

## Part 1) Visualization (50 points; 10%)

### Task 1. Spatial Analysis and Visualization with Python

โปรแกรม taxi\_app.py ที่ให้ มีคุณลักษณะดังต่อไปนี้

- ใช้ streamlit เพื่อสร้าง web app

\*\*\*สามารถรัน taxi\_app.py โดยการเข้าไปที่ command prompt (cmd) แล้วรันคำสั่ง

1. run “docker exec -it ds2023 bash”
  2. cd เข้าไปใน folder ที่มี taxi\_app.py
  3. run “streamlit run taxi\_app.py --server.port=8501 --server.address=0.0.0.0”
  4. เข้าผ่าน <http://127.0.0.1:8501/> หรือ shortcut\_url/Streamlit\_url.url ที่เตรียมไว้ให้
- มีการนำเข้าข้อมูลจากไฟล์ taxi\_od.csv ซึ่งเป็นข้อมูลการเดินทางด้วยรถแท็กซี่จำนวนหนึ่งภายในหนึ่งวัน  
ข้อมูลแต่ละแถวคือบันทึกเหตุการณ์การรับหรือส่งผู้โดยสาร ประกอบด้วยคอลัมน์ดังต่อไปนี้

ชื่อคอลัมน์	คำอธิบาย
taxi_ID	หมายเลขของแท็กซี่
trip_ID	หมายเลข trip แท็กซี่หนึ่งคันมีได้หลาย trip
time	timestamp อยู่ในรูปแบบ yyyy-mm-dd hh:mm:ss เช่น 2016-09-23 06:03:46
hour	timestamp เฉพาะส่วนชั่วโมง
status	O คือ origin หรือ pick-up แสดงว่าเป็นการรับผู้โดยสาร D คือ destination หรือ drop-off แสดงว่าเป็นการส่งผู้โดยสาร
latitude	พิกัดตำแหน่งละติจูดแบบทศนิยม
longitude	พิกัดตำแหน่งลองจิจูดแบบทศนิยม
distance_meter	ระยะทางของทริป หน่วยเป็นเมตร
duration_min	เวลาที่ใช้ในทริป หน่วยเป็นนาที

taxi_ID	trip_ID	time	hour	status	latitude	longitude	distance_meter	duration_min
AK2306	AK2306_T_6203	9/23/2016 6:03	6	O	13.73136425	100.650322	5	11
AK2306	AK2306_T_6203	9/23/2016 6:14	6	D	13.73135662	100.6502838	5	11
AK2306	AK2306_T_6204	9/23/2016 6:20	6	O	13.73130798	100.6501999	7456	26
AK2306	AK2306_T_6204	9/23/2016 6:46	6	D	13.73575211	100.6769485	7456	26
AK2306	AK2306_T_6206	9/23/2016 6:50	6	O	13.73414803	100.6917877	7163	36
AK2306	AK2306_T_6206	9/23/2016 7:26	7	D	13.75986385	100.6820679	7163	36

ตัวอย่างข้อมูล

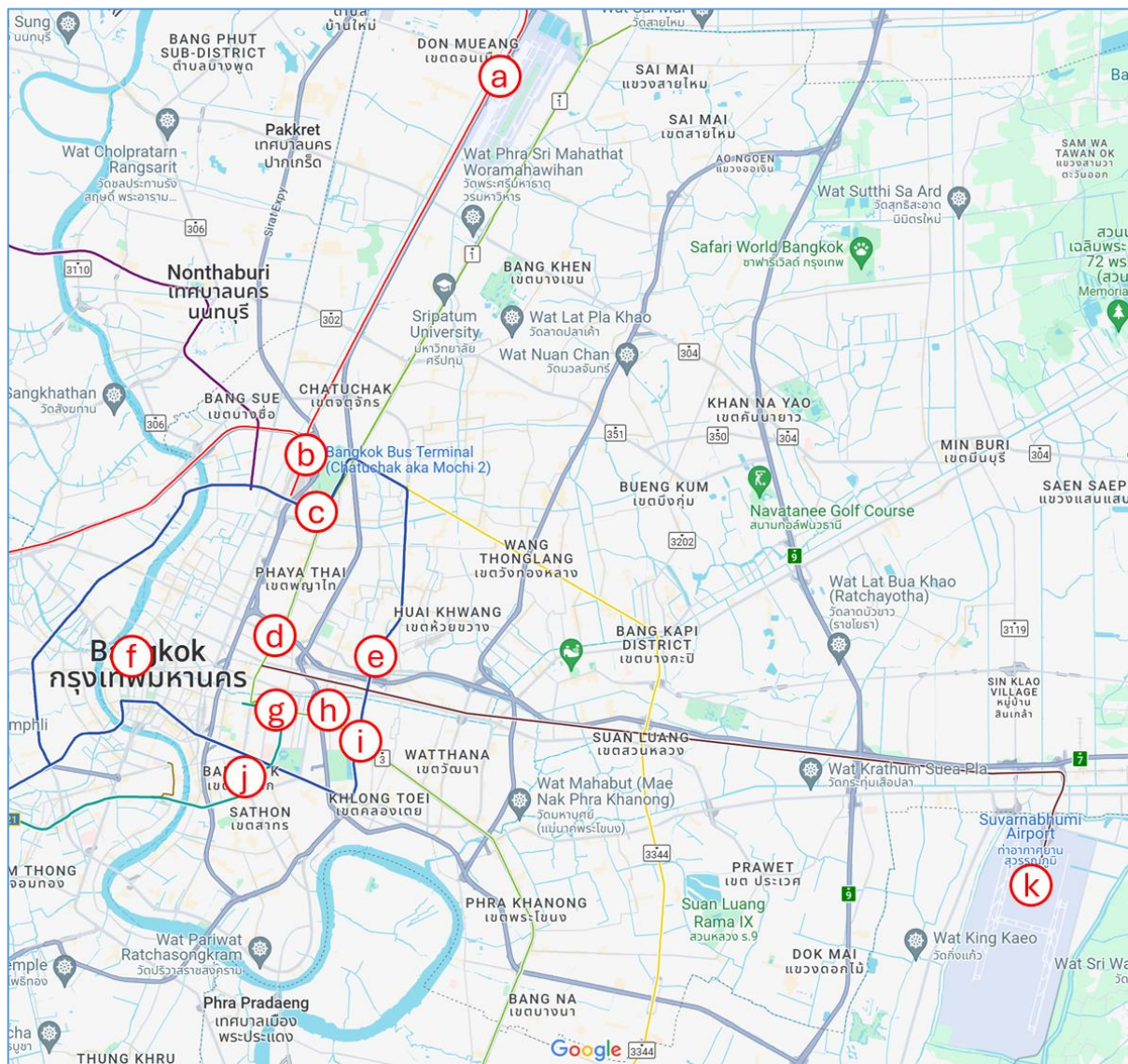
- มีการทำ clustering ของจุดรับส่งผู้โดยสาร ด้วยอัลกอริทึม DBScan เพื่อหาว่ามีการรับส่งผู้โดยสารในพื้นที่บริเวณไหนมากที่สุด โดยพิจารณาจากขนาดของคลัสเตอร์
- UI ของโปรแกรม แสดงจำนวนการรับส่งผู้โดยสารรายชั่วโมง, ขนาดของคลัสเตอร์เรียงจากมากไปน้อย, และแผนที่ basemap ที่ยังไม่ได้นำข้อมูลมาแสดง
- UI ของโปรแกรม ให้ผู้ใช้สามารถเลือกข้อมูลตามช่วงเวลา, status O/D, และจำนวนคลัสเตอร์ที่ต้องการแสดงผล

สมมติว่าனிสิตต้องการช่วยคนขับรถแท็กซี่วางแผนการรับผู้โดยสาร โดยใช้โปรแกรมนี้นี้ระบุพื้นที่ที่มีโอกาสสูงที่จะได้รับผู้โดยสาร ให้นิสิตปรับปรุงโปรแกรมเพื่อนำข้อมูลตำแหน่งรับส่งผู้โดยสาร ที่เป็นผลลัพธ์จากการทำ clustering มาแสดงบนแผนที่ ในรูปแบบ scatter map และ/หรือ heatmap และตอบคำถามต่อไปนี้ พร้อมกับ capture หน้าจอที่แสดงภาพของ visualization ที่นำมาสู่คำตอบแต่ละข้อ

ให้ใช้ตัวเลือกต่อไปนี้ในการตอบคำถามข้อ 2, 3, 5

- |                                       |                              |
|---------------------------------------|------------------------------|
| a) สนามบินดอนเมือง                    | f) ถนนข้าวสาร-บางลำพู        |
| b) สถานีขนส่งผู้โดยสารกรุงเทพ จตุจักร | g) เซ็นทรัลเวิลด์-ราชประสงค์ |
| หรือ สถานีหมอชิต 2                    | h) ถนนสุขุมวิทซอย3 (นานา)    |
| c) ตลาดนัดจตุจักร                     | i) เทอร์มินัล21-แยกอโศกมนตรี |
| d) อนุสาวรีย์ชัยสมรภูมิ               | j) สามย่าน-สีลม              |
| e) เซ็นทรัลพระราม9-รัชดาภิเษก         | k) สนามบินสุวรรณภูมิ         |

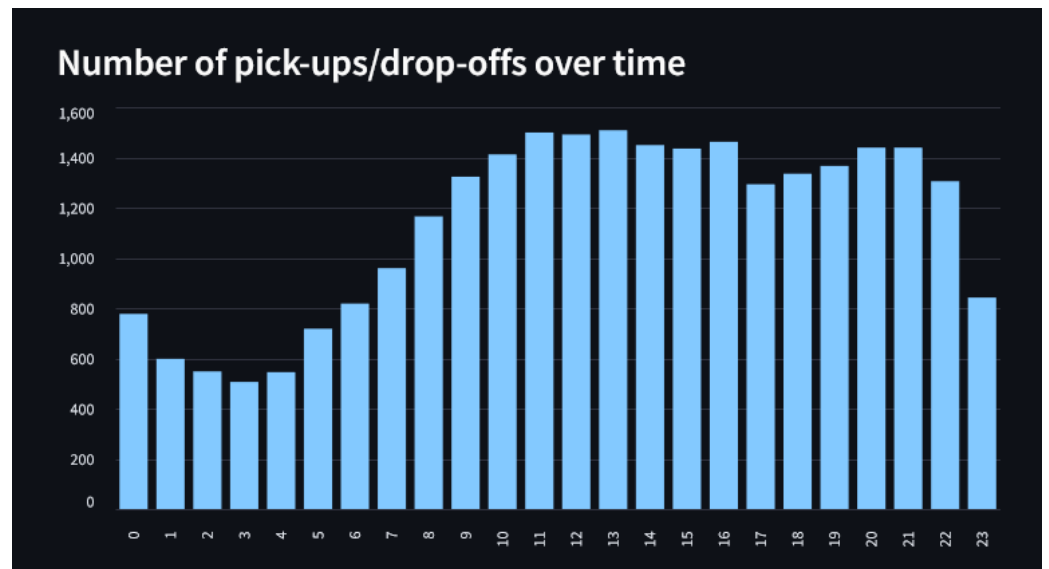
ภาพข้างล่างแสดงตำแหน่งโดยประมาณของตัวเลือก



- 1) ช่วงเวลาชั่วโมงใดของวัน ที่แท็กซี่ควรพักผ่อน เนื่องจากมีการรับผู้โดยสารน้อยที่สุด (ตอบเป็นตัวเลข 0 ถึง 23)

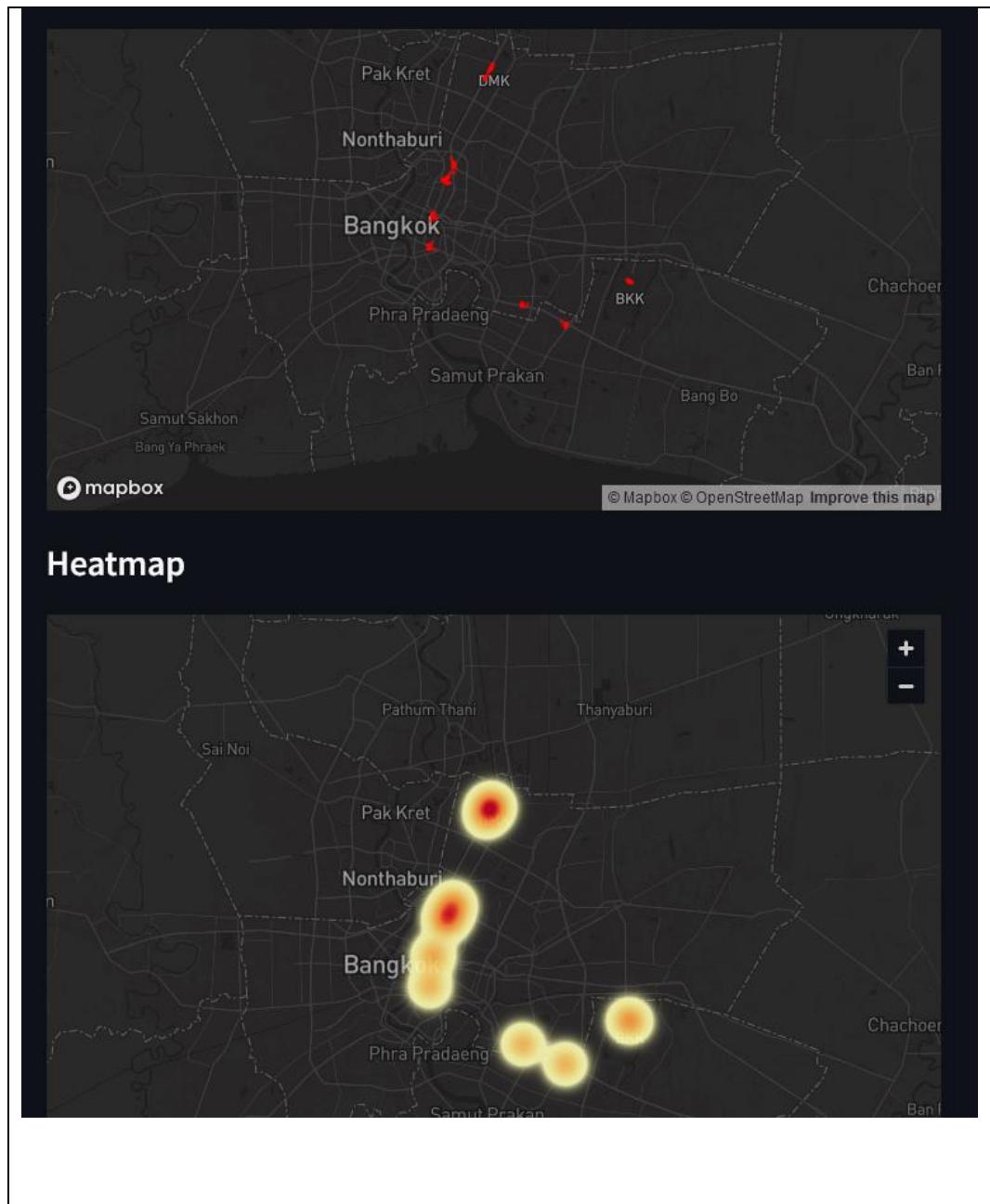
ตอบ 2

(ภาพประกอบ)



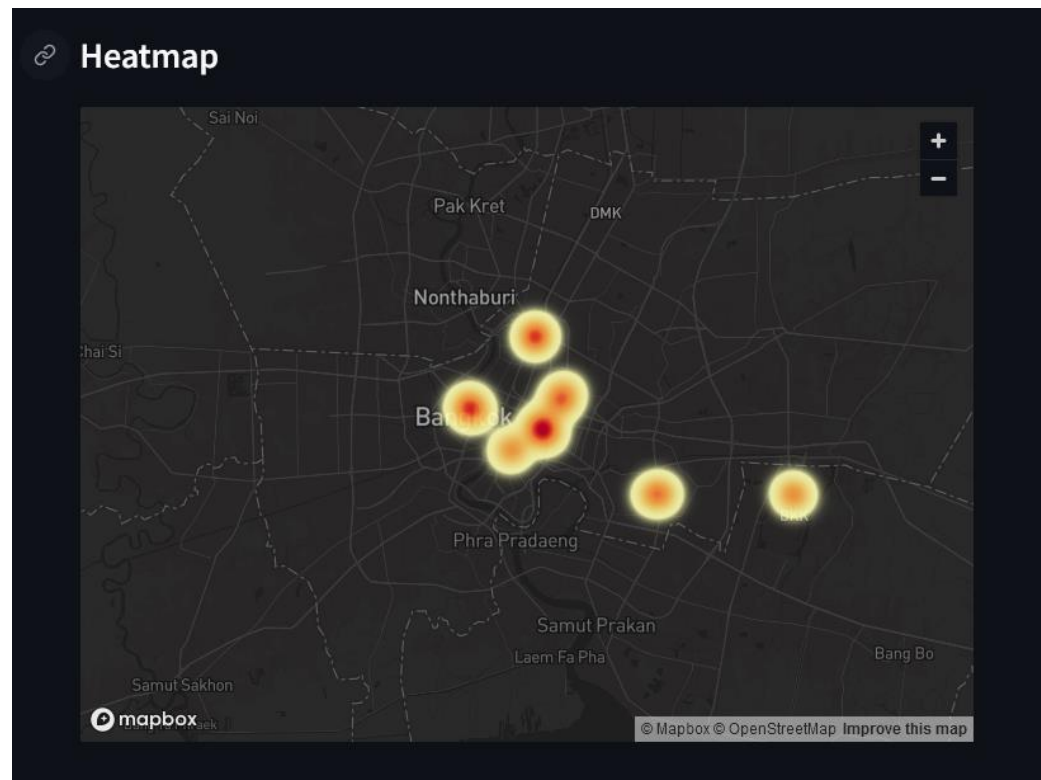
- 2) ช่วงเวลาที่ Hour=12 (12:00-12:59 น.) แท็กซี่ควรไปรับผู้โดยสารที่บริเวณใดมากที่สุด  
(ให้พิจารณาจากข้อมูลผู้โดยสารบนแผนที่ แล้วเลือกตัวเลือกที่อยู่ใกล้ศูนย์กลางของคลัสเตอร์มากที่สุด)  
ตอบ สนามบินดอนเมือง

(ภาพประกอบ)



- 3) ช่วงเวลาที่ Hour=21 (21:00-21:59 น.) แท็กชี้ควรไปรับผู้โดยสารที่บริเวณใดมากที่สุด  
 (ให้พิจารณาจากข้อมูลจุดรับผู้โดยสารบนแผนที่ แล้วเลือกตัวเลือกที่อยู่ใกล้ศูนย์กลางของคลัสเตอร์มากที่สุด)  
ตอบ ถนนสุขุมวิทซอย3 (นานา)

(ภาพประกอบ)

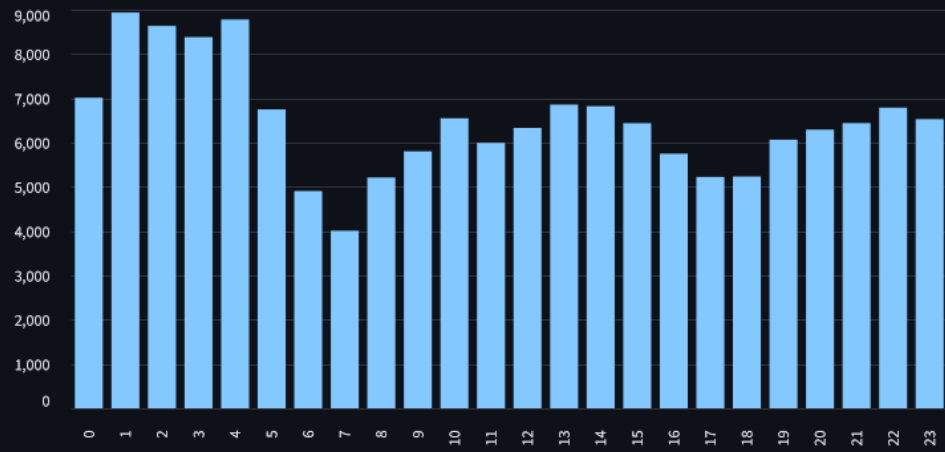


- 4) สำหรับแท็กซีที่ขอบว้างยาวๆ เพื่อจะได้ค่าโดยสารเที่ยวละมากๆ จึงต้องการรับผู้โดยสารแล้ว มี average distance สูงสุด ควรจะรับผู้โดยสารในชั่วโมงใด (ตอบเป็นตัวเลข 0 ถึง 23)

ตอบ 1

(ภาพประกอบ)

**Average meter of pick-ups/drop-off over time**





5) จากคำตอบในข้อ 4) ปลายทางที่นิยมมากที่สุดในช่วงโมงนั้น คือที่ไหน

ตอบ ถนนสุขุมวิทซอย3 (นานา)

(ภาพประกอบ)




## Task 2. Network Analysis and Visualization with Gephi

ให้นักศึกษาใช้โปรแกรม Gephi เปิดไฟล์ hero-social-network.gephi ซึ่งเป็นข้อมูลเครือข่ายความสัมพันธ์ของละครในนิตยสารภาพแนวซูเปอร์ฮีโร่ โดยแต่ละ node เป็นตัวละครแต่ละตัว และ edge แสดงความสัมพันธ์ระหว่างตัวละคร ให้นักศึกษาใช้เครื่องมือวิเคราะห์ในโปรแกรม Gephi และสร้าง network visualization เพื่อตอบคำถามข้างล่างนี้ โดยให้ capture หน้าจอที่แสดงผลลัพธ์ที่เกี่ยวข้องมาแปะก็ภาพก็ได้ พร้อมทั้งเขียนคำอธิบายประกอบภาพ

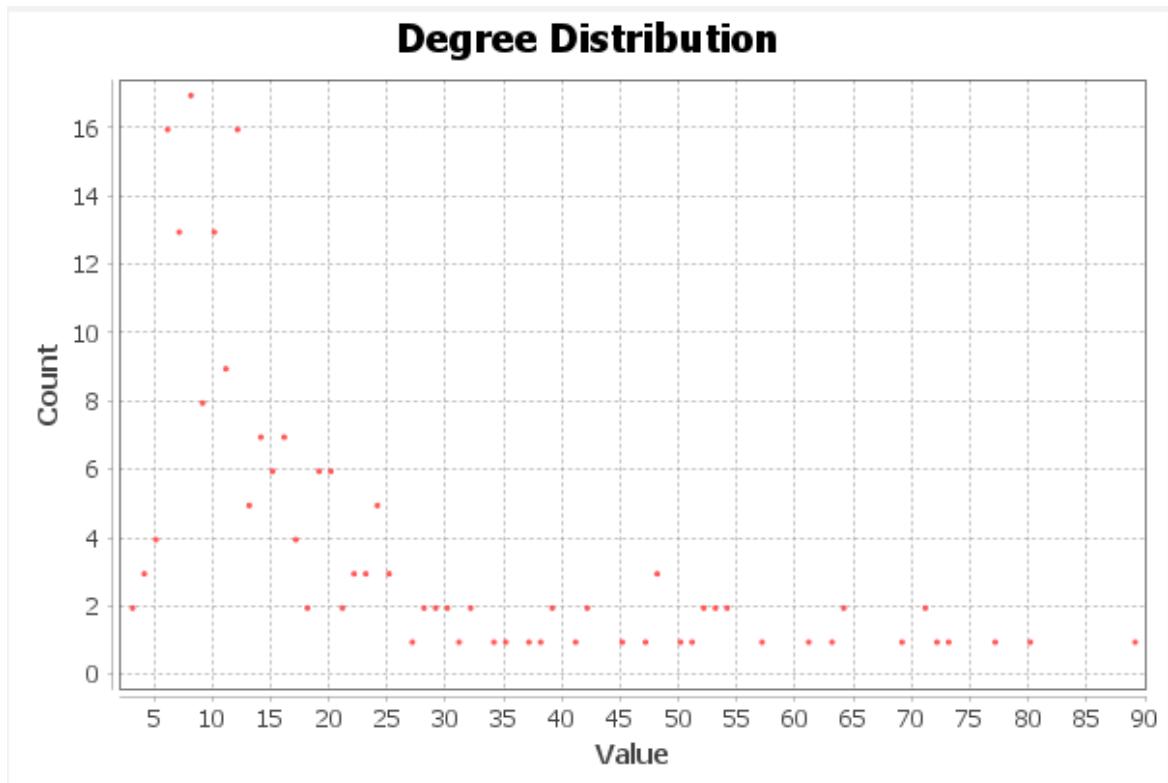
### Note

- ควรรันโปรแกรม Gephi ก่อน แล้วค่อยใช้เมนู File --> Open เพื่อเปิดไฟล์ hero-social-network.gephi แทนที่จะรันโปรแกรมด้วยการ double-click ที่ไฟล์ hero-social-network.gephi



- เมื่อเปิดไฟล์แล้ว หากไม่เห็นภาพเน็ตเวิร์ค ให้กดปุ่ม  (Center on Graph)
- 6) เมื่อพิจารณาจาก degree distribution เน็ตเวิร์คนี้มีคุณลักษณะใกล้เคียงกับเน็ตเวิร์คโมเดลประเภทใด
- a) random
  - b) small-world
  - c) scale-free

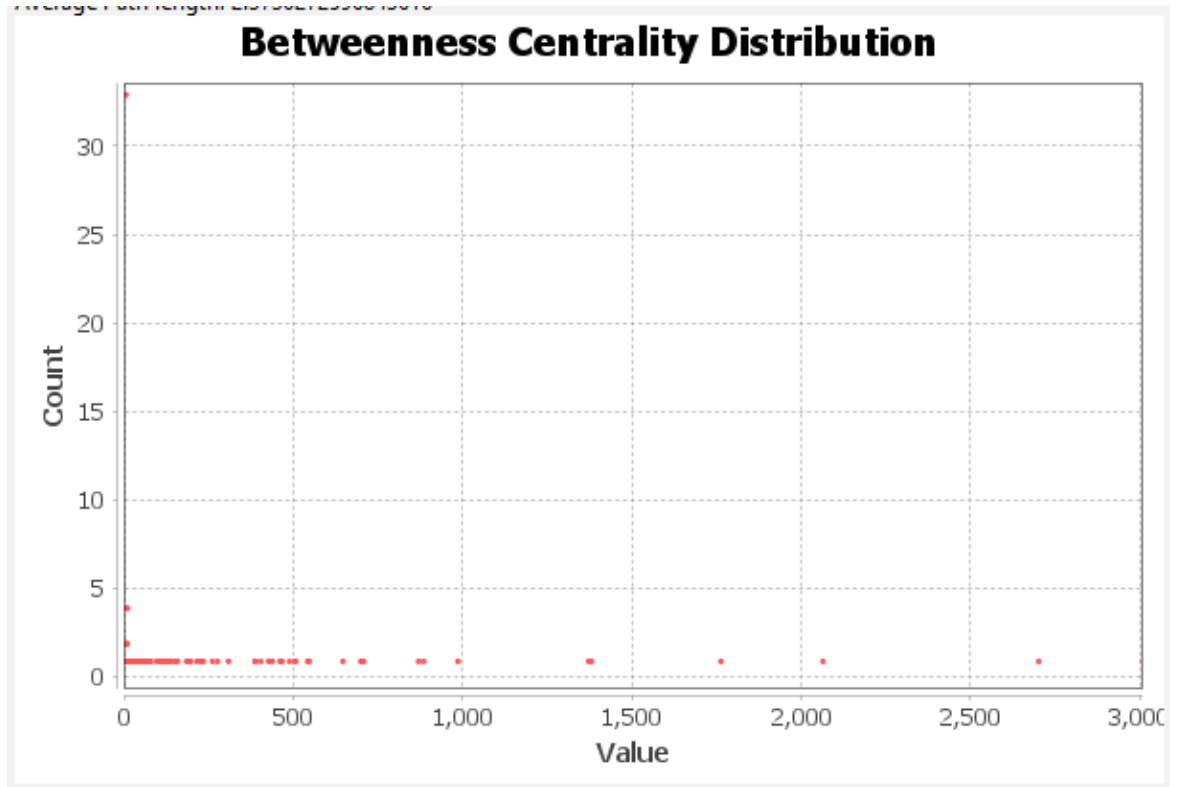
ภาพ degree distribution



- 7) ถ้าต้องการหาว่าใครคือตัวละครที่มีความสำคัญที่สุดโดยพิจารณาจาก centrality measure ที่ให้ความสำคัญกับโหนดที่เป็นตัวเชื่อมโยงระหว่างกลุ่มต่างๆ ในเครือข่าย ควรใช้ค่า centrality ตัวไหน
- a) degree centrality
  - b) closeness centrality
  - c) betweenness centrality
  - d) pagerank centrality
- 8) ด้วยวิธีการที่เป็นคำตอบในข้อ 7) ตัวละครตัวใดมีความสำคัญที่สุด 2 อันดับแรก

- a) Spider-man, Captain America
- b) Captain America, Thor
- c) Spider-man, Thor
- d) Captain America, Beast

กราฟ distribution ของ centrality ที่เลือกใช้



ตารางในแท็บ Data Laboratory ที่เรียงลำดับตามค่า centrality ตัวที่เลือกใช้จากมากไปน้อย

Label	Betweenness Centrality ▾
SPIDER-MAN/PETER PAR	3003.570054
THOR/DR. DONALD BLAK	2695.966696
CAPTAIN AMERICA	2058.466299

9) เมื่อรัน Modularity detection แล้วได้จำนวน class เท่าไหร่

- a) 3 class
- b) 4 class
- c) 5 class
- d) 6 class
- e) 7 class

10) ตัวละครที่เป็นคำตอบในข้อ 8) มี modularity class เดียวกันหรือต่างกัน

a) Class เดียวกัน

b) คนละ class

ภาพ network ที่แสดง modularity แต่ละ class ด้วยสีต่างกัน

