Τέταρτη Σειρά Ασκήσεων – Εξόρυξη Δεδομένων

Όνομα: Χρύσα Τεριζή
ΑΜ: 2553
Ημερομηνία: 27/6/2017
Free passes: Χρησιμοποίησα 4

Ερώτηση 1

Δεν την έχω υλοποιήσει.

Ερώτηση 2(Κατηγοριοποίηση)

Αρχικά κάνω import κάποιες βιβλιοθήκες. Έχω χρησιμοποιήσει και εδώ την συνάρτηση "**def getTagsHandles(x)**" που βρίσκει τα hashtags/handles που υπάρχουν σε κάποιο κείμενο και το επιστρέφει σαν λίστα. Ξεκινάω να βρω τους χρήστες που έχουν κάνει λιγότερα από 10 tweets και δεν ασχολούμαι με τα δικά τους δεδομένα. Ξεκινάω και διαβάζω το αρχείο "**clinton_trump_tweets.txt**", σε ένα λεξικό "**counterTweet**" κρατάω σαν κλειδί το id του χρήστη και σαν τιμή το πλήθος που το έχω συναντήσει καθώς διαβάζω το αρχείο με τα tweets. Σε μία λίστα "**finalNames**" αποθηκεύω τα ονόματα των χρηστών τους οποίους θέλουμε να χρησιμοποιήσουμε. Έπειτα, φτιάχνω ένα αρχείο "**dataTrain.txt**" το οποίο θα έχει όλα τα tweets των χρηστών που μας ενδιαφέρουν.

```
TIME find counter of tweet: 134.30304958634224 seconds
TIME find final user >10: 0.35527430338152044 seconds
TIME write dataFIle: 284.70442661244306 seconds
```

Βρήκα ότι υπάρχουν,

```
Total number of users: 70608
```

Στο λεξικό "**nameIDAndFollow**" φτιάχνω την αντιστοιχία για το ποιον ακολουθεί ο κάθε χρήστης. Τα δεδομένα που έχω από τα tweets είναι τα ακόλουθα,

```
Name : Cebel
ScreenName : Cebel6
UserID : 1519696717
FollowersCount : 132
FriendsCount : 263
Location : Little Rock, Arkansas
Description : Arkansas Razorback Fan Just trying to be #Uncommon one 1-0 day at a time.
CreatedAt : Sat Oct 29 08:10:06 EEST 2016
StatusID : 792232017094119425
Language : en
Place : null
RetweetCount : 0
FavoriteCount : 1
Text : @NWAJimmy I've read it now though brother. Was pretty spot on Lots of bright spots but a lot to work on. Exact
ly as an exhibition should be!
```

Μέσα στο σύνολο "**data**" αποθηκεύω οι καταχωρήσεις για κάθε χρήστη. Δοκίμασα διάφορα **features** να χρησιμοποιήσω. Χρησιμοποίησα το **location**, τον αριθμό **followersCount**, τον αριθμό **friendsCount**, το **description**, το **language** και το **text**. Βρίσκω ότι έχω συνολικά,

```
TIME: 170.07063598102542 seconds
Total numbe of kataxwrhseis sto data: 4760019
```

Τα δεδομένα αυτά του συνόλου τα έχω σε μία λίστα τώρα. Το **5 cross-validation** το έχω υλοποιήσει εγώ(δεν χρησιμοποίησα την συνάρτηση της python). Έχω φτιάξει 5 αρχεία για τα δεδομένα που πρέπει να κάνω train κάθε φορά και 5 αρχεία test με τα δεδομένα που πρέπει να κάνω τεστ. Έχω μία συνάρτηση "**def statistics(y_pred, y_test)**" όπου υπολογίζει το σκορ ανάμεσα στα πραγματικά labels και σε αυτά που μου επιστρέφουν οι κατηγοριοποιητές. Ξεκινάω τις 5 επαναλήψεις για το cross-validation και γεμίζω τις λίστες **Y_TRAIN = [] , X_TRAIN = []** με τα δεδομένα που θέλω. Την πληροφορία από τα hashtags/handles την έχω χρησιμοποιήσει με τον εξής τρόπο, βρίσκω από τα δεδομένα train το πλήθος που χρησιμοποιεί το κάθε hashtag/handle οι χρήστες του Trump και της Clinton. Βρίσκω ένα ποσοστό για το κάθε ένα και τώρα για τις καταχωρήσεις προσθέτω τις διαφορές των ποσοστών αυτών από τα hashtags/handles που χρησιμοποιεί και καταχωρώ αυτό το άθροισμα. Την ίδια επεξεργασία κάνω και για τα δεδομένα του **test** όταν αυτά τα διαβάζω από το αρχείο. Κατόπιν, εκτελώ τους αλγορίθμους της κατηγοριοποίησης. Όμως δεν έχω κάποια αποτελέσματα ποσοστού γιατί είχα ένα θέμα με την μνήμη του υπολογιστή μου και τον χρόνο αναμονής για να ολοκληρωθούν και οι 5 επαναλήψεις. Βρήκα ότι για το 1ο cross-validation το **decision tree** είχε επιτυχία **85%**. Είχα βάλει να τρέξω τον **SVM** ολόκληρο το βράδυ και κόλλησε οπότε δεν δοκίμασα να το ξανά τρέξω. Λόγω μνήμης δεν έτρεξα τους υπόλοιπους κατηγοριοποιητές και τις υπόλοιπες επαναλήψεις οπότε δεν έχω κάποιο αποτέλεσμα ποσοστού για αυτά. Επίσης, είχα θέμα ότι όλα αυτά γινότανε για λίγο στην μνήμη όπου είχα γρήγορα αποτελέσματα αλλά μετά έτρεχαν στον δίσκο και δεν τελείωναν ποτέ. Δεν ξέρω για ποιον λόγο γινότανε αλλά πήγαιναν κατευθείαν στον δίσκο να τρέξουν.

Kaggle

Κάνω την ίδια επεξεργασία για τα δεδομένα του kaggle που μας δώσατε. Και έπειτα έφτιαξα το **CSV** αρχείο. Στο kaggle είχα την τελευταία θέση, **4η και με ποσοστό 57%**.

Ερώτηση 3 (Ανάλυση Δικτύων)

Αρχικά, ανοίγω το αρχείο "**clinton_trump_tweets.txt**" για να το διαβάσω γραμμή-γραμμή, κρατάω μέσα σε κάποια σύνολα κάποιες πληροφορίες για τους χρήστες και για τις ακμές, πιο συγκεκριμένα το σύνολο "**retweetUsers**" κρατάει τα ονόματα όλων των χρηστών που κάποιος τους έχει κάνει κάποιο retweet, το σύνολο "**allUsers**" κρατάει όλα τα ονόματα των χρηστών που έχουν κάνει κάποιο tweet/retweet και το σύνολο "**edgesList**" κρατάει τα ζευγάρια στα οποία ανάμεσα θα πρέπει να υπάρχει ακμή.

```
Total users: 200938
Total retweet users: 435482
Total edges: 1898377
```

Οι πρώτες 5 πληροφορίες που έχουν τα παραπάνω σύνολα είναι οι ακόλουθες,

Names of users

['rodneyvullapah', 'SPiotter', 'kadencarter13', 'Dylan_Ville', 'maddyhilliard']

Names of users that someone have retweet something of them
['TransparencyMV', 'angellicabell', 'bbcmtd', 'Miner168', 'deborah_bivens']
---
some of edges
['mbassuk-squartney', 'JBlancarteNBA-TimBontemps', 'rabomed277-commonsenseng1', 'tonygin2000-OccupyBawlStree', 'MikeMq1-WDFx2EU7']

Στην συνέχεια φτιάχνω το γράφημα G, αρχικά κάνω import την βιβλιοθήκη που φτιάχνει το γράφημα την "**networkx**", φτιάχνω την λίστα με tuples "**edgesListWithTuples**" όλες τις ακμές ώστε να τις εισάγω στο γράφημα με την εντολή **G.add_edges_from()**. Κάποιες από τις ακμές που έχουν δημιουργηθεί και κάποιοι κόμβοι είναι οι παρακάτω,

Ακμές	Κόμβοι
[('mbassuk', 'squartney'), ('mbassuk', 'MarloMeekins'), ('mbassuk', 'jonnysun'), ('mbassuk', 'alplicable'), ('mbassuk', 'Lin_Manuel')]	['mbassuk', 'squartney', 'JBlancarteNBA', 'TimBontemps', 'rabomed277']

Έπειτα, αφαιρώ τους κόμβους που δεν είναι στο αρχείο (δηλαδή, τους κόμβους που δεν έχουν κάνει αυτοί κάποιο tweet), το πλήθος των χρηστών που αφαιρώ είναι

```
Total deleted users: 422128
```

Οπότε, το πλήθος των χρηστών που κάποιος έχει αναμεταδώσει ένα tweet τους και οι ίδιο υπάρχουν στο αρχείο είναι

```
Retweet users: 13350
```

Οπότε καταλήγω μέχρι στιγμής να έχω,

```
Nodes before: 528386
Edges before: 1737282
Nodes after: 144973
Edges after: 74550
```

Στην συνέχεια πρέπει να κάνω την επαναληπτική αφαίρεση των κόμβων που έχουνε βαθμό μικρότερο από 10, τα αποτελέσματα σταδιακά που βρίσκω είναι τα ακόλουθα,

```
Nodes before: 2436
Nodes after: 1698
Nodes before: 1698
Nodes after: 1558
Nodes before: 1558
Nodes after: 1531
Nodes before: 1531
Nodes after: 1523
Nodes before: 1523
Nodes after: 1520
Nodes before: 1520
Nodes after: 1516
Nodes before: 1516
Nodes after: 1513
Nodes before: 1513
Nodes after: 1513
```

Τέλος, καταλήγω να έχω συνολικά,

```
Nodes: 1513   ['tonygin2000', 'kim_granny', 'eddie736', 'jax_crab', 'csilberman70']
Edges: 21008 [('tonygin2000', 'ahtlam'), ('tonygin2000', 'Scarlett210'), ('tonygin2000', 'rooshv')]
```

Πρέπει να δουλέψω με την μεγαλύτερη συνεκτική συνιστώσα που υπάρχει στο γράφημα μου, στο δικό μου γράφημα έχω μία και αυτή είναι το ίδιο μου το γράφημα.

```
Total number of coherent component : 1
Nodes: 1513
Edges: 21008
```

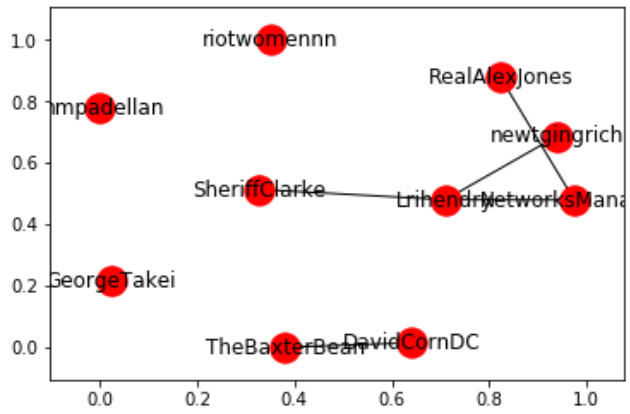
Τώρα πρέπει να βρω πόσοι χρήστες από αυτούς που έχω είναι ακόλουθοι του Trump και πόσοι είναι ακόλουθοι της Clinton. Για να το βρω αυτό θα χρησιμοποιήσω τα δεδομένα που έχει στο αρχείο "**clinton_trump_user_classes.txt**", επειδή εγώ έχω αποθηκεύσει τους χρήστες με τα usernames τους και το αρχείο τους έχει μέσα με το id τους θα πρέπει να κάνω την αντιστοιχία αυτή. Βρίσκω ότι,

```
Trump followers: 665
Trump followers %: 43.95241242564442
Clinton followers: 848
Clinton followers %: 56.04758757435558
```

Αλγόριθμος PageRank

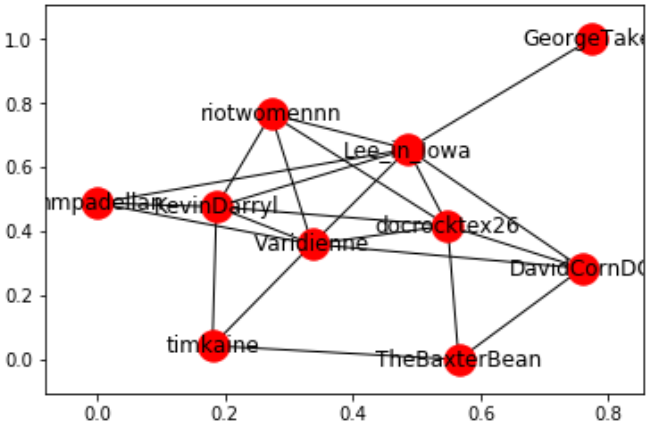
Τρέχω τον αλγόριθμο pageRank με τις default τιμές του, αυτό το οποίο μου επιστρέφει είναι ένα λεξικό και το περιεχόμενο του είναι σαν κλειδί οι κόμβοι και σαν τιμές το pagerank του κάθε κόμβου. Οι 10 καλύτεροι κόμβοι είναι οι εξής,

```
('newtgingrich', 0.008931499481883208)
('Lrihendry', 0.008517469611928785)
('SheriffClarke', 0.008161420548285208)
('riotwomennn', 0.007431626046229551)
('mmpadellan', 0.007228417497781389)
('DavidCornDC', 0.007010635139303156)
('GeorgeTakei', 0.006514185455234757 5)
('TheBaxterBean', 0.006114907640587179)
('RealAlexJones', 0.005968180394790666)
('NetworksManager', 0.005917498943030051)
```



Αλγόριθμος HITS

Τρέχω τον αλγόριθμο hits με τις default τιμές του, και κρατάω τα hubs και τα authorities που μου επιστρέφει. Ταξινομώ το λεξικό hubs σε φθίνουσα σειρά ώστε να βρω τους 10 καλύτερους κόμβους και οι οποίοι είναι οι ακόλουθοι,



```
('riotwomennn', 0.00887083677888395)
('mmpadellan', 0.008170831052727659)
('DavidCornDC', 0.007595813326102171)
('GeorgeTakei', 0.0069708865006572465)
('TheBaxterBean', 0.006948449267683307)
('Lee_in_Iowa', 0.006562001507445762)
('docrocktex26', 0.006480539500114367)
('Varidienne', 0.006395788258693659)
('KevinDarryl', 0.005972947099166974)
('timkaine', 0.005737364266618995)
```

Εφόσον αυτοί οι κόμβοι που έχουν προκύψει και από τους δυο αλγορίθμους έχουν μεγάλο pagerank και hub-authority αντίστοιχα σημαίνει ότι δείχνουν σε αυτούς τους κόμβους κάποιοι άλλοι κόμβοι οι οποίοι είναι σημαντικοί. Και όπως βλέπουμε και από τα γραφήματα στον σημαντικότερο από τους 10 κόμβους πηγαίνουν άλλοι επίσης σημαντικοί κόμβοι.

Σε αυτό το σημείο θα προσπαθήσουμε να ξεχωρίσουμε τους followers του Trump και της Clinton χρησιμοποιώντας αλγορίθμους για community detection.

Αλγόριθμος Girvan-Newman

Τρέχω τον αλγόριθμο που υπολογίζει το betweenness_centrality του γραφήματος.

Χρόνος Girvan-Newman για να αφαιρέσω διαφορετικό πλήθος από ακμές

Number of edges	1	100	500	1000
Time(seconds)	179.9900677525264	17999.006775252637	89995.03387626319	179990.06775252637

Τον χρόνο για να αφαιρέσω 100, 500, και 1000 ακμές τον υπολόγισα προσεγγιστικά, δηλαδή αν για να αφαιρέσει 1 ακμή ο αλγόριθμος θέλει χρόνο 179.9900677525264 seconds τότε για να αφαιρέσει 100 ακμές θα θέλει τον 100πλάσιο χρόνο και βασικά λίγο λιγότερο γιατί θα έχει να υπολογίσει το betweenness μείον των γραμμών που έχει αφαιρέσει.

Τροποποίηση αλγορίθμου Girvan-Newman

Τώρα πρέπει να εκτελέσω τον αλγόριθμο με μία παραλλαγή, να αφαιρώ κάθε φορά όχι μία ακμή αλλά Κ επαναληπτικά και να βλέπω αν ήτανε επιτυχής η διάσπαση, δηλαδή αν χωρίστηκαν καλά οι ακόλουθοι του Trump και της Clinton. Στην μεταβλητή "**sortedBC**" έχω ταξινομημένες σε φθίνουσα σειρά τις ακμές που μου επιστρέφει το **nx.edge_betweenness_centrality(G)** του γραφήματος. Έχω φτιάξει μία συνάρτηση "**def runAlgorithm(iterations)**" η οποία δέχεται σαν όρισμα τις διαφορετικές τιμές του Κ(ξεκινάω να το τρέχω από τ0 500 και στην συνέχεια καταλήγω στο 20), έχω ακόμη και την συνάρτηση "**def statistics(sg)**" που υπολογίζει όλα τα στατιστικά που επιθυμώ.

- **def runAlgorithm(iterations)** → Στην λίστα "**ll**" κρατάω τα ζευγάρια από τις ακμές σε φθίνουσα σειρά. Στην μεταβλητή "**start**" κρατάω την έναρξη του χρόνου, το "**t = 1**" δηλώνει τις επαναλήψεις που έχω κάνει μέχρι να τελειώσει η επαναληπτική αφαίρεση ακμών, το "**subCounter**" δείχνει πόσα υπογραφήματα έχω και το "**addAfter**" είναι μία λίστα όπου κρατάω όλες τις ακμές που αφαιρώ ώστε όταν τελειώσει αυτή η επανάληψη για να πάω στην

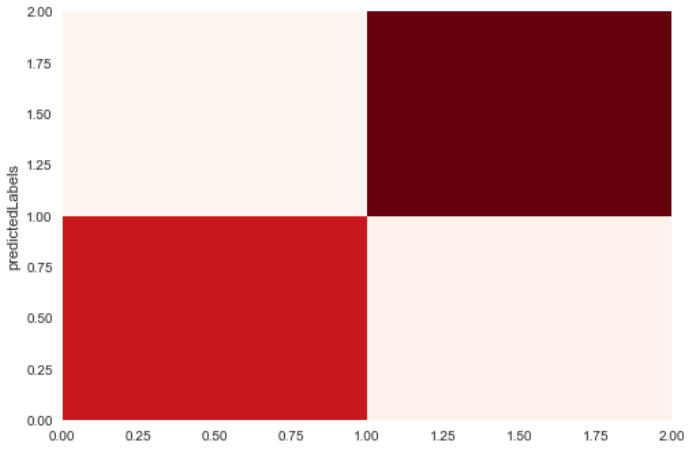
επόμενη(δηλαδή για διαφορετικό K) θα πρέπει να έχω όλο το γράφημα μου πάλι από την αρχή οπότε θα πρέπει να ξανά προσθέσω τις ακμές που αφαίρεσα. Μέσα σε μία while(έως ότου το subCounter <= 1 δηλαδή θα αφαιρώ ακμές συνεχόμενα μεγέθους K μέχρι το γράφημα του να χωριστεί σε 2 ή περισσότερα κομμάτια) κρατάω στην μεταβλητή "**top**" τις καλύτερες K ακμές από την λίστα "**ll**", τις προσθέτω και στην λίστα "**addAfter**" και έπειτα τις αφαιρώ από το γράφημα, υπολογίζω το πλήθος από τα υπογραφήματα που έχουνε δημιουργηθεί και αν είναι >= 2 τότε σταματάω την επανάληψη, διαφορετικά πρέπει να ξανά υπολογίσω το "**nx.edge_betweenness_centrality(G)**" του γραφήματος και να ανανεώσω την λίστα "**ll**". Όταν τελειώσει η while τότε βρίσκω τον χρόνο που χρειάστηκε για όλες αυτές τις επαναλήψεις. Επιστρέφω την λίστα "**return(addAfter)**" ώστε και μετά που θα υπολογίσω τα στατιστικά να ξανά προσθέσω τις ακμές.

- **def statistics(sg)** → Παίρνει σαν όρισμα τα υπογραφήματα για το κάθε διαφορετικό K, στην μεταβλητή "**nodesList**" κρατάω τα ονόματα των κόμβων που ανήκουν σε κάθε υπογράφημα, η λίστα "**trueLabels**" κρατάει τις πραγματικές τιμές από το 0 ή 1 δηλαδή αν είναι ακόλουθος του Trump ή της Clinton, το "**zeroOneList**" κρατάει τις τιμές που βρίσκουμε από το γράφημα, αν σε ένα υπογράφημα η πλειοψηφία των κόμβων είναι ακόλουθοι του ενός πολιτικού τότε και οι υπόλοιποι που ανήκουν στο συγκεκριμένο υπογράφημα παίρνουν την τιμή του πολιτικού που ακολουθεί η πλειοψηφία. Με αυτήν την λογική βλέπω αν είχε επιτυχία μία διάσπαση του γραφήματος. Στην συνέχεια για το κάθε υπογράφημα υπολογίζω τον αλγόριθμο PageRank και τον HITS και κρατάω τους 10 καλύτερους κόμβους. Υπάρχουν περιπτώσεις που ένα υπογράφημα έχει 1 μόνο κόμβο οπότε δεν μπορώ να υπολογίσω τον αλγόριθμο HITS και υπάρχουν περιπτώσεις που ένα υπογράφημα δεν έχει περισσότερους από 10 κόμβους οπότε αντί για τους 10 καλύτερους κόμβους επιστρέφω όσους έχει σε φθίνουσα σειρά.

Αποτελέσματα

K = 500
Επαναλήψεις: 3
Χρόνος: 336.9332993625651 seconds
Μέση επιτυχία: 97.63982331436591%
Confusion matrix:

```
[[631  34]
 [ 19 829]]
```



1ο υπογράφημα

PageRank

```
('Lrihendry', 0.020778549624847074)
('newtgingrich', 0.020693198360199224)
('SheriffClarke', 0.019763996768852883)
('RealAlexJones', 0.014722262961328834)
('roycan79', 0.014535362357836357)
('NetworksManager', 0.014513688605868321)
('AnnCoulter', 0.012732636244668501)
('singernews', 0.00960980388531781)
('HeyTammyBruce', 0.008817552065621037)
('joelpollak', 0.008610608092849605)
```

HITS

```
('Lrihendry', 0.01365379319698627)
('newtgingrich', 0.01242736737260801)
('SheriffClarke', 0.012009036345671308)
('NetworksManager', 0.009959085895632725)
('roycan79', 0.009685341309535946)
('RealAlexJones', 0.009540343277979526)
('AnnCoulter', 0.007723659094334587)
('Scarlett210', 0.006033544455781519)
('gs777gs777', 0.005955957229973127)
('Juliet777777', 0.005932274281262001)
```

2ο υπογράφημα

PageRank

```
('riotwomennn', 0.013412603366969828)
('mmpadellan', 0.013050355591450126)
('DavidCornDC', 0.012495714317929375)
('GeorgeTakei', 0.011657036325984796)
('TheBaxterBean', 0.010848356075593287)
('docrocktex26', 0.009503901461745153)
('timkaine', 0.009031209086225905)
('Slate', 0.008407898302190391)
('Lee_in_Iowa', 0.008088840612267627)
('KevinDarryl', 0.00777226119690073)
```

HITS

```
('riotwomennn', 0.009566496228447593)
('mmpadellan', 0.008805548289792308)
('DavidCornDC', 0.008132103082218296)
('GeorgeTakei', 0.00748441011434733)
('TheBaxterBean', 0.0073620628486044O7)
('Lee_in_Iowa', 0.007061172988013255)
('docrocktex26', 0.006999027906801186)
('Varidienne', 0.006895659403018608)
('KevinDarryl', 0.006470999416713855)
('timkaine', 0.006123025458117605)
```

3ο υπογράφημα

PageRank

```
('Sparblack1213', 1.0)
```

4ο υπογράφημα

PageRank

```
('KHerriage', 1.0)
```

5ο υπογράφημα

PageRank

```
('kleegrubaugh', 1.0)
```

6ο υπογράφημα

PageRank

```
('Oenonewept', 1.0)
```

7ο υπογραφημα

PageRank

```
('ChinaCatSun', 1.0)
```

8ο υπογράφημα

PageRank

```
('HotlineJosh', 0.3987949981601664)
('RAMrants', 0.3817175683766465)
('Heminator', 0.21948743346318683)
```

HITS

```
('RAMrants', 0.44504186825969294)
('HotlineJosh', 0.35689586676529916)
('Heminator', 0.19806226497500784)
```

9ο υπογράφημα

PageRank

```
('DanWheatley4', 0.6491226380641377)
('frontlinepbs', 0.3508773619358619)
```

HITS

```
('DanWheatley4', 0.618033988205325)
('frontlinepbs', 0.38196601179467493)
```

10ο υπογράφημα

PageRank

```
('Bygd1', 1.0)
```

11ο υπογράφημα

PageRank

```
('RobGeorge', 1.0)
```

12ο υπογράφημα

PageRank

```
('ggreenwald', 1.0)
```

13ο υπογράφημα

PageRank

```
('Benito35ddDavis', 1.0)
```

14ο υπογράφημα

PageRank

```
('NamelessCulture', 1.0)
```

15ο υπογράφημα

PageRank

```
('4AllSoulKind', 1.0)
```

16ο υπογράφημα

PageRank

```
('froomkin', 1.0)
```

17ο υπογράφημα

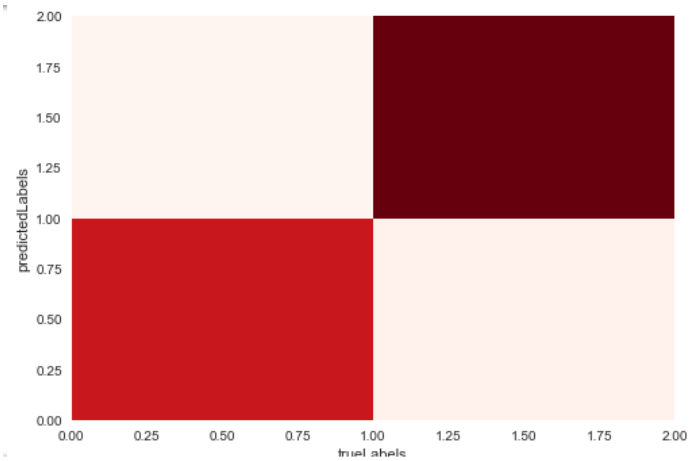
PageRank

```
('property1', 1.0)
```

Συμπέρασμα: Στο 1ο και 2ο υπογράφημα οι καλύτεροι 10 κόμβοι συμπίπτουν με τα αποτελέσματα του γραφήματος χωρίς να αφαιρώ ακμές.

K = 250
Επαναλήψεις: 4
Χρόνος: 468.9127557014742 seconds
Μέση επιτυχία: 99.4772016763279%
Confusion matrix:

```
[[632  33]
 [ 19 829]]
```



1ο υπογράφημα

PageRank

```
('newtgingrich', 0.021040328845272956)
('Lrihendry', 0.020208177008159817)
('SheriffClarke', 0.01816671389312046)
('NetworksManager', 0.01439140481355378)
('RealAlexJones', 0.014390960004785342)
('roycan79', 0.014240585262083548)
('AnnCoulter', 0.012772770790923352)
('singernews', 0.009473593191795523)
('HeyTammyBruce', 0.008585651154891228)
('joelpollak', 0.008542092187367597)
```

HITS

```
('Lrihendry', 0.013147344469332417)
('newtgingrich', 0.012392080332165689)
('SheriffClarke', 0.011299388399913443)
('NetworksManager', 0.00995501379628607)
('RealAlexJones', 0.009376817755921618)
('roycan79', 0.009364918887003786)
('AnnCoulter', 0.00767104706585338)
('Scarlett210', 0.005903585128876195)
('gs777gs777', 0.005865728275740918)
('Juliet777777', 0.0057392264106211925)
```

2ο υπογράφημα

PageRank

```
('riotwomennn', 0.01337039773924598)
('mmpadellan', 0.012969502713163066)
('DavidCornDC', 0.012518419749952244)
('GeorgeTakei', 0.011786257273650295)
('TheBaxterBean', 0.010853892104603233)
('docrocktex26', 0.00934802153641744)
('timkaine', 0.009307800028885803)
('Slate', 0.008557012693301058)
('Lee_in_Iowa', 0.008076490807348395)
('KevinDarryl', 0.007748185701562943)
```

HITS

```
('riotwomennn', 0.009524249167667559)
('mmpadellan', 0.008758643580588326)
('DavidCornDC', 0.008097734697819083)
('GeorgeTakei', 0.007475808060624111)
('TheBaxterBean', 0.0073239708828726265)
('Lee_in_Iowa', 0.007014974474920685)
('docrocktex26', 0.00685397266779013)
('Varidienne', 0.006747769769034612)
('KevinDarryl', 0.006446312257584613)
('timkaine', 0.006106539218906269)
```

3ο υπογράφημα

PageRank

('TripawDaisy', 1.0)

4ο υπογράφημα
PageRank

('Sparblack1213', 1.0)

5ο υπογράφημα
PageRank

('kleegrubaugh', 1.0)

6ο υπογράφημα
PageRank

('Oenonewept', 1.0)

7ο υπογράφημα
PageRank

('Heminator', 1.0)

8ο υπογράφημα
PageRank

('jpwilloughby', 1.0)

9ο υπογράφημα
PageRank

('HotlineJosh', 1.0)

10ο υπογράφημα
PageRank

('RobGeorge', 1.0)

11ο υπογράφημα
PageRank

('BarrieNJ', 1.0)

12ο υπογράφημα
PageRank

('MGoesler', 1.0)

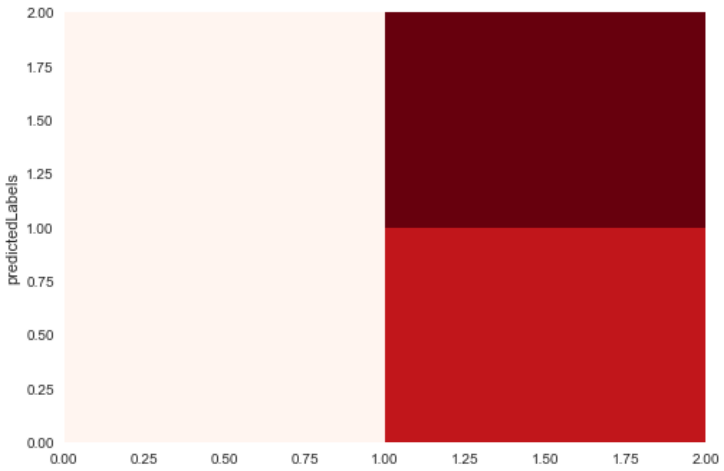
13ο υπογράφημα
PageRank

('TheStalwart', 1.0)

Συμπέρασμα: Στο 1ο και 2ο υπογράφημα οι καλύτεροι 10 κόμβοι συμπίπτουν με τα αποτελέσματα του γραφήματος χωρίς να αφαιρώ ακμές.

K = 100
Επαναλήψεις: 6
Χρόνος: 900.6638198035894 seconds
Μέση επιτυχία: 85.35186410765498%
Confusion matrix:

```
[[   1  664]
 [   0  848]]
```



1ο υπογράφημα
PageRank

('newtgingrich', 0.009118848247196635)
('Lrihendry', 0.008843757712558842)
('SheriffClarke', 0.008427133481235614)
('riotwomennn', 0.00744387367481426)
('mmpadellan', 0.007332534231164927)
('DavidCornDC', 0.007055417000550637)
('GeorgeTakei', 0.006598803068916929)
('RealAlexJones', 0.006190929573796133)
('TheBaxterBean', 0.006136590809309848)
('NetworksManager', 0.006122530751987624)

HITS

('riotwomennn', 0.009326277733496544)
('mmpadellan', 0.00863134783333872)
('DavidCornDC', 0.00799410366758237)
('GeorgeTakei', 0.007346037613767311)
('TheBaxterBean', 0.007303304223668688)
('Lee_in_Iowa', 0.006922528856784735)
('docrocktex26', 0.006839547521326934)
('Varidienne', 0.006684359962192671)
('KevinDarryl', 0.006310945534410338)
('timkaine', 0.006052347009953569)

2ο υπογράφημα
PageRank

```
('RobGeorge', 1.0)
```

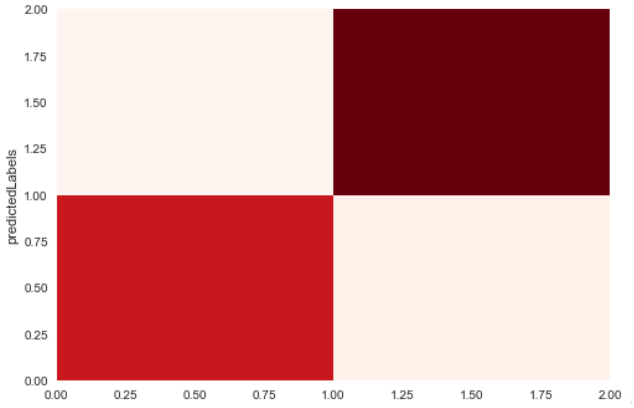
3ο υπογράφημα
PageRank

```
('NamelessCulture', 1.0)
```

Συμπέρασμα: Στο 1ο υπογράφημα οι καλύτεροι 10 κόμβοι συμπίπτουν με τα αποτελέσματα του γραφήματος χωρίς να αφαιρώ ακμές.

K = 50
Επαναλήψεις: 11
Χρόνος: 1701.1778273117734 seconds
Μέση επιτυχία: 98.9990859936225 %
Confusion matrix:

```
[[630  35]
 [ 19 829]]
```



1ο υπογράφημα
PageRank

```
('newtgingrich', 0.021400235291276357)
('Lrihendry', 0.020689246259901033)
('SheriffClarke', 0.019770648348974625)
('RealAlexJones', 0.014499334210190236)
('NetworksManager', 0.01427575920113994)
('roycan79', 0.014254517089412356)
('AnnCoulter', 0.012620751661421334)
('singernews', 0.009460228118050123)
('HeyTammyBruce', 0.008704255065744206)
('joelpollak', 0.008630992302699661)
```

HITS

```
('Lrihendry', 0.01312054799340763)
('newtgingrich', 0.012288112776229734)
('SheriffClarke', 0.011669075844312588)
('NetworksManager', 0.009690238544631 54)
('roycan79', 0.009194361485282574)
('RealAlexJones', 0.009152770084452966)
('AnnCoulter', 0.007467862166366947)
('Scarlett210', 0.00580000525340989)
('gs777gs777', 0.005739200170221909)
('Juliet777777', 0.005558064342073505)
```

2ο υπογράφημα
PageRank

```
('riotwomennn', 0.013178630558422657)
('mmpadellan', 0.012806803395020548)
('DavidCornDC', 0.01232181547458221)
('GeorgeTakei', 0.011436260986485283)
('TheBaxterBean', 0.01081665917280748)
('timkaine', 0.009270758266038126)
('docrocktex26', 0.009265229477633648)
('Slate', 0.008592350311169065)
('Lee_in_Iowa', 0.008059476779676462)
('KevinDarryl', 0.007570221436990444)
```

HITS

```
('riotwomennn', 0.009289815466214265)
('mmpadellan', 0.008554061589586406)
('DavidCornDC', 0.007898025150041392)
('TheBaxterBean', 0.00733555661340397)
('GeorgeTakei', 0.007241508787525514)
('Lee_in_Iowa', 0.006923074293242 8705)
('docrocktex26', 0.006829641141880854)
('Varidienne', 0.006690666242048946)
('KevinDarryl', 0.006315929990767412)
('timkaine', 0.005975807788794711)
```

3ο υπογράφημα
PageRank

```
('Heminator', 0.5)
('Oenonewept', 0.5)
```

HITS

```
('Heminator', 0.5)
('Oenonewept', 0.5)
```

4ο υπογράφημα
PageRank

```
('Szrti716', 1.0)
```

5ο υπογράφημα
PageRank

```
('HotlineJosh', 1.0)
```

6ο υπογράφημα
PageRank

```
('RobGeorge', 1.0)
```

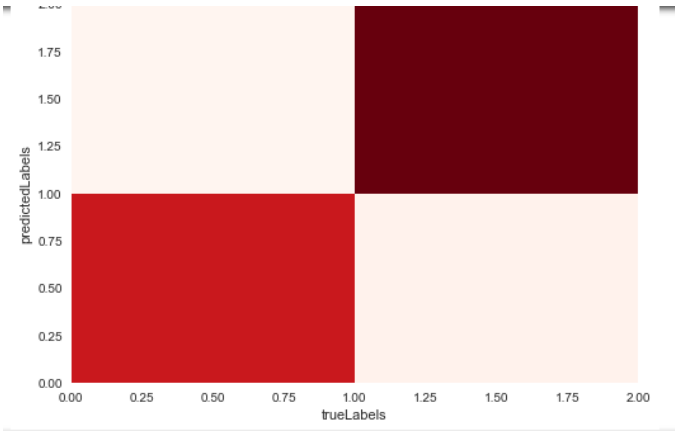
7ο υπογράφημα
PageRank

```
('vintagegoddess', 1.0)
```

Συμπέρασμα: Στο 1ο και 2ο υπογράφημα οι καλύτεροι 10 κόμβοι συμπίπτουν με τα αποτελέσματα του γραφήματος χωρίς να αφαιρώ ακμές.

K = 20
Επαναλήψεις: 24
Χρόνος: 4247.587440515937 seconds
Μέση επιτυχία: 97.51851435565045%
Confusion matrix:

```
[[630  35]
 [ 22 826]]
```



1ο υπογράφημα

PageRank

```
('newtgingrich', 0.021262293653381822)
('Lrihendry', 0.020627800816728298)
('SheriffClarke', 0.019722161469272595)
('RealAlexJones', 0.01440348998195661)
('NetworksManager', 0.014254315249156478)
('roycan79', 0.014229449350020094)
('AnnCoulter', 0.012630704302728944)
('singernews', 0.009611772132141172)
('HeyTammyBruce', 0.008715611134004765)
('joelpollak', 0.008596811240440688)
```

HITS

```
('Lrihendry', 0.013084185412513182)
('newtgingrich', 0.01223391348743046)
('SheriffClarke', 0.011640829280987558)
('NetworksManager', 0.009669124948574491)
('roycan79', 0.009186666051708082)
('RealAlexJones', 0.009116731447553936)
('AnnCoulter', 0.007432217746475381)
('Scarlett210', 0.00579612224937206)
('gs777gs777', 0.005718583792310398)
('Juliet777777', 0.005534638512724513)
```

2ο υπογράφημα

PageRank

```
('riotwomennn', 0.013165663861253835)
('mmpadellan', 0.012800927501173823)
('DavidCornDC', 0.012340962791802024)
('GeorgeTakei', 0.011508057858985595)
('TheBaxterBean', 0.010783949518814341)
('timkaine', 0.009273772290550491)
('docrocktex26', 0.009237432664335854)
('Slate', 0.008554567625137064)
('Lee_in_Iowa', 0.008034063286431832)
('KevinDarryl', 0.007548701944311027)
```

HITS

```
('riotwomennn', 0.009322463700127697)
('mmpadellan', 0.008589218309431096)
('DavidCornDC', 0.007968821237793215)
('GeorgeTakei', 0.007317113981379121)
('TheBaxterBean', 0.007294778548496983)
('Lee_in_Iowa', 0.006897002765523807)
('docrocktex26', 0.006808159115288752)
('Varidienne', 0.006660114461296011)
('KevinDarryl', 0.006279190936812585 4)
('timkaine', 0.006026518489450861)
```

<u>3ο υπογράφημα</u>
PageRank

```
('kleegrubaugh', 1.0)
```

<u>Συμπέρασμα</u>: Στο 1ο και 2ο υπογράφημα οι καλύτεροι 10 κόμβοι συμπίπτουν με τα αποτελέσματα του γραφήματος χωρίς να αφαιρώ ακμές.

Ερώτηση 4

Δεν την έχω υλοποιήσει αλλά θα σας αναφέρω με ποιον τρόπο θα δοκίμαζα να την υλοποιήσω. Θα δοκίμαζα να υλοποιήσω τον SVD αλγόριθμο. Δηλαδή θα είχα έναν “πίνακα” αραιό και σαν γραμμές θα θεωρούσα τους χρήστες και σαν στήλες θα θεωρούσα τα καλύτερα και πιο συχνά hashtags/handles αυτών που ακολουθούν τον Trump και αυτών που ακολουθούν την Clinton. Ο πίνακας θα γέμιζε με το πλήθος των φορών που κάποιος χρήστης χρησιμοποιεί ένα hashtag/handle. Με το SVD θα έβλεπα αν μπορεί να μειωθεί η διάσταση. Φαντάζομαι ότι θα γινότανε μία τέτοια μείωση. Θα έβλεπα ότι το rank του αρχικού πίνακα είναι μεγαλύτερο από το μέγεθος του πίνακα U (τετραγωνικός). Θα έβλεπα πόση διαφορά υπάρχει και θα διέγραφα τις στήλες/γραμμές που δεν χρειαζότανε σαν το παράδειγμα που είχαμε στις διαφάνειες με το πως να αφαιρέσουμε τον θόρυβο. Και μετά από αυτό θα είχα έναν καινούριο τελικό πίνακα και θα είχανε γεμίζει με κάποια τιμή οι θέσεις που είχανε μηδέν συχνότητα κάποιου hashtag/handle. Και θα έβλεπα κατά πόσο οι χρήστες της Clinton χρησιμοποιούν ένα συχνό hashtag/handle που χρησιμοποιούν πολύ οι χρήστες του Trump. Και θα έβλεπα ίσως αν υπάρχει κάποιο χάσμα ανάμεσα στους ακόλουθους του Trump και της Clinton. Τον τρόπο αυτόν δεν τον έχω υλοποιήσει.