

# Social and Economic Networks: exercise 2

*Katrina Walker, Frank Fanteev, David Rosenfeld*

a)

This exercise was done in a separate document because it used igraph in R and we were not able to include it in the Jupyter notebook alongside the rest.

Here we import a dataset from SNAP about the California road network. Here are a couple of basic statistics about the dataset:

```
## [1] "Number of nodes: 1971281"
```

```
## [1] "Number of edges: 5533214"
```

We then compute degree centrality, eigenvector centrality, PageRank centrality and HITS centrality for each node, and save the top 100 most central nodes in each case:

We then look at which nodes appear in all of the top 100 lists, or at least in 2 distinct sets of top 100 list:

```
## [1] "Nodes in the top 100 most central in all centrality measures:  "
```

```
## [1] "Nodes in the top 100 most central in degree centrality and eigenvector centrality:  "
```

```
## [1] "Nodes in the top 100 most central in degree centrality and PageRank centrality:  1965207"
```

```
## [1] "Nodes in the top 100 most central in degree centrality and HITS centrality:  "
```

```
## [1] "Nodes in the top 100 most central in eigenvector centrality and PageRank centrality:  "
```

```
## [1] "Nodes in the top 100 most central in eigenvector centrality and HITS centrality:  "
```

```
## [1] "Nodes in the top 100 most central in PageRank centrality and HITS centrality:  "
```

In conclusion, there is only 1 node (node #1965207) in common among all the 100 most central nodes using different centrality measures. This node appears both in the most central nodes in terms of degree centrality and PageRank centrality. In short, the different centrality measures produce very different rankings of nodes in terms of centrality.

b)

We now consider some of the macro properties of the graph by calculating the diameter and average distance. Unfortunately, the graph was so large that the calculations took so long, so for the purpose of demonstrating how to compute these macro properties, we use a random subsample of the graph roughly half the size:

```
## [1] "Diameter of the network:  272"
```

```
## [1] "Average distance of the network:  35.2827347588126"
```

```
## [1] "Number of triangles in the network:  183456"
```

```
## [1] "Average clustering coefficient:  0.0482738803825787"
```