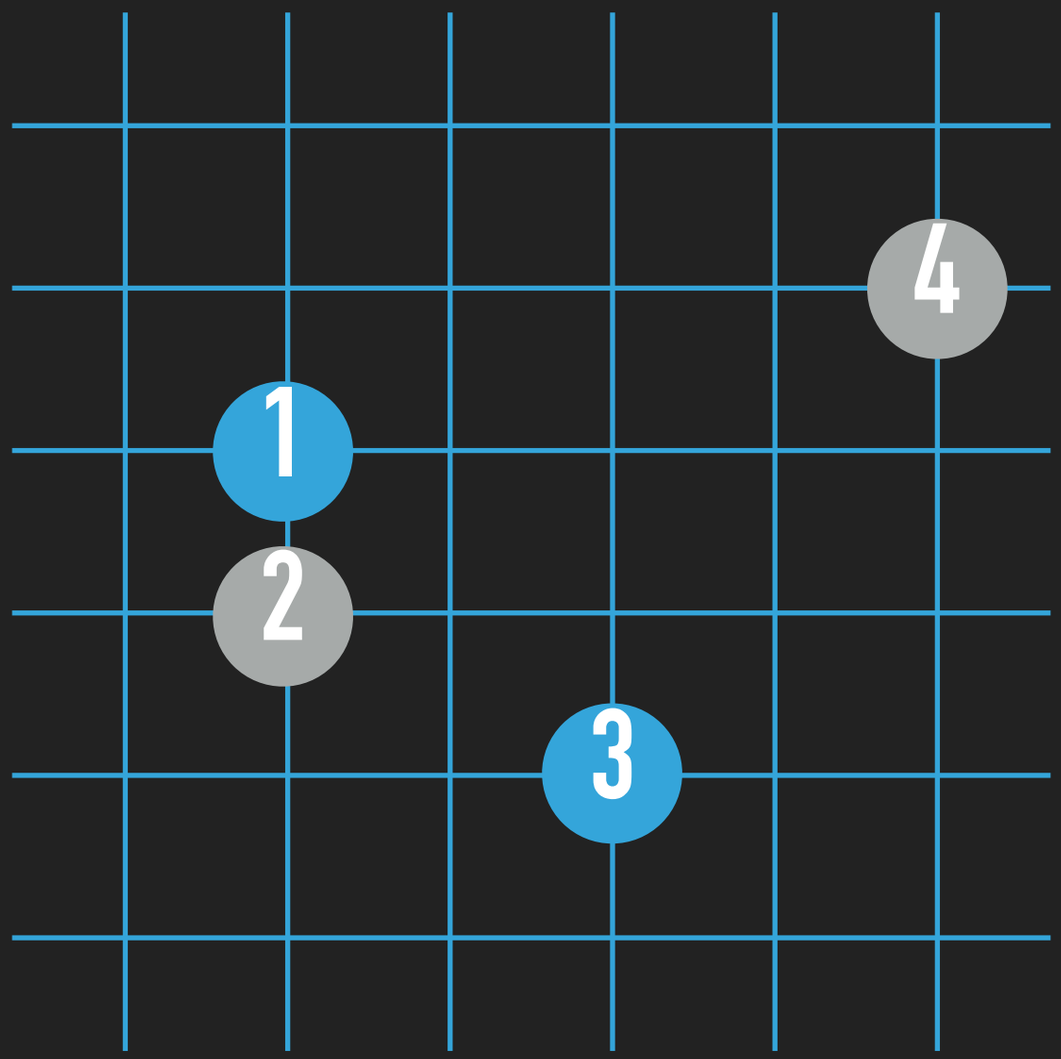
```
stone_175 = {'value':175, 'col':3, 'row':16}
stone_154 = {'value':154, 'col':3, 'row':17}
stone_151 = {'value':151, 'col':4, 'row':17}
stone_164 = {'value':164, 'col':4, 'row':17}
stone_122 = {'value':122, 'col':17, 'row':11}
stone_163 = {'value':163, 'col':7, 'row':19}
stone_169 = {'value':169, 'col':7, 'row':19}
stone_166 = {'value':166, 'col':8, 'row':19}
stone_171 = {'value':171, 'col':8, 'row':19}
```

DF의 행렬을 돌면서
수 / 행 / 열 DICT를
LIST에 저장

단수 등으로 기존
위치에 추가된
수는 추가 입력

직전 수와의 거리 산출 (맨하탄 거리)

```
FROM SCIPY.SPATIAL.DISTANCE IMPORT CITYBLOCK
```



격자위의 가로세로
이동 거리의 합

1 -> 2: 1

2 -> 3: 3

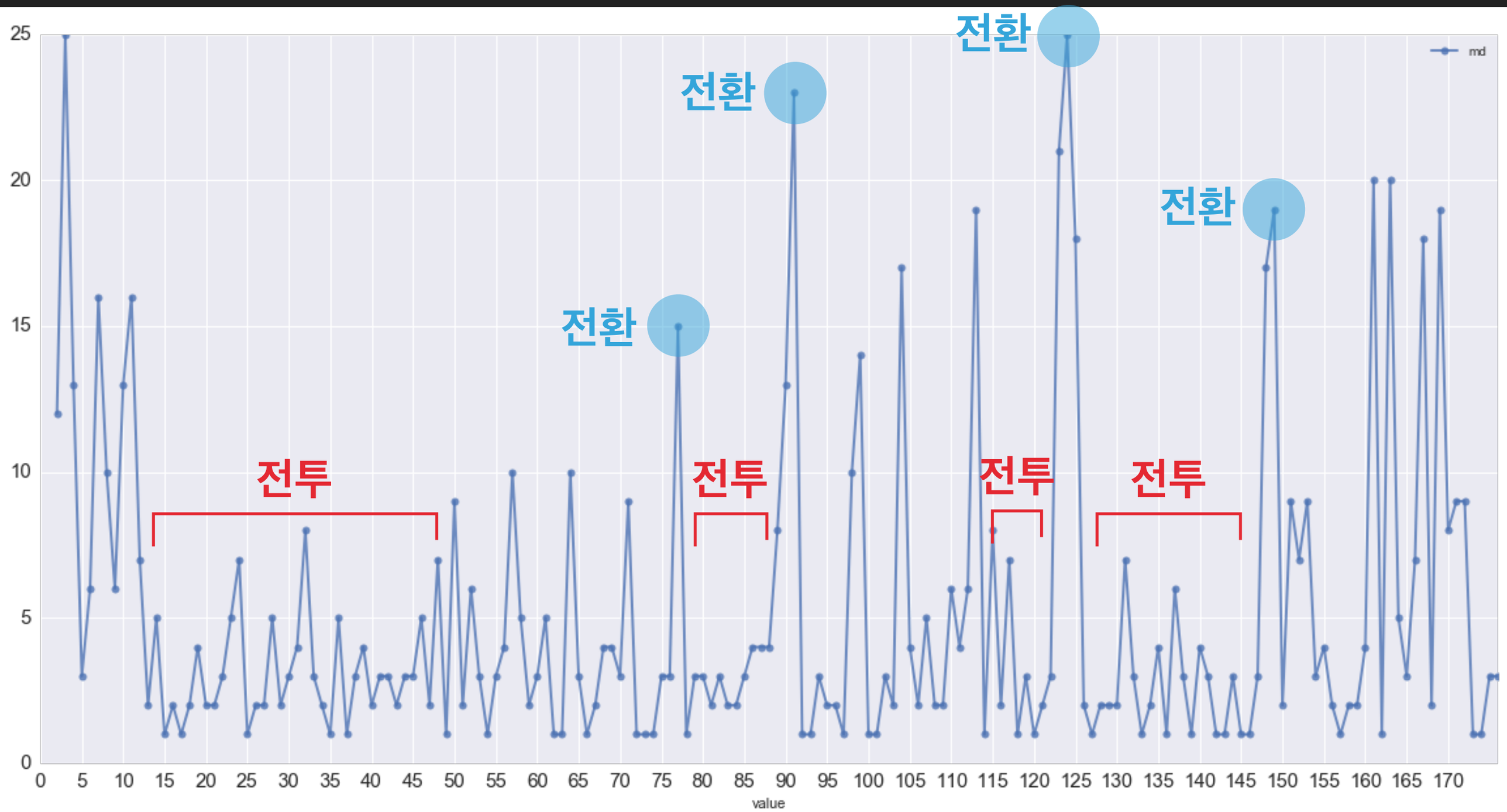
3 -> 4: 5

데이터 가공

```
## row와 col을 합쳐 list로 column에 저장
t['loc'] = t[['row', 'col']].values.tolist()


## 직전 row의 loc 정보를 prev_loc에 저장
t['prev_loc'] = t['loc'].shift(1)


## df의 row를 돌면서 loc과 prev_loc의 맨하탄 거리 계산
t['md'] = t.apply(lambda row:
                  cityblock(row['loc'], row['prev_loc']),
                  axis=1)
```

	value	row	col	loc	prev_loc	md
0	1	4	16	[4, 16]	NaN	NaN
1	2	16	16	[16, 16]	[4, 16]	12.0
2	3	4	3	[4, 3]	[16, 16]	25.0
3	4	16	4	[16, 4]	[4, 3]	13.0
4	5	14	3	[14, 3]	[16, 4]	3.0
5	6	17	6	[17, 6]	[14, 3]	6.0
6	7	4	9	[4, 9]	[17, 6]	16.0
7	8	6	17	[6, 17]	[4, 9]	10.0
8	9	3	14	[3, 14]	[6, 17]	6.0
9	10	13	17	[13, 17]	[3, 14]	13.0
10	11	10	4	[10, 4]	[13, 17]	16.0
11	12	4	5	[4, 5]	[10, 4]	7.0
12	13	3	4	[3, 4]	[4, 5]	2.0
13	14	7	5	[7, 5]	[3, 4]	5.0
14	15	6	5	[6, 5]	[7, 5]	1.0

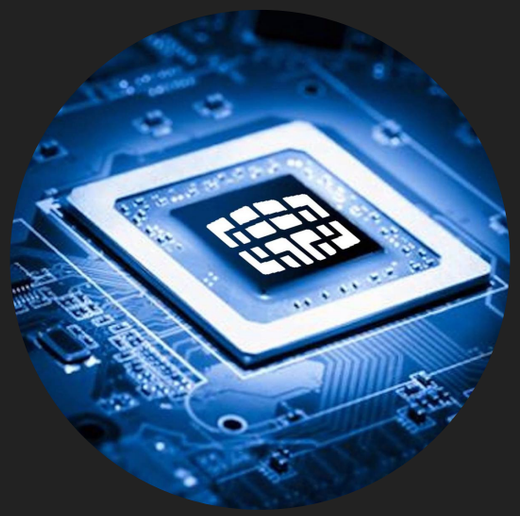
제 3국 흐름 차트


md
전환
전환
전환
전환
전환
전투
전투
전투
전투
25
20
15
10
5
0
0 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100 105 110 115 120 125 130 135 140 145 150 155 160 165 170
value

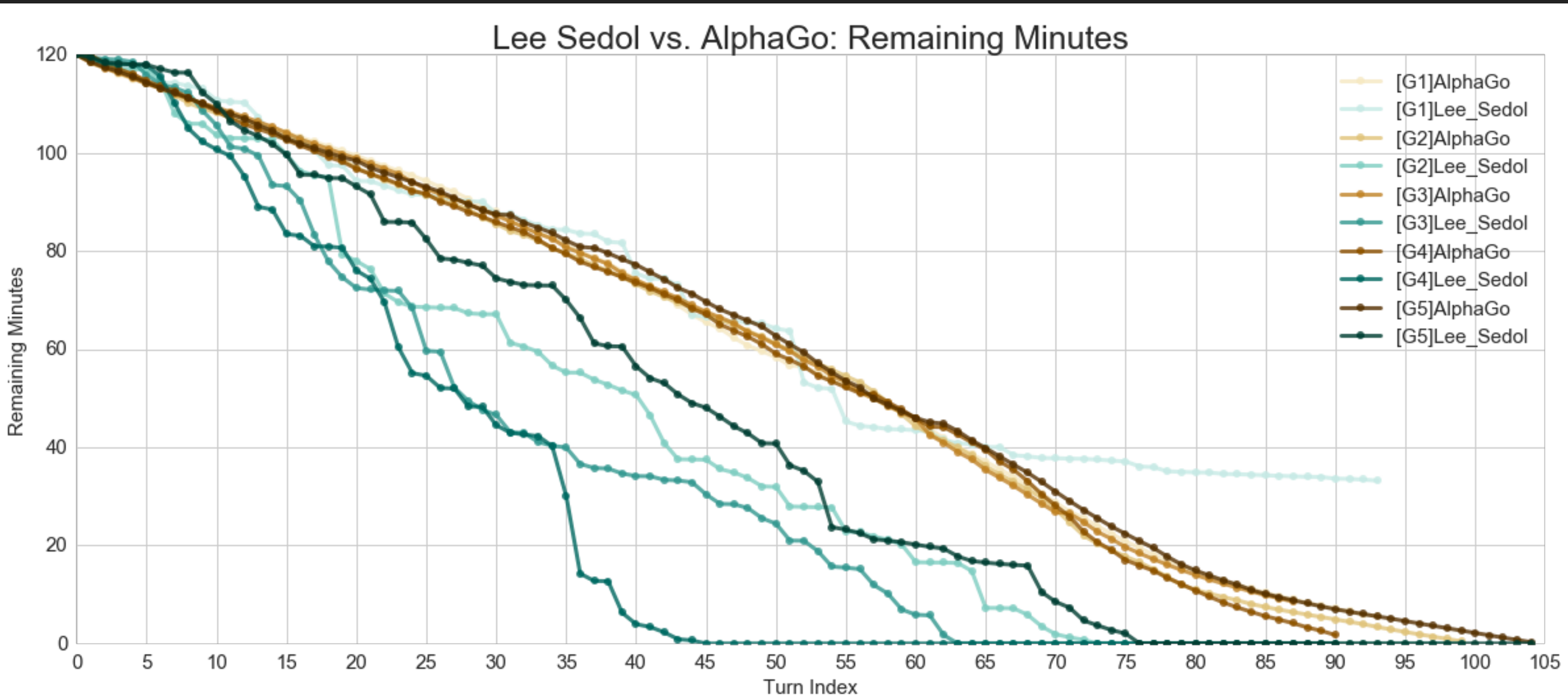
QUESTION 2.

인간과 기계를 어떻게 분류할까?

TURING TEST FOR GO



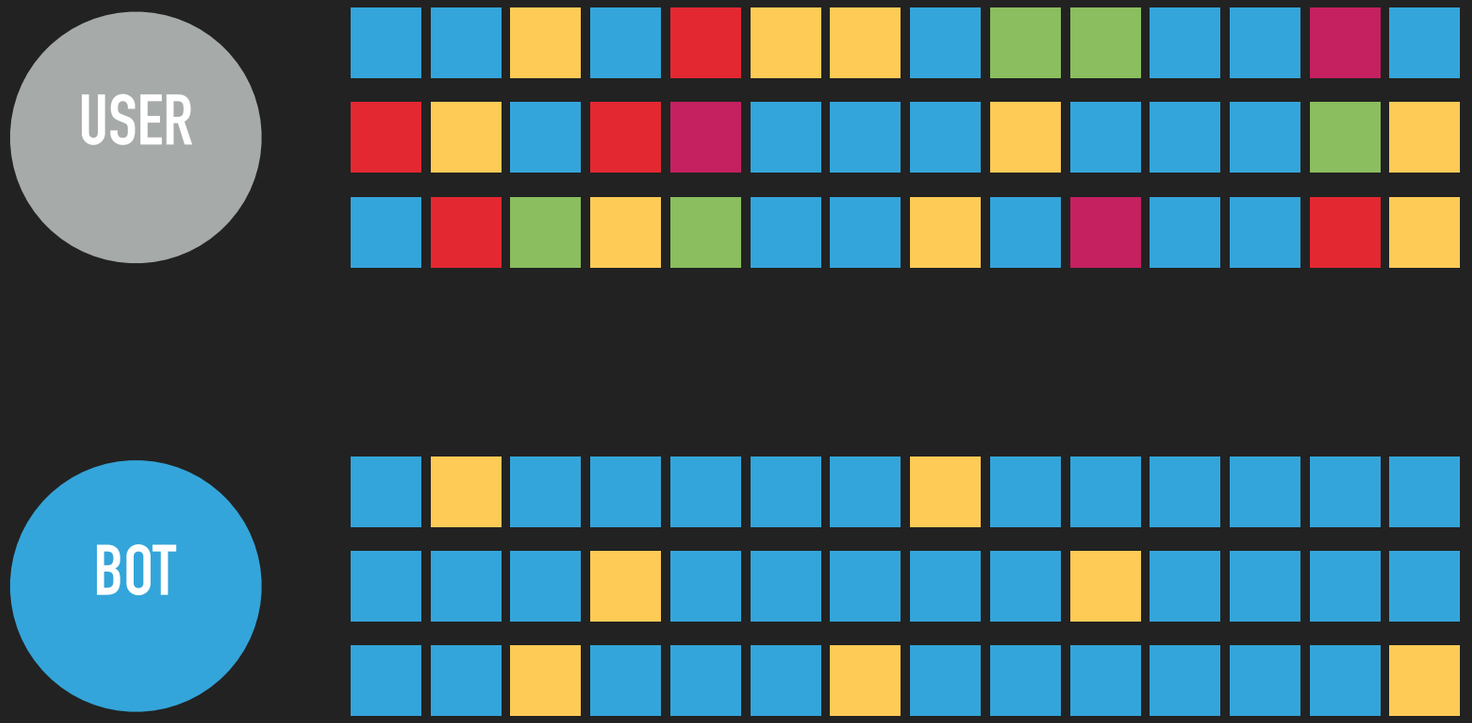
VS.



이세돌과 알파고의 차이: 대국간 착점시간의 편차
=> 자신의 플레이가 **서로 얼마나 유사한지** 비교해보자!

자기유사도 (SELF SIMILARITY)

MMORPG 게임에서 매크로 계정을 탐지하는 방법 중 하나



자기 유사도를 이용한 MMORPG 게임봇탐지 시스템
// 고려대학교 김휘강교수님 등

LEVENSHTEIN DISTANCE

- 두 문자열간의 유사도를 측정하는 방법
- S와 T라는 문자열이 주어졌을때,
  S가 T가 되기 위해 지우고, 추가하고, 교체하는
  문자열의 수만큼 거리가 계산됨
- 예) NEXON, NEXEN —> LD 1 (O -> E)
  DATA, DATUM —> LD 2 (A -> U, M)
  MOIST, CHOICE —> ??
- 활용분야: 스펠링체크, 음성인식, DNA 분석, 표절검색

HTTP://PEOPLE.CS.PITT.EDU/~KIRK/CS1501/ASSIGNMENTS/EDITDISTANCE/LEVENSHTEIN%20DISTANCE.HTM

LEVENSHTEIN DISTANCE

- STEP 1) N = LEN(S)
  M = LEN(T)
  IF N == 0: RETURN M
  IF M == 0: RETURN N
  0부터 M까지 행,
  0부터 N까지의 열로 된
  매트릭스를 구성

		M	O	I	S	T
	0	1	2	3	4	5
C	1					
H	2					
O	3					
I	4					
C	5					
E	6					

HTTP://PEOPLE.CS.PITT.EDU/~KIRK/CS1501/ASSIGNMENTS/EDITDISTANCE/LEVENSHTEIN%20DISTANCE.HTM

LEVENSHTEIN DISTANCE

- STEP 2) COST
  IF ROW == COLUMN: RETURN 1
  ELSE: RETURN 0

- STEP 3) TOTAL VALUE
  MIN(UP+1,
  LEFT+1,
  UPPER-LEFT + COST)

		M	O	I	S	T
	0	1	2	3	4	5
C	1	1				
H	2	2				
O	3	3				
I	4	4				
C	5	5				
E	6	6				

HTTP://PEOPLE.CS.PITT.EDU/~KIRK/CS1501/ASSIGNMENTS/EDITDISTANCE/LEVENSHTEIN%20DISTANCE.HTM

LEVENSHTEIN DISTANCE

- STEP 4) LD = D[M,N]

		M	O	I	S	T
	0	1	2	3	4	5
C	1	1	2	3	4	5
H	2	2	2	3	4	5
O	3	3	2	3	4	5
I	4	4	3	2	3	4
C	5	5	4	3	3	4
E	6	6	5	4	4	4

HTTP://PEOPLE.CS.PITT.EDU/~KIRK/CS1501/ASSIGNMENTS/EDITDISTANCE/LEVENSHTEIN%20DISTANCE.HTM

LEVENSHTEIN DISTANCE ON ALPHAGO

Table 1-1

turn_index	Lee_Sedol
1	19
2	86
3	75
4	2
5	67
6	450
...	...

▶ ABABECDE

HISTOGRAM의 BIN처럼 구간화하자!

LEVENSHTEIN DISTANCE ON ALPHAGO

쾌속 착점 ▶

조금 생각 ▶

약간 고민 ▶

고민 ▶

장고 ▶

```
def string_classifier(time_spent):
    if time_spent <= 10.0:
        return 'A'
    elif time_spent <= 40.0:
        return 'B'
    elif time_spent <= 120.0:
        return 'C'
    elif time_spent <= 300.0:
        return 'D'
    else:
        return 'E'
```

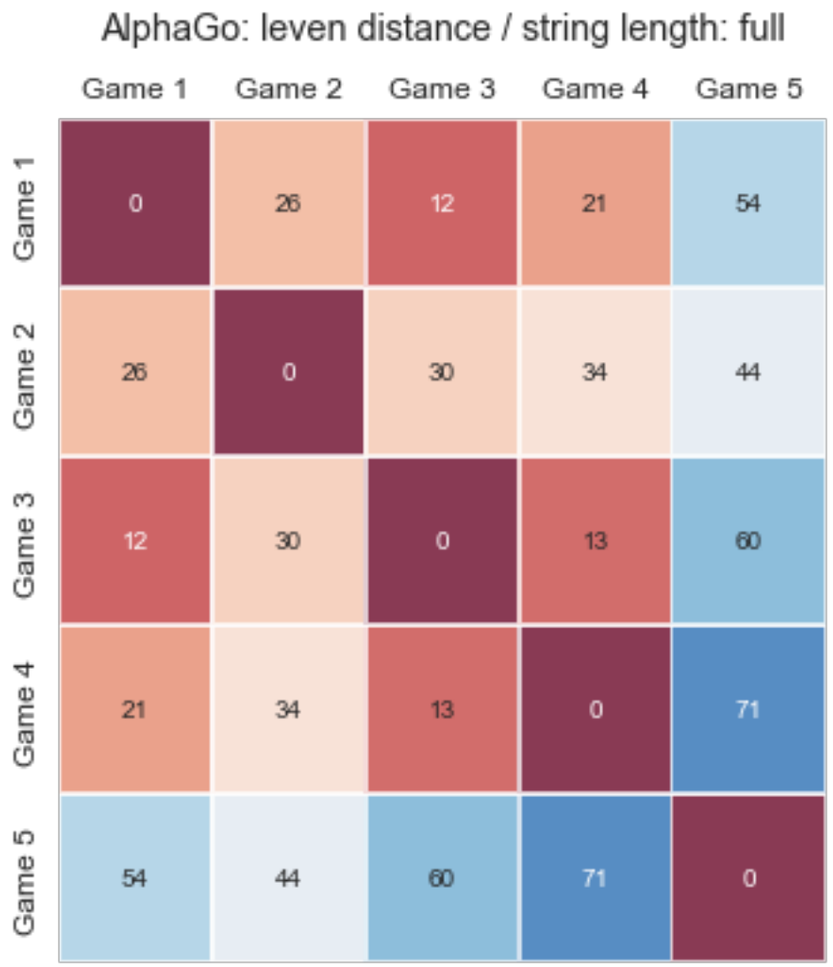
```
def levheatmap(alist, blist, length):
    al = []
    bl = []
    if length != 'full':

        for a in alist:
            a = a[:length]
            al.append(a)

        for b in blist:
            b = b[:length]
            bl.append(b)
    else:
        al = alist
        bl = blist

    array_A = np.zeros([len(al),len(al)])
    for i in range(0, len(al)):
        for j in range(0, len(al)):
            array_A[i][j] = levenshtein(al[i], al[j])

    array_B = np.zeros([len(bl),len(bl)])
    for i in range(0, len(bl)):
        for j in range(0, len(bl)):
            array_B[i][j] = levenshtein(bl[i], bl[j])

    maxval = max(array_A.max(), array_B.max())

    f, (ax1, ax2) = plt.subplots(1, 2, figsize=(12,6))
    ax1_title = 'AlphaGo: leven distance / string length: {}'.format(length)
    ax2_title = 'Lee Sedol: leven distance / string length: {}'.format(length)

    sns.heatmap(array_A, alpha=0.75, vmax=maxval, cmap='RdBu', linewidths=.5, cbar=False, annot=True, xticklabels=['Game

    sns.heatmap(array_B, alpha=0.75, vmax=maxval, cmap='RdBu', linewidths=.5, cbar=False, annot=True, xticklabels=['Game
    ax1.set_title(ax1_title, y=1.08)
    ax2.set_title(ax2_title, y=1.08)

    ax1.xaxis.tick_top()
    ax2.xaxis.tick_top()

    plt.show()
```

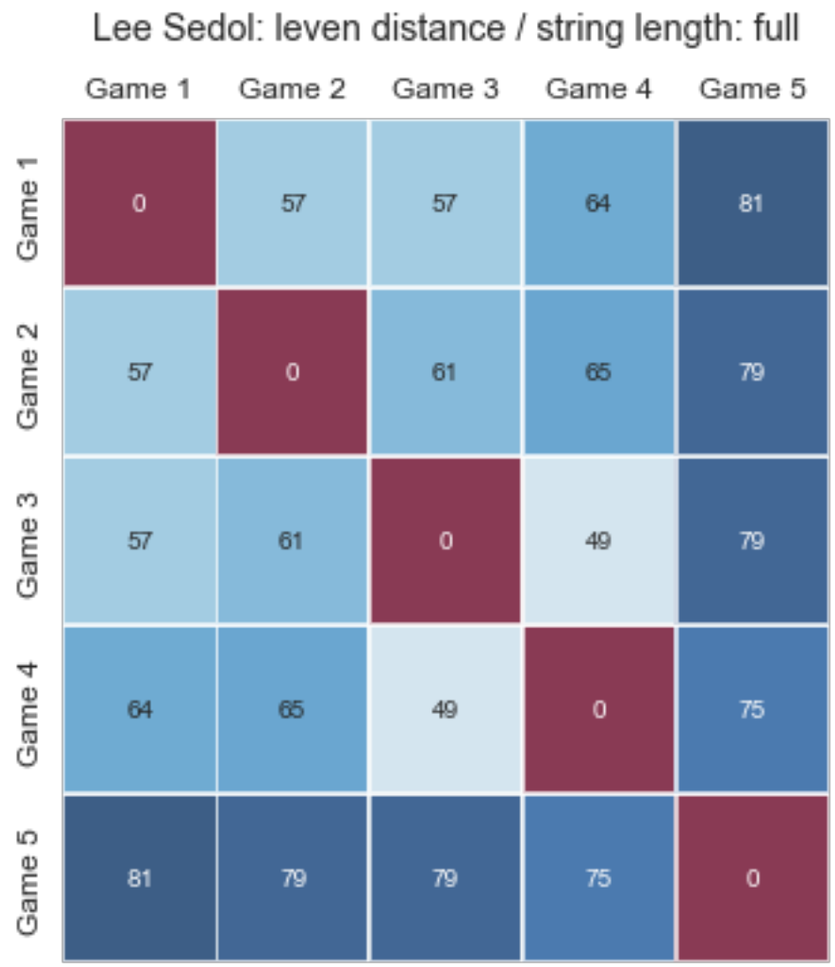
문자열을 길이에 맞게 조정
(FULL OR FIRST 10, 30, ETC)

동일 플레이어의 대국간
레벤슈타인 거리 산출

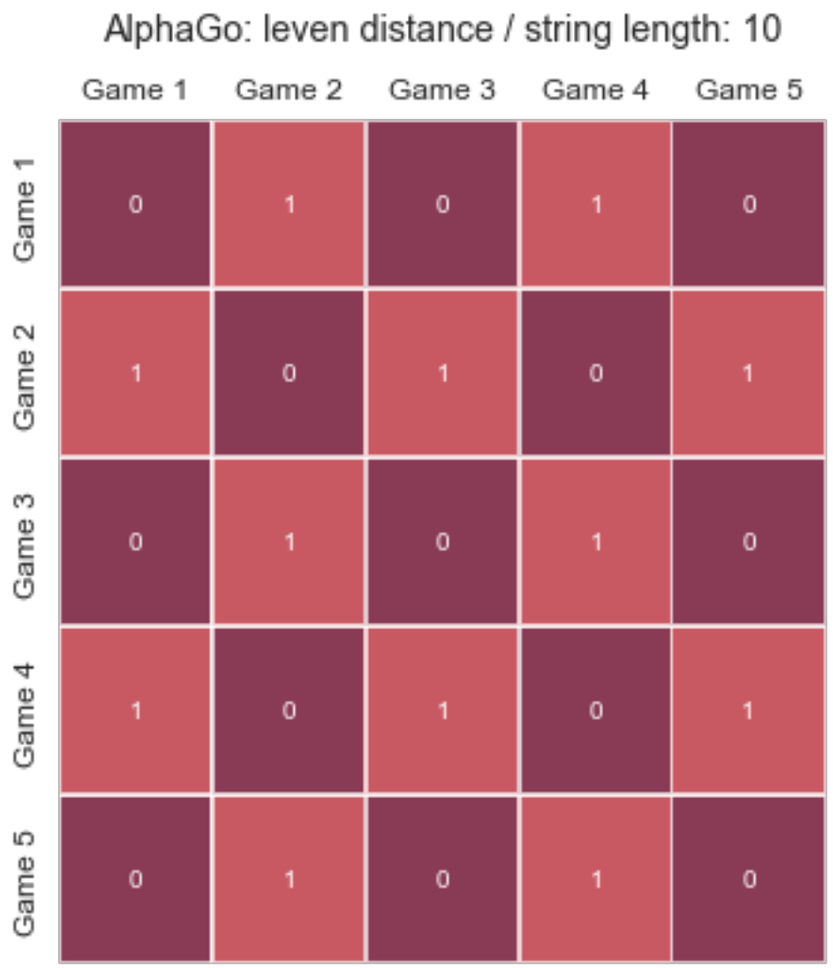
히트맵 시각화

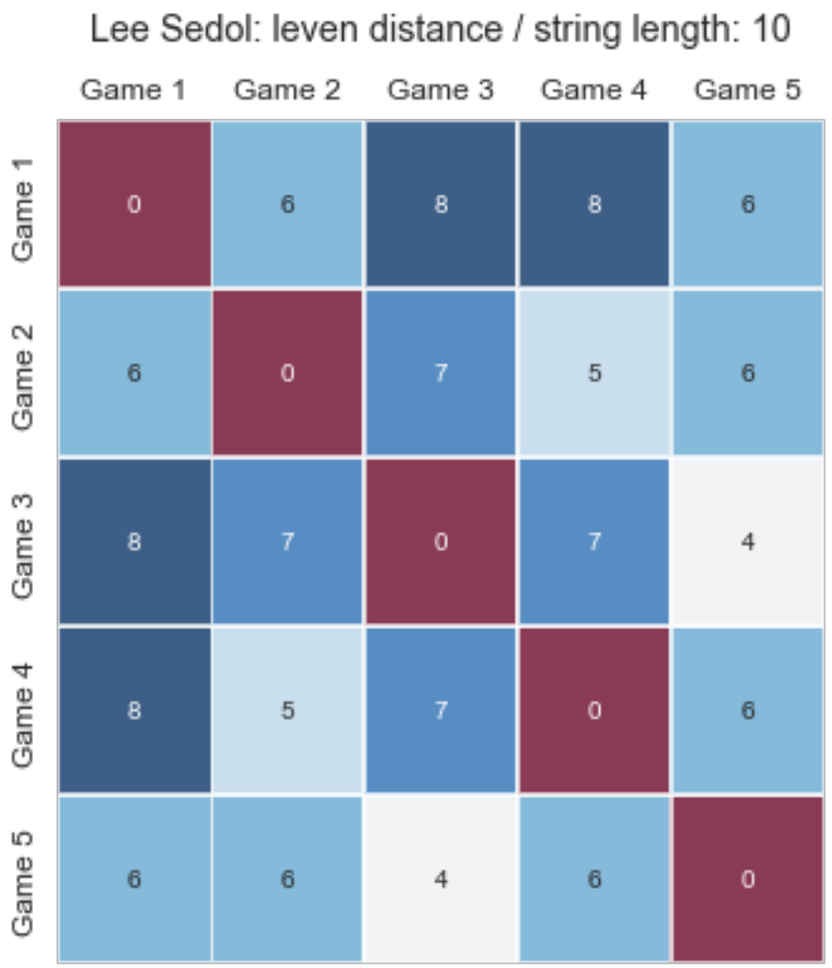
MATCH STRING: FULL





MATCH STRING: FIRST 10





FULL

AlphaGo: leven distance / string length: full

	Game 1	Game 2	Game 3	Game 4	Game 5
Game 1	0	26	12	21	54
Game 2	26	0	30	34	44
Game 3	12	30	0	13	60
Game 4	21	34	13	0	71
Game 5	54	44	60	71	0

Lee Sedol: leven distance / string length: full

	Game 1	Game 2	Game 3	Game 4	Game 5
Game 1	0	57	57	64	81
Game 2	57	0	61	65	79
Game 3	57	61	0	49	79
Game 4	64	65	49	0	75
Game 5	81	79	79	75	0

FIRST 10

AlphaGo: leven distance / string length: 10

	Game 1	Game 2	Game 3	Game 4	Game 5
Game 1	0	1	0	1	0
Game 2	1	0	1	0	1
Game 3	0	1	0	1	0
Game 4	1	0	1	0	1
Game 5	0	1	0	1	0

Lee Sedol: leven distance / string length: 10

	Game 1	Game 2	Game 3	Game 4	Game 5
Game 1	0	6	8	8	6
Game 2	6	0	7	5	6
Game 3	8	7	0	7	4
Game 4	8	5	7	0	6
Game 5	6	6	4	6	0

FIRST 30

AlphaGo: leven distance / string length: 30

	Game 1	Game 2	Game 3	Game 4	Game 5
Game 1	0	1	0	2	1
Game 2	1	0	1	1	2
Game 3	0	1	0	2	1
Game 4	2	1	2	0	3
Game 5	1	2	1	3	0

Lee Sedol: leven distance / string length: 30

	Game 1	Game 2	Game 3	Game 4	Game 5
Game 1	0	17	23	21	17
Game 2	17	0	22	19	20
Game 3	23	21	0	19	19
Game 4	20	19	19	0	19
Game 5	17	20	19	18	0

FIRST 90

AlphaGo: leven distance / string length: 90

	Game 1	Game 2	Game 3	Game 4	Game 5
Game 1	0	13	9	18	8
Game 2	13	0	14	18	15
Game 3	9	14	0	13	10
Game 4	18	18	13	0	21
Game 5	8	15	10	21	0

Lee Sedol: leven distance / string length: 90

	Game 1	Game 2	Game 3	Game 4	Game 5
Game 1	0	54	54	62	59
Game 2	54	0	52	53	58
Game 3	54	52	0	49	52
Game 4	62	53	49	0	49
Game 5	59	58	52	48	0

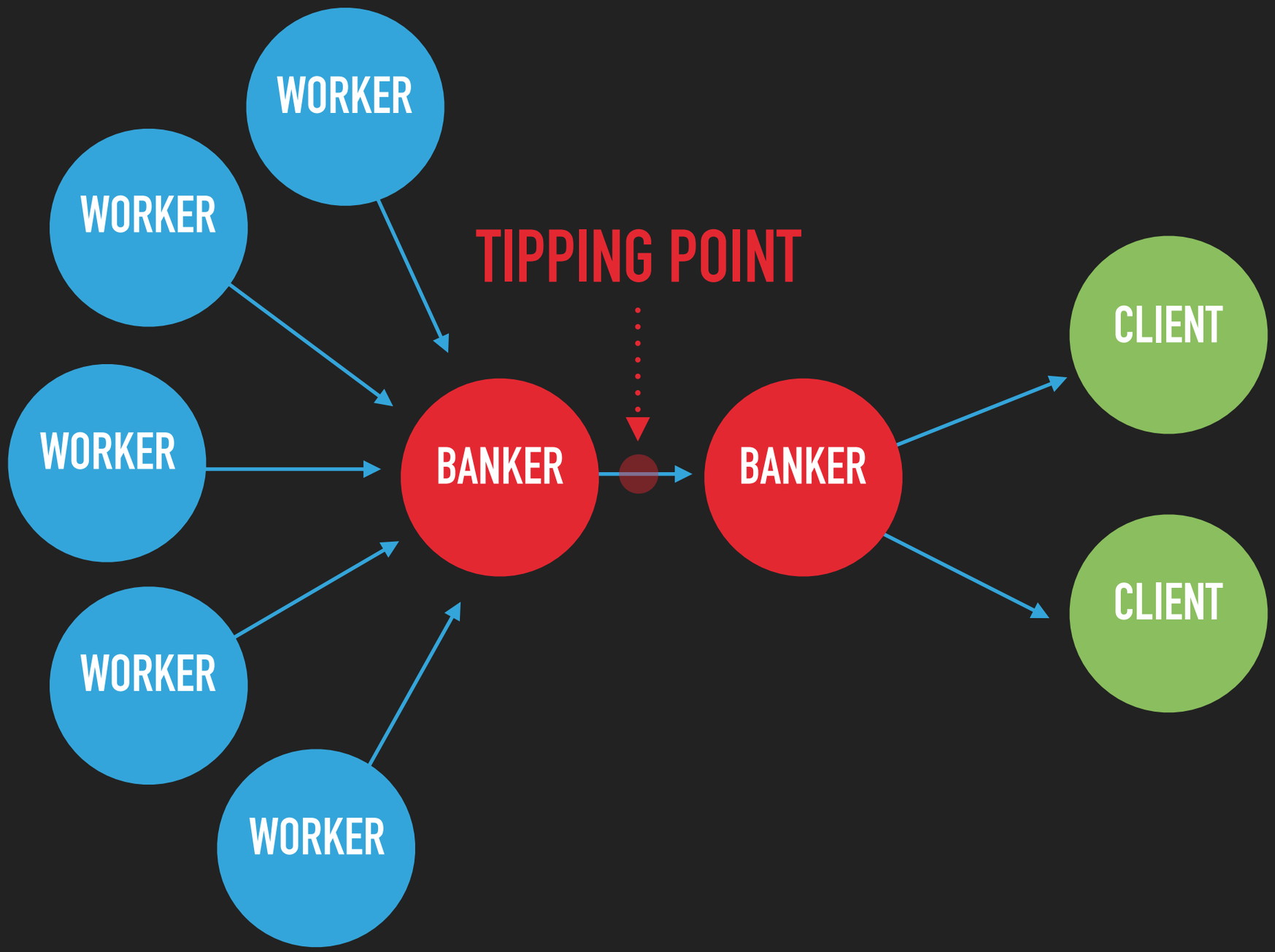
ALPHAGO @ NEXON

TO CREATE VS. TO DESTROY

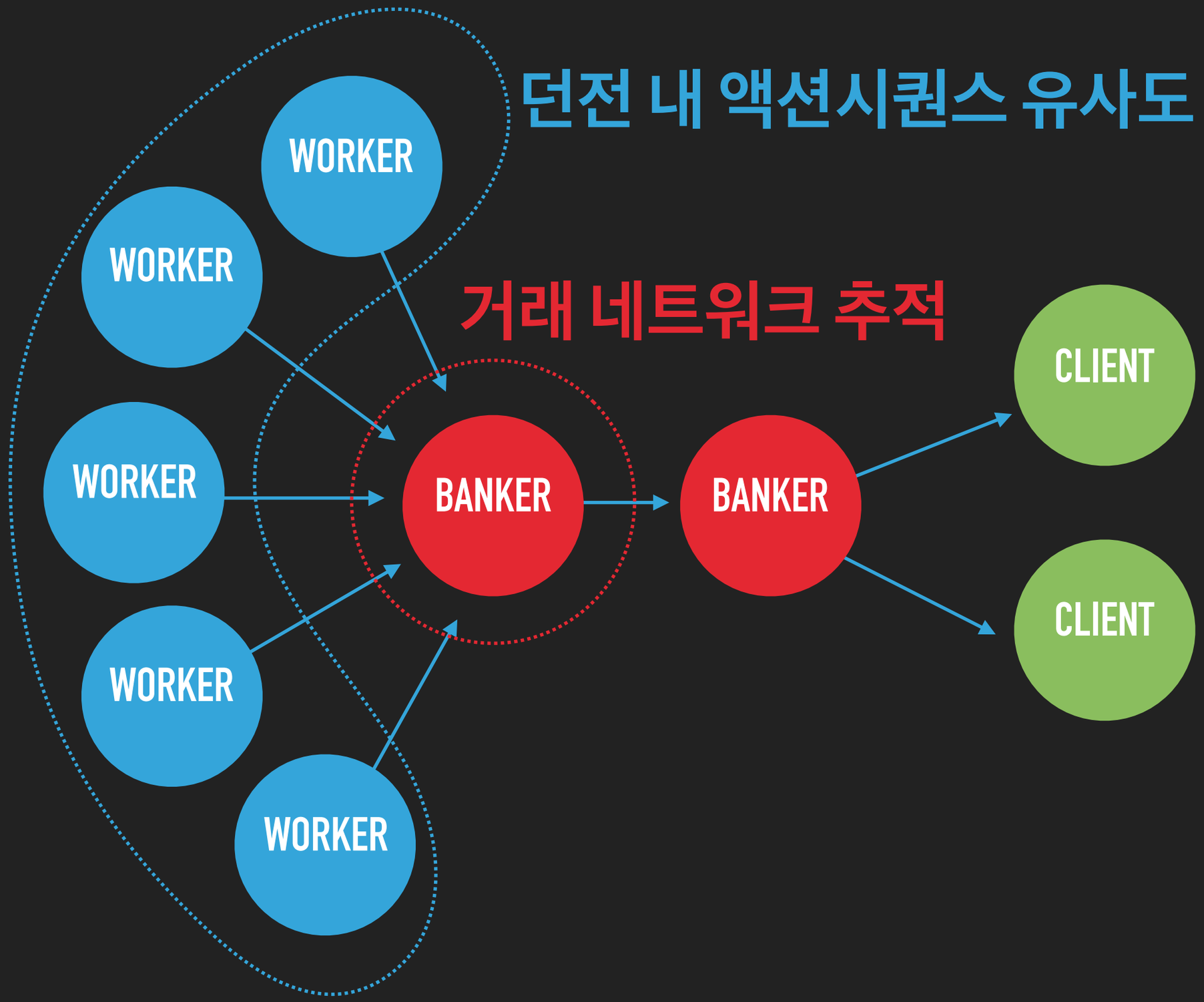
source: https://www.youtube.com/watch?v=K8k3NQctcNY

무리지어 다니는 작업장 매크로

작업장의 일반적인 조직도



작업장 검거 프로세스 예시



~~알파고를 검겨할~~

동료를 찾습니다

분석 / 개발

머신러닝

대용량 데이터 처리

마치며

source: http://bluehillresearch.com/what-data-analysts-actually-do/

DATA ANALYST OVERWHELMED

- 데이터분석 -

#실질객관 #미래예측
#회귀분석 #통계검정

source: http://www.shadowmatic.com/presskit/

SHADOWMATIC

감사합니다!

http://jsideas.net/python/2016/03/16/Google_DeepMind.html

http://jsideas.net/python/2016/07/07/levenshtein.html

https://github.com/junkwhinger/AlphaGo_raw