# Agrupación de Noticieros en Tweeter

Francisco Alonso, Miguel Astor, Jesús Hernandez, Javier Machado, Yilber Sisco

April 14, 2016

# Propuesta

Tomando una muestra de tweets de cuentas de diversos noticieros de Latinoamérica, agrupar cada cuenta según las características de los tweets que publica.

Agrupar un subconjunto de los noticieros latinoamericanos presentes en Twitter, puede proveer información sobre similitudes o diferencias entre noticieros de varios países.

## Objetivo

Obtener informacón sobre similitudes y diferencias entre noticieros de Latinoamerica presentes en Twitter.

## Factores que determinan el éxito del proceso

Las características de la data influyen enormemente en el proceso de agrupación, se intentará determinar que modelos se adaptan mejor a estas características y permiten crear grupos con más precisión.

## Fuente de Datos

# Objetivos del proceso de minería de datos

- Agrupar a los noticieros y obtener un conjunto de clases que describan las diferencias y similitudes entre cada grupo de noticieros. Tarea de MD: Agrupación o Clustering.
- Validar los modelos creados para seleccionar el que proporciones mejores resultados.

# Recolección de datos

Se autentica la aplicación contra Twitter y se obtiene el usuario "Proyectominería", luego se obtienen los usuarios que este sigue.

```
setup_twitter_oauth(api_key, api_secret, token, token_sec
theUser <- twitteR::getUser(user = "Proyectomineria")
followingList <- theUser$getFriends()
```

Por cada usuario seguido se obtienen 50 tweets que son limpiados y lematizados. Se almacena el nombre de usuario y sus tweets como un sólo texto.

```
userTweetsHelper <- userTimeline(followingList[[i]], 50,
userTweetsHelper = sapply(userTweetsHelper, function(x) x
userTweetsHelper = removeWords(userTweetsHelper, c(stopwo
userTweetsHelper = removeWords(userTweetsHelper,sw)
userTweetsHelper = cleanme(userTweetsHelper)
userTweetsHelper <- paste(userTweetsHelper, collapse= " "
userTweets <- c(userTweets, userTweetsHelper)
userNames <- c(userNames, followingList[[i]]$name)
```

# Preparación de los datos

Se cargan el archivo en un data.frame y se procede a crear el objeto Corpus con todos los documentos.

```
allUserTweets <- rbind(file1, file2)
allUserTweets <- select(allUserTweets, userNames, userTwe
corpuses <- Corpus(VectorSource(allUserTweets$userTweets)
corpuses <- tm_map(corpuses, removeWords, myStopwords)
```

Se genera la matriz Término Documento calculando los pesos usando TF-IDF, esta muestra cada término y la cantidad de veces que se encontró ese término entre los tweets de un usuario específico.

```
dtm <- DocumentTermMatrix(corpuses, control = list(remo
                                                   stopwo
                                                   weight
                                                     weig
```

# Minería de datos

Se probaron 3 modelos para este conjunto de datos: k-Medias, Pam y Clusterización Jerárquica (hclust). Se generaron 9 modelos para K de 2 a 10, luego se calculó el valor de Silueta para cada uno de esos K para realizar la comparación entre los modelos. Fue necesario calcular una matriz de distancia para calcular los modelos.

# k-Medias

```
kmeans.2 <- kmeans(m3, 2)
#...
kmeans.10 <- kmeans(m3, 10)

dissE = dist(m3)
sil2 <- silhouette(kmeans.2$cluster, dissE)
#...
sil10 <- silhouette(kmeans.10$cluster, dissE)

kmean.sil.values <- c(
summary(sil2)[["avg.width"]]
#...
,summary(sil10)[["avg.width"]])
```

# Pam

```
pam.2 <- pam(dist(m3),k=2,diss = T)
# ...
pam.10 <- pam(dist(m3),k=10,diss = T)

sil2 <- silhouette(pam.2)
#...
sil10 <- silhouette(pam.10)

pam.sil.values <- c(
summary(sil2)[["avg.width"]]
#...
,summary(sil10)[["avg.width"]])
```

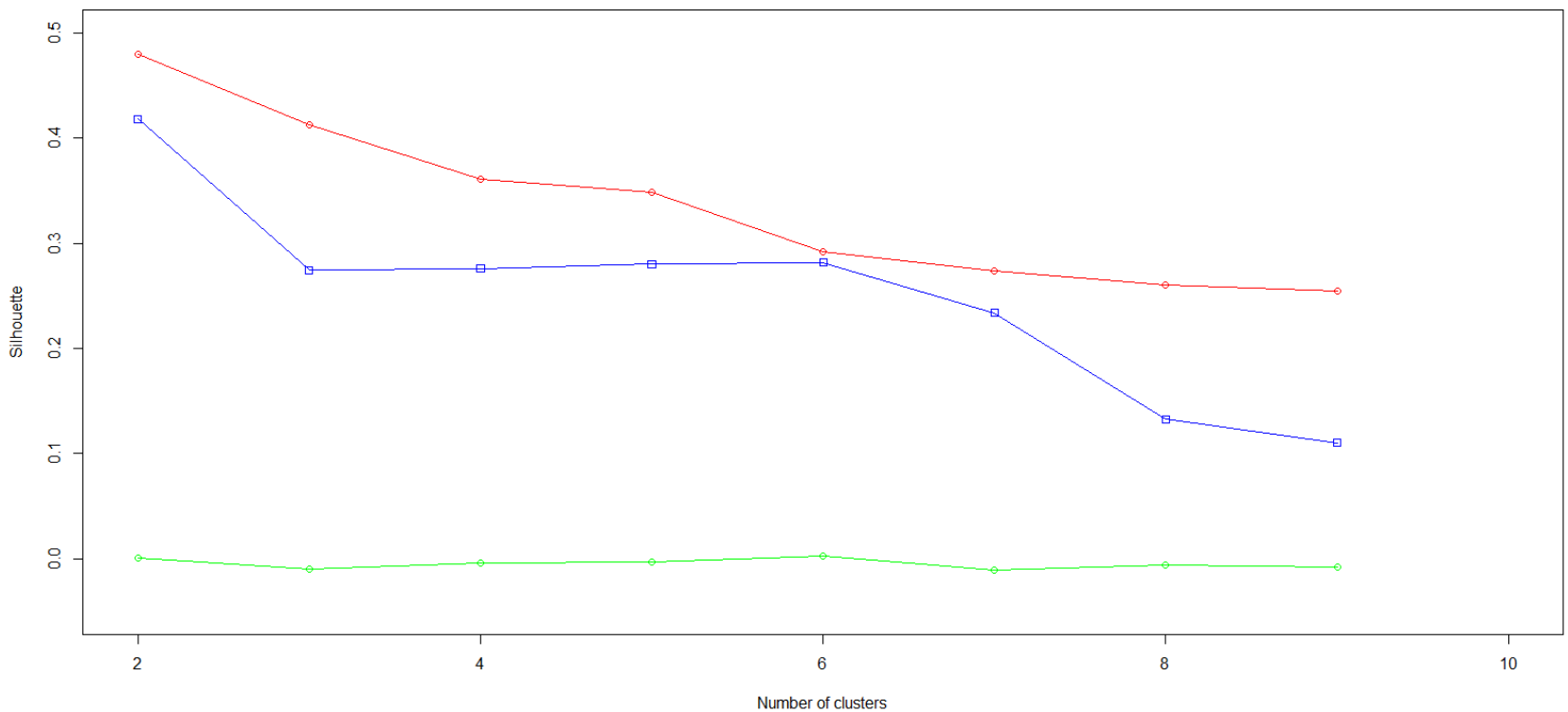
# Agrupación Jerárquica

```
hierarchical.clustering <- hclust(dist(m3),method="averag

hclust.2 <- cutree(hierarchical.clustering,k=2)
#...
hclust.10 <- cutree(hierarchical.clustering,k=10)



sil2 <- silhouette(hclust.2,dist=similarity.matrix)
#...
sil10 <- silhouette(hclust.10,dist=similarity.matrix)

hclust.sil.values <- c(
  summary(sil2)[["avg.width"]]
  #...
  ,summary(sil10)[["avg.width"]])
```

# Evaluación

Se comparan los modelos en base a su valor de silueta para cada K.

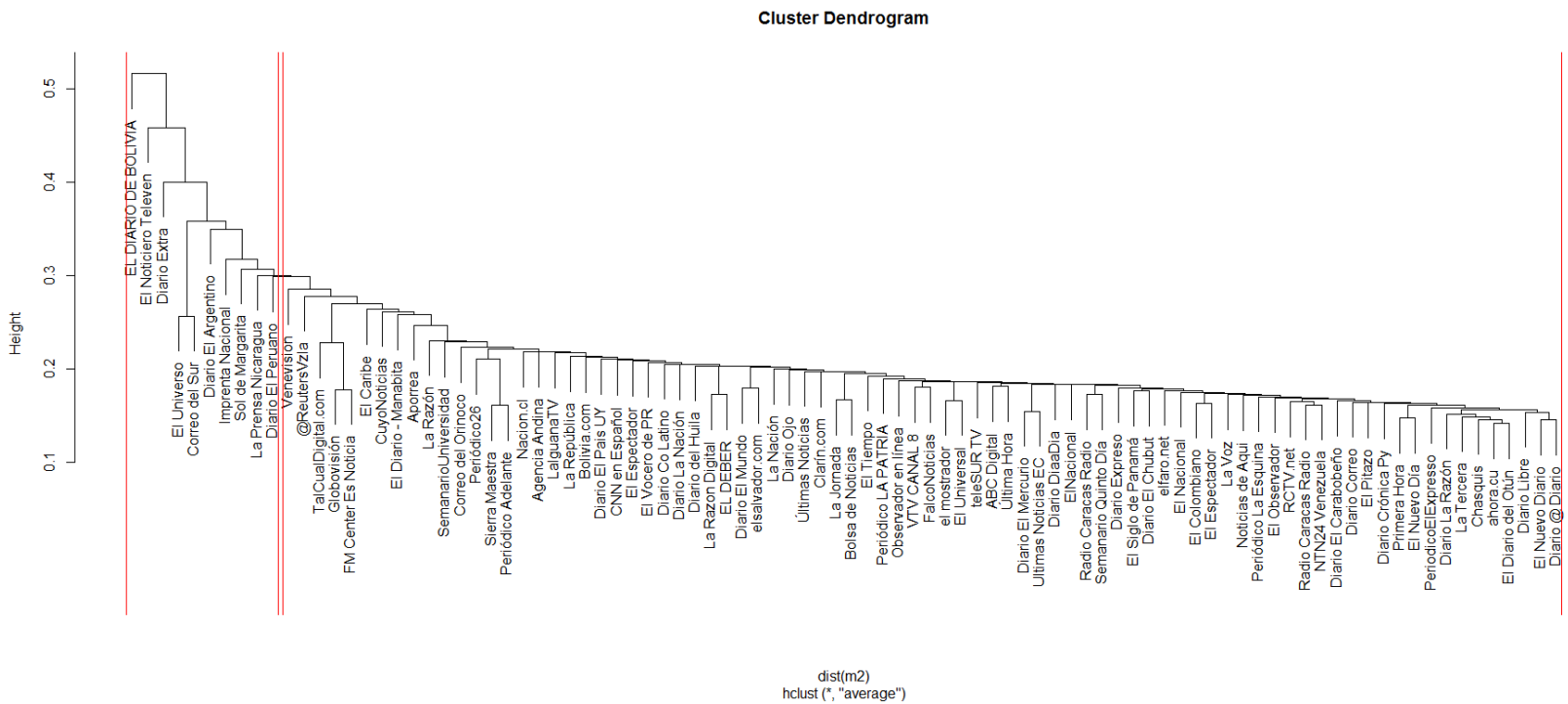
```
plot(2:10,kmean.sil.values[2:10],type="o",col="blue",pch=
lines(2:10,hclust.sil.values[2:10],type="o",col="red",pch
lines(2:10,pam.sil.values[2:10],type="o",col="green",pch=
```



# Evaluación

Se selecciona dado el mejor valor obtenido un K = 2 para el modelo arrojado por la Agrupación Jerárqica.

```
hierarchical.clustering <- hclust(dist(m3),method="averag
hclust.2 <- cutree(hierarchical.clustering,k=2)
```



# Interpretación

Se pudo notar que los noticieros ubicados en el primer grupo o cluster resultaron tener más tweets relacionados a tópicos como farándula, deportes o tecnología que acotencimientos y eventos de tipo político o geográfico en la región, dentro de la muestra tomada. Esto es un indicativo de la complejidad que tiene este tipo de datos.