

# Informe Proyecto 3: Grandes Volumnes de Datos

Santiago Uribe Pastás  
Juan Manuel Cuellar Borrero  
Nicolás Ibagón Rivera

Diciembre 2020

## Abstract

En el siguiente documento se presenta un informe acerca de los resultados obtenidos al predecir cierta variable basado en las medidas de un grafo VS predecir cierta variable de forma regular, es decir, con aprendizaje automático. Se trabajará con un data set llamado Wikipedia Norms (2015) [1]. Para el manejo de los grafos se hará uso de la base de datos Neo4j y para el aprendizaje automático se utilizara `pyspark`.

## 1. Data Set

Debido a que el data set utilizado en proyectos anteriores no es óptimo para la generación de un grafo no trivial a partir de la información que contiene, se decidió cambiar el data set por uno que permitiera la creación de un grafo del cual se pudieran obtener medidas topológicas relevantes. El nuevo data set se llama Wikipedia Norms, el cual fue tomado de The Colorado Index of Complex Networks [2]. El data set describe los nodos como las entradas o páginas de Wikipedia, y dos páginas están vinculadas por un arco dirigido si una se hiper-vincula (existe un link) a la otra. El data set cuenta con 1.976 nodos y 17.235 arcos.

### 1.1. Atributos

- PageID: Id de la página.
- Name: Nombre de la página.
- CreationDate : Fecha de creación de la página.
- Type: Tipo de articulo escrito en la página.
- Louvain Community: Método para detección de comunidades en el data-set, aquellos nodos que se encuentran por fuera del componente se encuentran como “-1”.

- Final In Degree: Cantidad de arcos entrantes a cierto nodo  $u$ .
- EC Estimate: Este valor cuantifica la importancia de una página (nodo) basado en su accesibilidad general dentro de la red.
- Fraction of total pageviews (July 2015): Número de vistas de pagina sobre el número de vistas de las páginas totales hasta Julio de 2015.
- Number of Edits: Número de ediciones hechas a la página.
- Unique Editors: Número de editores únicos de la página.
- Number of Talk Page Edits: Número de ediciones hechas en los Talk Page de Wikipedia en el página relacionada.
- Unique Talk Page Editors:Número de editores únicos en las Talk Page de Wikipedia asociadas a la página.
- Page size: Tamaño de la página en bytes.

## 1.2. Target

Para poder realizar las predicciones se selecciono como variable target “Number of Edits”, que tal y como su nombre lo indica el número de edits que se realizan en una pagina de Wikipedia. El machine learning se hace a través del método de regresión lineal.

## 1.3. Limpieza de datos

A la hora de realizar la limpieza de dato únicamente se eliminaron atributos que consideramos irrelevantes, por un lado se eliminó *PageId* pues el Id no aporta en nada para la predicción, de igual manera se eliminó el atributo *FinalInDegree* pues todos los valores de los registros eran 0. Por otro lado el atributo *Name* se eliminó pues no se podía transformar a un valor numérico relevante, de igual forma *CreationDate* se eliminó pues no sólo contenía la fecha sino también la hora y zona horaria de creación.

## 2. Medidas Topológicas

Las medidas topológicas escogidas para poder realizar las predicciones son las siguientes:

- Grado de Salida (Outdegree): La cantidad de página a las que redirecciona una página.
- Grado de Entrada (Indegree): La cantidad de de páginas que redireccionan a dicha página.

- Numero de Strongly Connected Component (SCC): Un componente fuertemente conexo es un subconjunto de nodos en el cual cada nodo es accesible (hay un camino) desde cualquier otro nodo perteneciente a su mismo componente. En el caso del data set: conjuntos de páginas que son alcanzables entre sí a través de links.
- Betweenness: Indica un puntaje de la cantidad de caminos más cortos que pasan por dicho nodo, por lo que aquellos nodos con puntuación alta son calificados como puentes entre 2 partes del grafo. En el caso del data set: las páginas que conectan temas distintos.
- Coeficiente de cercanía: Medida que indica la lejanía promedio a todos los demás nodos, se interpreta como las páginas que más información distribuyen de forma eficiente.

### 3. Neo4j

Para manejar la base de datos Neo4j se hizo uso de la librería py2neo con la que gracias a sus módulos `Node` se pudo crear cada uno de los nodos y con las diferentes funciones del modulo `Graph` se pudieron crear los arcos y evaluar las querys para obtener las métricas respectivas.

Para el calculo de las métricas se utilizaron las siguientes querys:

- **Indegree:**

```
match(p1:WikipediaPage) where p1.PageID= $u
match(p2:WikipediaPage) where (p2) -[:Hyperlink]->(p1) return
count(p2)
```

Como primer paso recorremos los nodos con un ciclo y buscamos una página de tipo `WikipediaPage` la cual llamaremos `p1`, la cual su `PageID` coincida con el iterador del ciclo (en este caso `u`). Seguidamente se busca otra pagina `p2` la cual tenga una relación de tipo `HyperLink` con `p1`, es decir, `p2` apunta a `p1`. Para finalizar y determinar cuento Indegree tiene el nodo `u` se procede a utilizar la función `count` con respecto a `p2`.

- **Outdegree:**

```
match(p1:WikipediaPage) where p1.PageID= $u
match(p2:WikipediaPage) where (p1) -[:Hyperlink]->(p2) return
count(p1)
```

Como primer paso recorremos los nodos con un ciclo y buscamos una página de tipo `WikipediaPage` la cual llamaremos `p1`, la cual su `PageID` coincida con el iterador del ciclo (en este caso `u`). Seguidamente se busca otra pagina `p2` la cual tenga una relación de tipo `HyperLink` de `p1` a `p2`, es decir, `p1` apunta a `p2`. Para finalizar y determinar cuento Outdegree tiene el nodo `u` se procede a utilizar la función `count` con respecto a `p1`.

- **Numero de SCC:**

```
CALL gds.alpha.scc.stream(nodeProjection: 'WikipediaPage',
  relationshipProjection: 'Hyperlink')
```

Se calculan los SCC con la anterior query y se retorna una lista donde el índice de la lista representa el nodo y el valor de la lista en ese índice representa el SCC al que pertenece dicho nodo.

- **Betweenness:**

```
CALL gds.betweenness.stream(nodeProjection: 'WikipediaPage',
  relationshipProjection: 'Hyperlink')
```

Se calculan los valores de betweenness con la anterior query y se retorna una lista donde el índice de la lista representa el nodo y el valor de la lista en ese índice representa el valor de betweenness de dicho nodo.

- **Coeficiente de cercanía:**

```
CALL gds.alpha.closeness.stream(nodeProjection: 'WikipediaPage',
  relationshipProjection: 'Hyperlink')
```

Se calculan los coeficientes de cercanía con la anterior query y se retorna una lista donde el índice de la lista representa el nodo y el valor de la lista en ese índice representa el coeficiente de cercanía de dicho nodo.

Las funciones usadas para el calculo de las medidas topológicas fueron tomadas de la pagina oficial de neo4j [3]. En la figura 1 se puede apreciar la representación de grafo del data set.

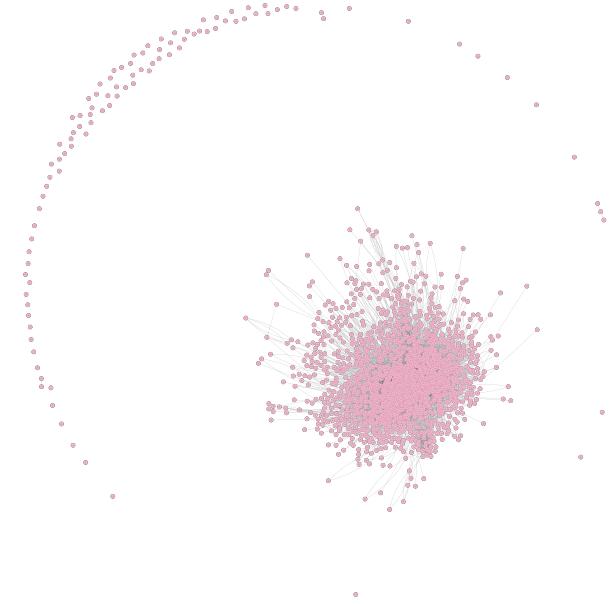


Figura 1: Data set como grafo

## 4. Comparación Métricas de Regresión

Los resultados obtenidos en las métricas de regresión para las predicciones sin el uso del grafo con la técnica de regresión lineal fueron:

Raíz del error cuadrático medio:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} = 224,19$$

El coeficiente de determinación o  $R^2$ :

$$R^2 = 0,883$$

Y para el error medio absoluto:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| = 50,61$$

Los resultados de las métricas de regresión obtenidas teniendo en cuenta las medidas topológicas del grafo son:

Raíz del error cuadrático medio:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} = 131,39$$

El coeficiente de determinación o  $R^2$ :

$$R^2 = 0,951$$

Y para el error medio absoluto:

$$MAE = \sum_{i=1}^N |y_i - \hat{y}_i| = 54,14$$

## 5. Conclusiones

A continuación se presenta un resumen de todas las conclusiones clave de todos los proyectos realizadas a lo largo del curso.

### 5.1. Proyecto 1

Se realizó machine learning con la metodología de *Gradient Boosted Tree*, *Decision Tree* y *Linear Regression* y aquel con mejores resultados fue *Linear Regression* obteniendo las métricas reportadas anteriormente. Por lo que para la comparación entre tener medidas topográficas en el data set y no tenerlas se utilizó el modelo de *Linear Regression*.

### 5.2. Proyecto 2

Tomando en cuenta los resultados obtenidos en el segundo proyecto, donde se utilizó streaming para la predicción utilizando machine learning, se concluyó que utilizar streaming es provechoso pues si bien existe un margen de error con respecto a tener el conjunto de datos completos, dicho margen es mínimo a comparación de la ventaja de poder trabajar con menos datos, pudiendo tomar decisiones de forma más ágil, dando así una alta capacidad de adaptabilidad.

### 5.3. Proyecto 3

Con respecto a la utilización de medidas topológicas del grafo asociado con el data set, se puede concluir que el uso de las mismas contribuye a una mejor predicción para el machine learning, pues tal y como se vio en los resultados, el error es casi nulo, por lo que al añadir al data set propiedades del grafo efectivamente contribuye a la obtención de las predicciones.

## Referencias

- [1] B. Heaberlin and S. DeDeo, “The Evolution of Wikipedia’s Norm Network.” Future Internet 8(2), 14 (2016)
- [2] “Index of Complex Networks”, Index of Complex Networks. [Online]. Available: <https://icon.colorado.edu/networks>.
- [3] “Chapter 6. Algorithms - The Neo4j Graph Data Science Library Manual v1.4”, Neo4j.com. [Online]. Available: <https://neo4j.com/docs/graph-data-science/current/algorithms/>.