TRABAJO 3: ANÁLISIS COVID-19

Kevin Arley Parra Henao
código: 201710093010
kaparrah@eafit.edu.co

Agustín Nieto García
código: 201710009010
anietog1@eafit.edu.co

Introducción

El propósito de este trabajo es aplicar los conocimiento adquiridos en el curso de tópicos especiales en telemática, para realizar un ejercicio que consiste en un análisis exploratorio de datos con información sobre la actual contingencia del covid-19, concretamente lo que se refiere a los casos infectados. Este trabajo analizará la información de algunos datasets con datos relacionados con el covid-19, provistos por organizaciones como la organización mundial de la salud (WHO, World Health Organization).

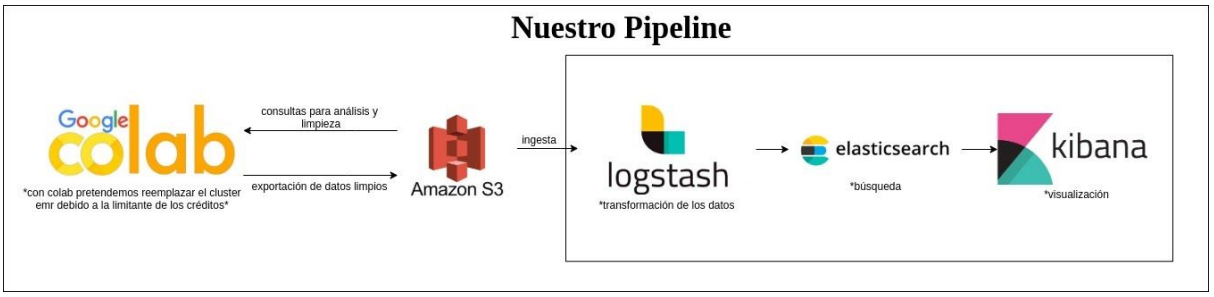
WHO-datasets: https://data.humdata.org/dataset/coronavirus-covid-19-cases-and-deaths

El código y los datasets en específico usados para este trabajo se pueden encontrar en el repositorio: https://github.com/anietog1/telem-s7/tree/trabajo3

Diseño

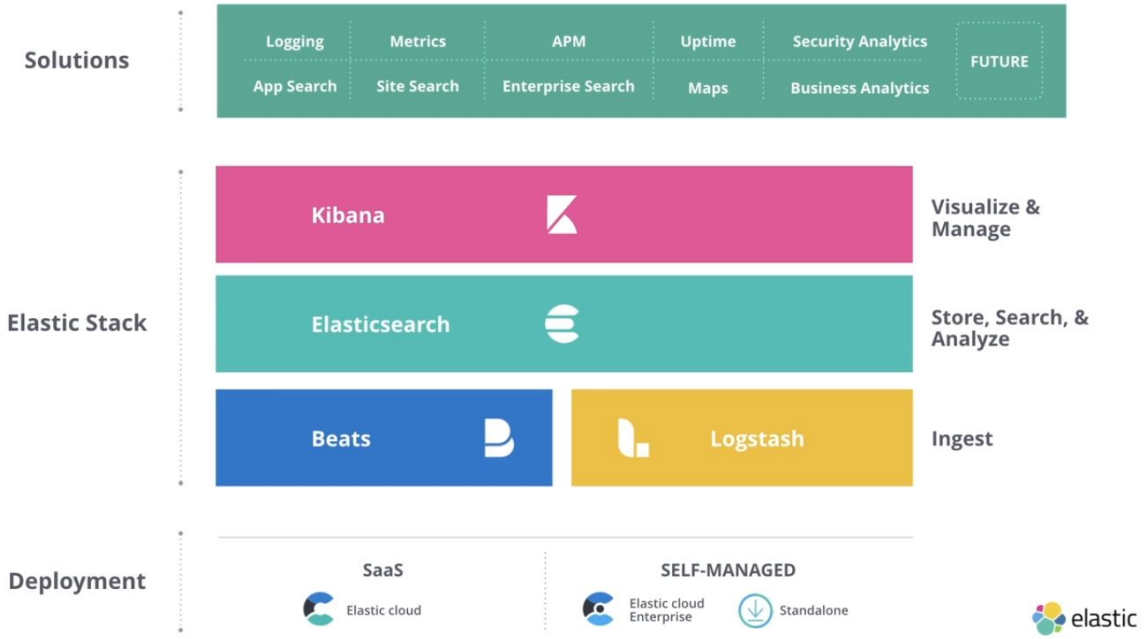
En este ejercicio se realizará un procedimiento de Big data analytics, cubriendo el ciclo de vida analitico de datos, que van desde la ingesta, en la cual la fuente de datos serán los datasets enunciados anteriormente; Se tendrá también el procesamiento y exploración de datos, con pyspark en el entorno de colab. Por su parte el almacenamiento se tendrá en s3 y finalmente para la visualización se utilizará principalmente Kibana, pero para poder usar esta herramienta requerimos de logstash y elasticsearch, lo que vendría conformando el Stack ELK, que será utilizado con una cuenta free trial de 14 días en www.elastic.co, que ofrece la creación de clusters automáticamente y la ejecución de los servidores de elasticsearch y kibana en nube.

A continuación se ilustra de manera general el flujo y servicios utilizados:



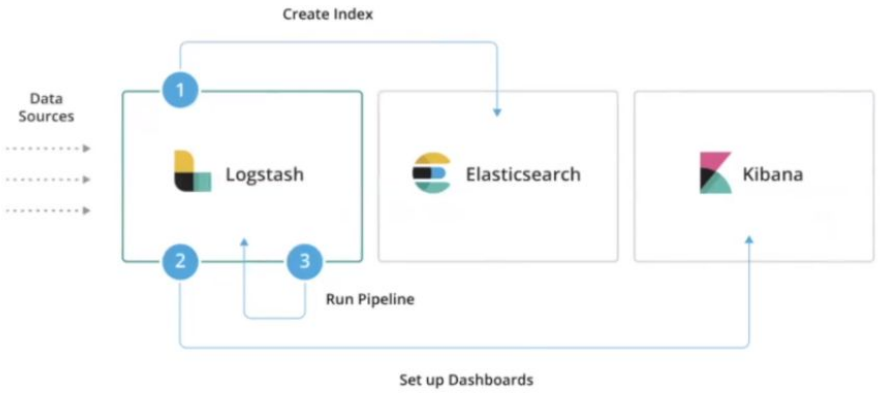
Elastic Stack:

El servicio proveído por elastic es robusto y permite la implementación de diferentes soluciones relacionadas con manejo de grandes volúmenes de información, tanto estructurada como no estructurada, como lo puede ser Business Analytics, APM y Security Analytics.



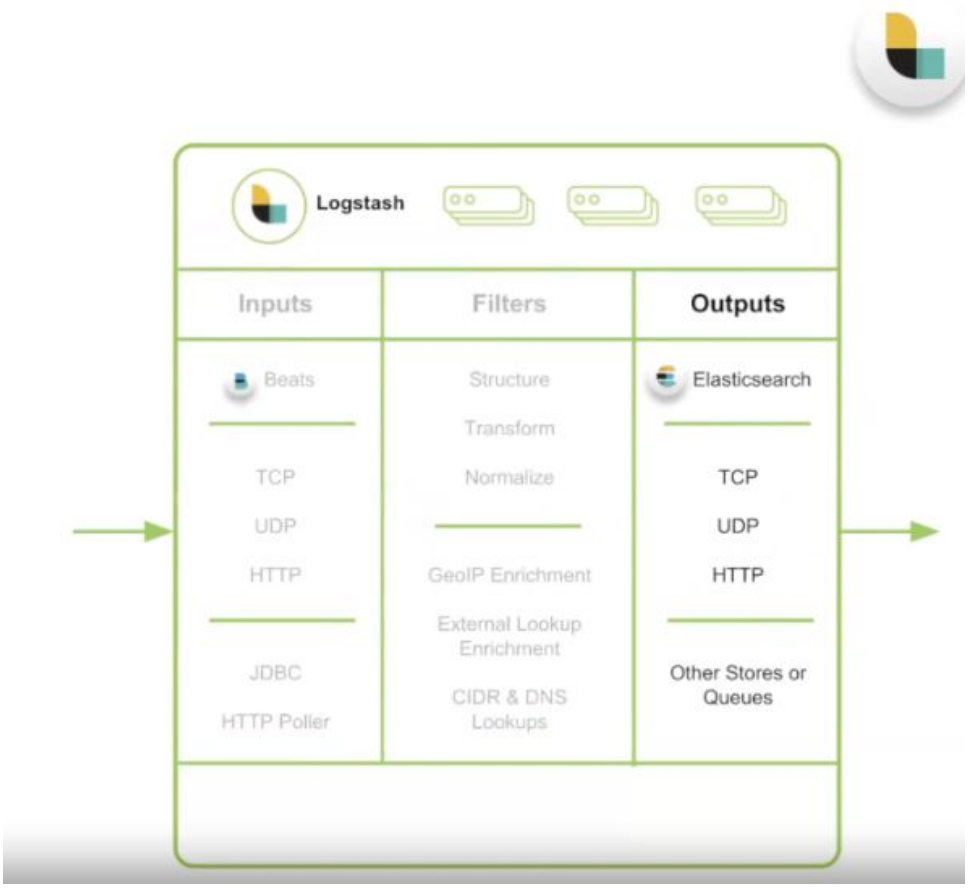
Se observa que con este cluster de servicios se tienen cubiertas todas las etapas del ciclo de vida analito de datos.

De manera general, la implementación que se tendrá en elastic stack incluye la ingesta desde una fuente de datos (Amazon S3 en este caso), la creación del index respectivo para cada documento creado en elasticsearch, la ejecución del pipeline de logstash, que se explica más adelante y finalmente la definición de las diferentes vistas que se tienen en kibana junto con los dashboards para agrupar estas vistas. Todo esto se ilustra a continuación.



Logstash:

Es un servicio para la ingesta de datos, el cual permite consumir datos de un amplia gama de fuentes de datos, y en un amplio espectro de formatos como lo pueden ser formatos no estructurados como imágenes, vídeo, documentos; datos semi-estructurados como JSON, y datos estructurados como CSV o bases de datos SQL. En logstash se definen un pipeline que contiene la entrada, filtrado y salida de datos. En este proyecto, utilizaremos principalmente la entrada para consumir desde s3 los datasets procesados por pyspark, y la salida para enviar los datos hacia elasticsearch y posteriormente visualizarlos en kibana. Aunque el procesamiento se puede hacer en logstash o elasticsearch, dados los requerimientos del ejercicio académico, usamos pyspark para realizar esta etapa.

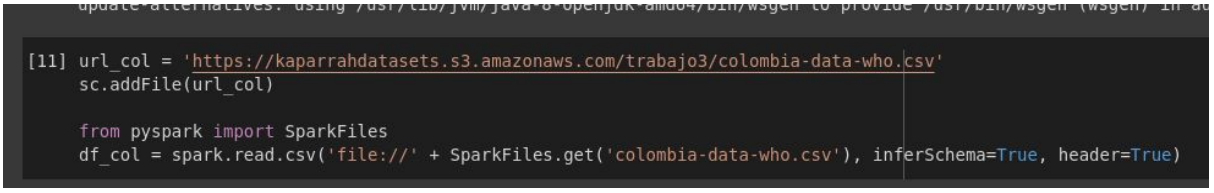


Procedimiento

Exploración de datasets:

Se tendrán dos datasets, uno con los datos a nivel mundial, del cual podremos hacer una comparación entre Colombia y el resto del mundo. Adicionalmente se tiene otro datasets que tiene más en detalle los datos de Colombia, incluye edad y sexo por ejemplo. Una alternativa que encontramos para explorar los datasets y evitar usar recursos de las cuenta de AWS educate, que ya estaban escasos, fue utilizar notebooks de colab, que es un servicio gratuito de google y permite tener entornos de ejecución. Lo primero que se hizo fue descargar los datasets y subirlos al repositorio. Es importante aclararle al lector que la información que se tomó en los datasets van hasta el día 11 de mayo, último día en que se actualizaron los datos. Por tanto el resultado aquí presentado no constituye un análisis de la situación real al momento de publicar este trabajo.

Realizamos la lectura de los datasets desde s3:



Hacemos un análisis exploratorio de los datos globales y limpiamos para sacar las columnas que nos interesan únicamente:

```
[6] # https://kaparrahdatasets.s3.amazonaws.com/trabajo3/colombia-data-who.csv
    df_global = spark.read.csv('/content/telem-s7/datasets/global-data-who.csv', inferSchema=True, header=True)

[7] df_global.show(10)
```

```
+--------+----------+----------+-----------+-------------------+-------+-------+--------+--------+-------------+-----------
|OBJECTID|ISO_2_CODE|ISO_3_CODE|  ADM0_NAME|        date_epicrv|NewCase|CumCase|NewDeath|CumDeath|Short_Name_ZH|Short_Name_
+--------+----------+----------+-----------+-------------------+-------+-------+--------+--------+-------------+-----------
|       1|        AF|       AFG|Afghanistan|2020-02-24 00:00:00|      1|      1|       0|       0|       阿富汗|  Afghanist
|       2|        AF|       AFG|Afghanistan|2020-02-25 00:00:00|      0|      1|       0|       0|       阿富汗|  Afghanist
|       3|        AF|       AFG|Afghanistan|2020-02-26 00:00:00|      0|      1|       0|       0|       阿富汗|  Afghanist
|       4|        AF|       AFG|Afghanistan|2020-02-27 00:00:00|      0|      1|       0|       0|       阿富汗|  Afghanist
|       5|        AF|       AFG|Afghanistan|2020-02-28 00:00:00|      0|      1|       0|       0|       阿富汗|  Afghanist
|       6|        AF|       AFG|Afghanistan|2020-02-29 00:00:00|      0|      1|       0|       0|       阿富汗|  Afghanist
|       7|        AF|       AFG|Afghanistan|2020-03-01 00:00:00|      0|      1|       0|       0|       阿富汗|  Afghanist
|       8|        AF|       AFG|Afghanistan|2020-03-02 00:00:00|      0|      1|       0|       0|       阿富汗|  Afghanist
|       9|        AF|       AFG|Afghanistan|2020-03-03 00:00:00|      0|      1|       0|       0|       阿富汗|  Afghanist
|      10|        AF|       AFG|Afghanistan|2020-03-04 00:00:00|      0|      1|       0|       0|       阿富汗|  Afghanist
+--------+----------+----------+-----------+-------------------+-------+-------+--------+--------+-------------+-----------
only showing top 10 rows
```

```
[10] df_global.where(df_global['ADM0_NAME'] == 'Colombia').show(10)
```

```
+--------+----------+----------+---------+-------------------+-------+-------+--------+--------+-------------+------------
|OBJECTID|ISO_2_CODE|ISO_3_CODE|ADM0_NAME|        date_epicrv|NewCase|CumCase|NewDeath|CumDeath|Short_Name_ZH|Short_Name_F
+--------+----------+----------+---------+-------------------+-------+-------+--------+--------+-------------+------------
|    2862|        CO|       COL| Colombia|2020-03-06 00:00:00|      1|      1|       0|       0|     哥伦比亚|     Colombie
|    2863|        CO|       COL| Colombia|2020-03-07 00:00:00|      0|      1|       0|       0|     哥伦比亚|     Colombie
|    2864|        CO|       COL| Colombia|2020-03-08 00:00:00|      0|      1|       0|       0|     哥伦比亚|     Colombie
|    2865|        CO|       COL| Colombia|2020-03-09 00:00:00|      0|      1|       0|       0|     哥伦比亚|     Colombie
|    2866|        CO|       COL| Colombia|2020-03-10 00:00:00|      2|      3|       0|       0|     哥伦比亚|     Colombie
|    2867|        CO|       COL| Colombia|2020-03-11 00:00:00|      0|      3|       0|       0|     哥伦比亚|     Colombie
|    2868|        CO|       COL| Colombia|2020-03-12 00:00:00|      6|      9|       0|       0|     哥伦比亚|     Colombie
|    2869|        CO|       COL| Colombia|2020-03-13 00:00:00|      7|     16|       0|       0|     哥伦比亚|     Colombie
|    2870|        CO|       COL| Colombia|2020-03-14 00:00:00|      8|     24|       0|       0|     哥伦比亚|     Colombie
|    2871|        CO|       COL| Colombia|2020-03-15 00:00:00|      0|     24|       0|       0|     哥伦比亚|     Colombie
+--------+----------+----------+---------+-------------------+-------+-------+--------+--------+-------------+------------
only showing top 10 rows
```

Hacemos drop de todo lo que no nos interesa:

```
df_global.drop('OBJECTID', 'ISO_2_CODE', 'ADM0_NAME', 'Short_Name_ZH', 'Short_Name_FR', 'Short_Name_RU', 'Short_Name_AR')
```

Y exportamos a un nuevo archivo, con la intención de optimizar en Kibana:

```
df_xd.write.csv('global-condensado-header.csv', header=True)
```

Ese archivo posteriormente lo subimos a s3. Hacemos un análisis del dataset de Colombia y limpiamos, también. Sabemos que contiene una columna que no necesitamos que es de metadatos, entonces la removemos:

```
[11] df_tmp = spark.read.csv('telem-s7/datasets/colombia/covid19_colombia.csv', inferSchema=True, header=True)

[12] df_tmp = df_tmp.where(df_tmp['ID de caso'] != '#meta+id')
     df_tmp.describe()

     DataFrame[summary: string, ID de caso: string, Fecha de notificación: string, Codigo DIVIPOLA: string, Ciudad de ubicación:
```

Revisamos algunos datos:

```
[19] df_tmp.agg({'Fecha de notificación' : 'max'}).show()

+--------------------------+
|max(Fecha de notificación)|
+--------------------------+
|      2020-04-27T00:00:...|
+--------------------------+

[20] df_tmp.agg({'Fecha de notificación' : 'min'}).show()

+--------------------------+
|min(Fecha de notificación)|
+--------------------------+
|      2020-03-02T00:00:...|
+--------------------------+

[21] df_tmp.groupBy('Atención').count().show()

+------------+-----+
|    Atención|count|
+------------+-----+
|Hospital UCI|  117|
|   Fallecido|  253|
|        Casa| 3732|
|    Hospital|  278|
|        casa|    7|
|  Recuperado| 1210|
+------------+-----+
```

Por ejemplo, en atención existe Casa y casa, lo cual hay que corregir. Lo mismo con los géneros:

```
[22] df_tmp.groupBy('Sexo').count().show()

+----+-----+
|Sexo|count|
+----+-----+
|   F| 2677|
|   m|    1|
|   f|    2|
|   M| 2917|
+----+-----+
```

Realizamos limpieza. La consulta final no retornó registros debido a que fue limpiado.

```
[62] from pyspark.sql.functions import upper
     df_tmp = df_tmp.withColumn('Sexo', upper(df_tmp['Sexo']))

[63] df_tmp.where(df_tmp['Sexo'] == 'm').show() # nice!
```

Los estados los ordenamos. Había un “Leve” y un “leve”

```
[66] from pyspark.sql.functions import initcap # NICE!!
     df_tmp.withColumn('Estado', initcap(df_tmp['Estado'])).groupBy('Estado').count().show()

+---------+-----+
|   Estado|count|
+---------+-----+
|Fallecido|  253|
| Moderado|  309|
|     Leve| 4918|
|    Grave|  117|
+---------+-----+
```

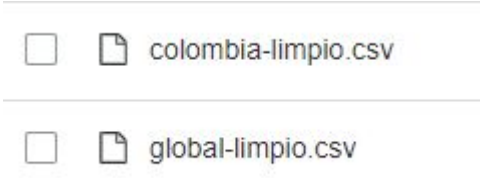
Limpiamos aún más y exportamos:

```
[88] df_tmp.where(df_tmp['Atención'] == 'casa').count() # retornaba 7
     df_tmp = df_tmp.withColumn('Atención', initcap(df_tmp['Atención']))

[91] df_tmp.write.csv('colombia-limpio-with-header.csv', header=True)

     df_tmp.groupBy('Atención').count().show()

+------------+-----+
|    Atención|count|
+------------+-----+
|   Fallecido|  253|
|        Casa| 3739|
|    Hospital|  278|
|Hospital Uci|  117|
|  Recuperado| 1210|
+------------+-----+
```

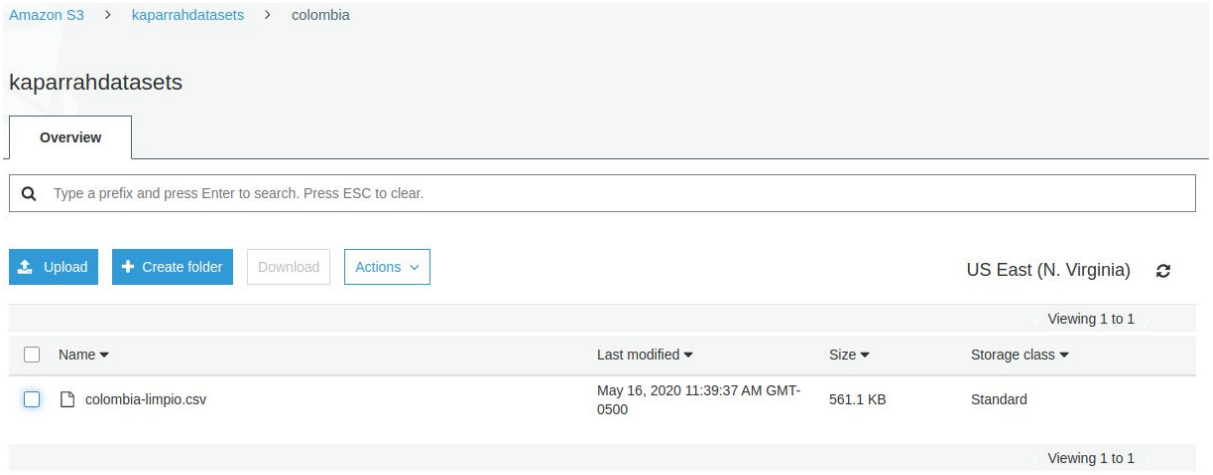
Esos archivos posteriormente se suben al s3 para ser consumidos por Kibana.

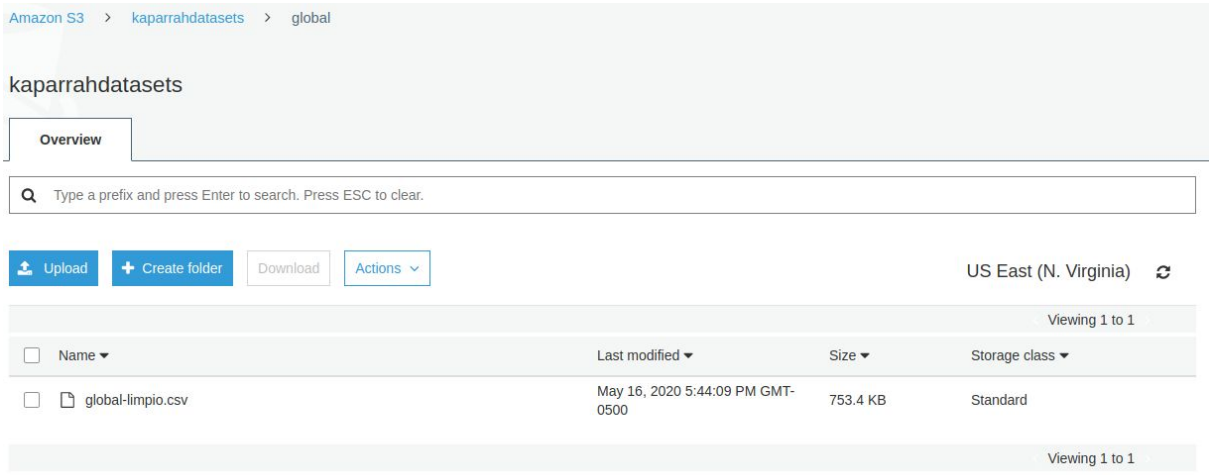
Los subimos a s3 con el uso de boto3:

```
[95] !pip install boto3 -q

[98] import boto3

     s3r = boto3.resource('s3', aws_access_key_id='ASIAQRGERHXB6FNZFG4Q',
                        aws_secret_access_key='QBtk5F/uo4YmGDGiAVSmTwAE+MZCqdGgKZ/Wofk9',
                        aws_session_token='FwoGZXIvYXdzEGAaDPPadI/sdQzQTz200SLEAZUn9VcQqvvZe5CjEwClY1WJ+iit8XzI4cc
     buck = s3r.Bucket('kaparrahdatasets')
     buck.upload_file('colombia-limpio-with-header.csv/part-00000-2c9df75a-3008-4317-a15f-5eaebf4e4d3a-c000.csv', 'c
     buck.upload_file('global-condensado-header.csv/part-00000-00dbd79a-b850-41ac-a8b4-6f0ec5bcc8c6-c000.csv', 'glob

[ ]
```



Para poder realizar la visualización, requerimos tener los datos en un almacenamiento que nos permita acceso con las diferentes herramientas que vamos a utilizar. Es por esto que utilizaremos el servicio Amazon S3, para almacenar los datasets. Cargamos los datasets a S3, global lo ponemos en la carpeta global y los datos de colombia en la carpeta colombia:





Nos aseguramos que sean públicos para el fácil acceso.

Logstash:

Procedemos a cargar los datos utilizando un cliente de logstash local. Se realiza la configuración del pipeline en un archivo de configuración:

Para los datos globales:

global.config

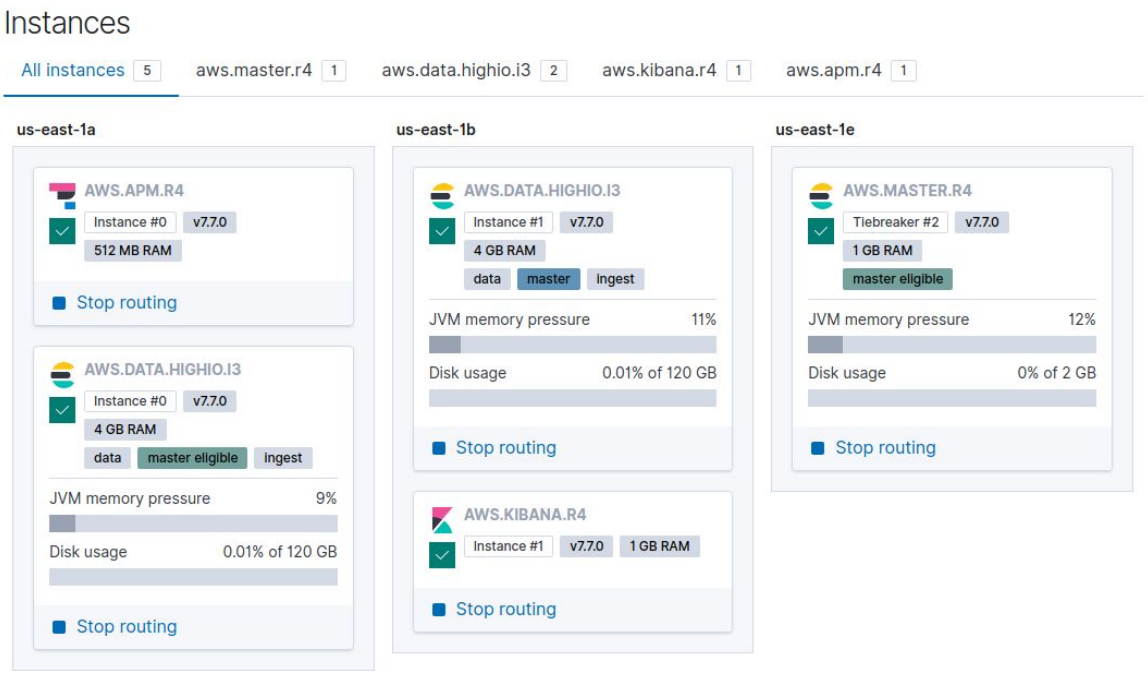
```
input {
    s3 {
        bucket => "kaparrahdatasets"
        prefix => "global/"
        region => "us-east-1"
    }
}
filter {
    csv {
        separator => ","
        columns => ["ISO_3_CODE", "date_epicrv","NewCase","CumCase","NewDeath","CumDeath","Short_Name_ES"]
        skip_header => true
    }

    mutate { convert => ["NewCase", "integer"]}
    mutate { convert => ["CumCase", "integer"]}
    mutate { convert => ["NewDeath", "integer"]}
    mutate { convert => ["CumDeath", "integer"]}
}

output {
    elasticsearch {
        hosts => "https://c43860fdea8c492e8aa18c6f917cd997.us-east-1.aws.found.io:9243"
        index => "covid-global-data"
        user => "elastic"
        password => "OFgLZvR8LkKDTAuU8PgNPLsx"
    }
    stdout {}
}
```

colombia.config

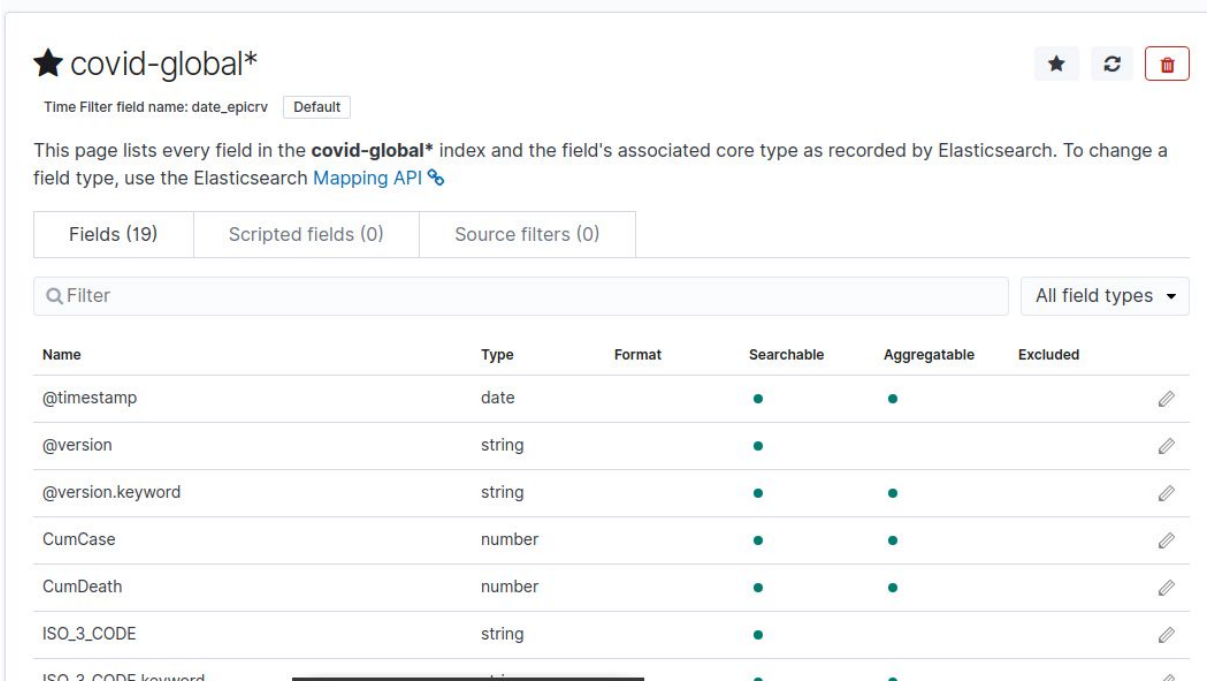
```
input {
    s3 {
        bucket => "kaparrahdatasets"
        prefix => "colombia/"
        region => "us-east-1"
    }
}
filter {
    csv {
        separator => ","
        columns => ["Ciudad de ubicación","Departamento o Distrito","  Atención","Edad","
        skip_header => true
    }


}

output {
    elasticsearch {
        hosts => "https://c43860fdea8c492e8aa18c6f917cd997.us-east-1.aws.found.io:9243"
        index => "covid-colombia-data"
        user => "elastic"
        password => "OFgLZvR8LkKDTAuU8PgNPLsx"
    }
    stdout {}
}
```

Elastic Stack:

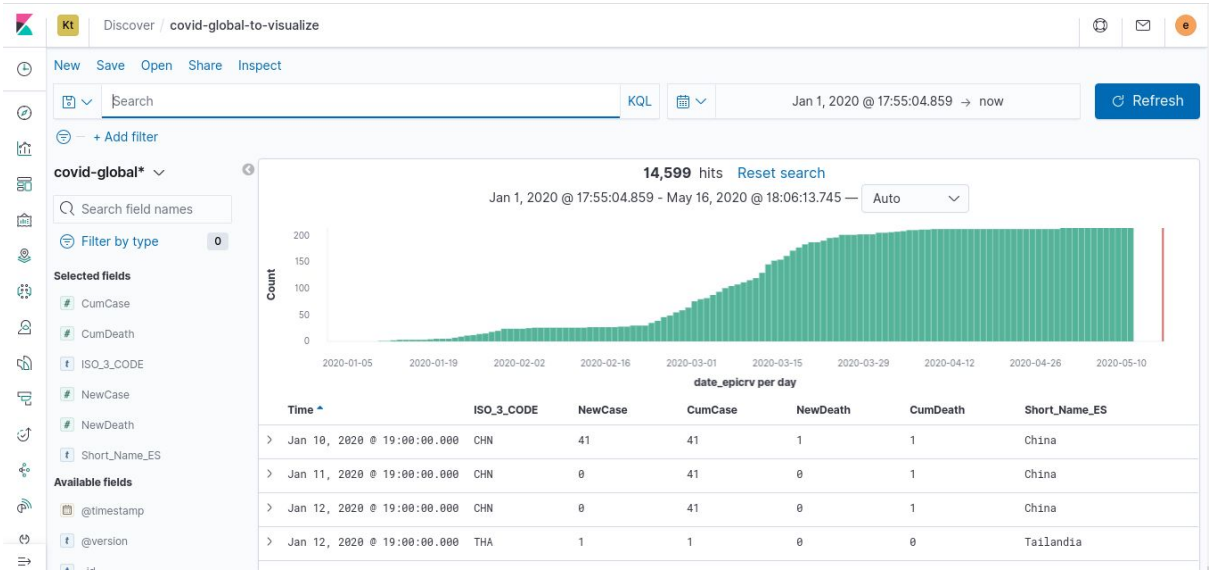
Se crea el deployment en la interfaz de elastic y se genera el cluster que utilizaremos para almacenar y visualizar los datos. Este cluster se ejecuta en aws:



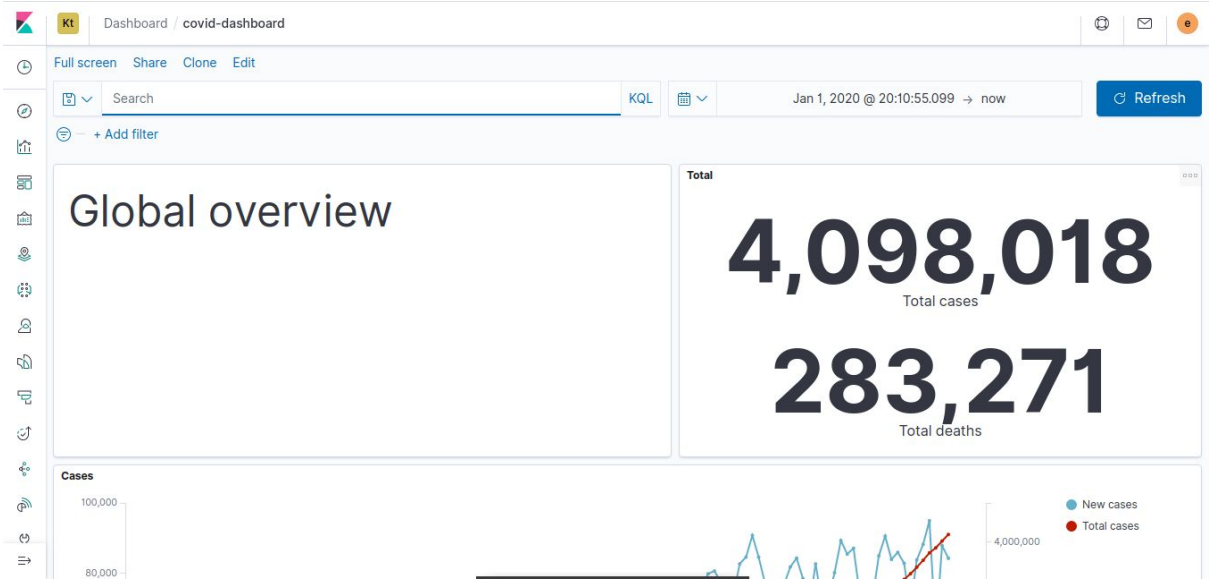
Una vez cargados, debemos crear el index para poder consultar los datos desde kibana:



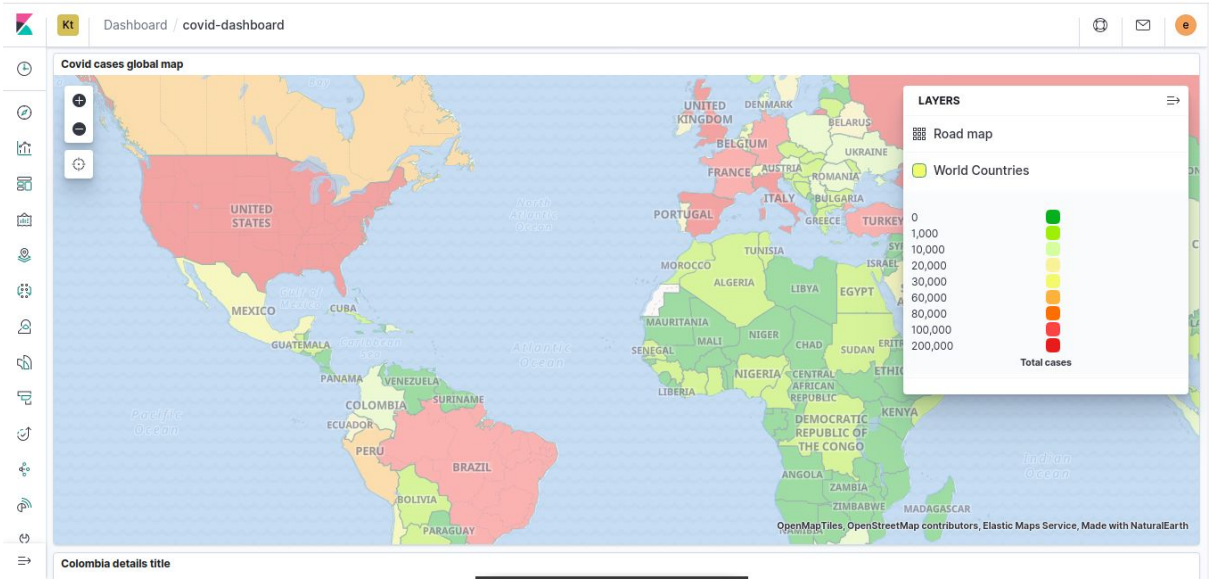
Una vez realizado este paso, podremos explorar los datos y seleccionar los campos que queremos usar para las visualizaciones:



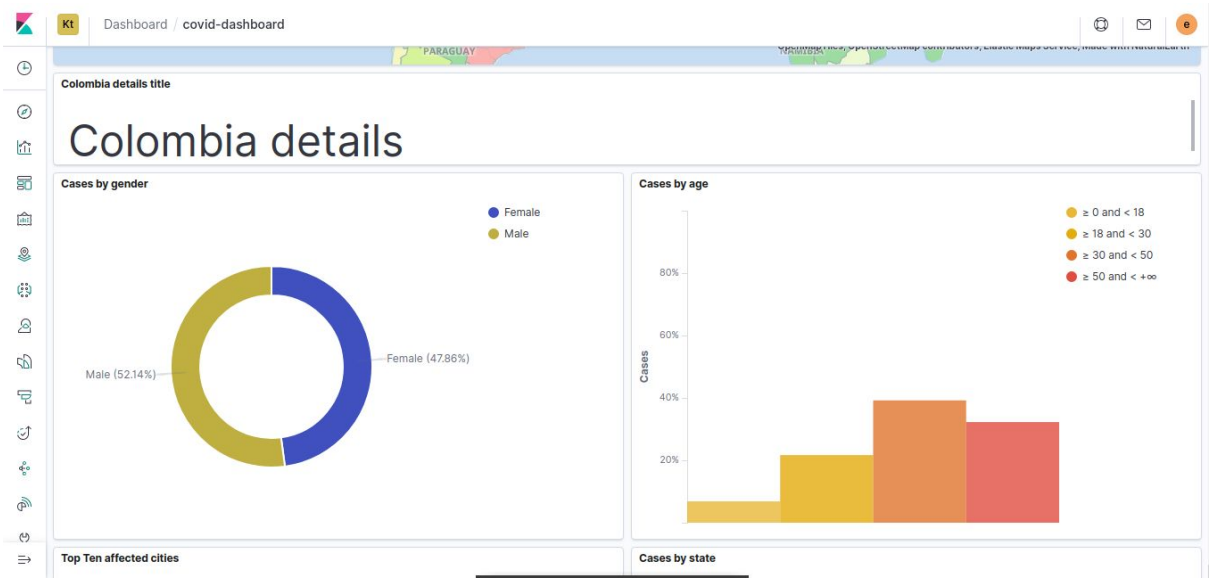
Se realizan primero las gráficas a nivel mundial de los casos de covid y de las muertes causada por esta enfermedad, luego se presentan estos datos a nivel de colombia y finalmente se comparan en dos gráficos de barras.



También se presenta un mapa que permite fácil comparación de los casos entre países, con solo ubicar el mouse sobre cada país se puede tener información de los casos y las muertes por covid en ese territorio. Adicionalmente se presenta clasificación en escala de colores de acuerdo a la cantidad de casos que se tengan de covid.



Por último, con ayuda del otro dataset, se realiza un análisis de la situación de Colombia en detalle, ya que este datasets tiene otra información que nos permite tener mayor información de los casos, como su edad y el sexo al que pertenecen.



Nota: Anexo a este trabajo se presenta el reporte completo en pdf que se realizó con Kibana. También lo encuentra en: https://github.com/anietog1/telem-s7/tree/trabajo3/reports

Conclusiones

A partir de los datos se puede concluir que la situación de Colombia con respecto a la pandemia por el Covid-19 no es tan dramática, pues no encontramos lejos de países como Estados Unidos (más de un millón de casos) o Brasil (más de cien mil casos); Adicionalmente vemos que los casos en Colombia empezaron mucho después que en otros países, teniendo reportes desde el 7 de enero en China y en Colombia apenas el primer reporte que se tiene en el datasets fue el 5 de marzo del 2020. Todo esto haría pensar que el país va por buen camino, y los datos podrían ser engañosos, por lo que los expertos recomiendan no bajar la guardia, pues si bien hay pocos casos y el crecimiento parece no ser tan acelerado, se debe tener en cuenta que los reportes pueden tener cierto delay desde el momento en que se toma la muestra hasta que se obtienen los resultados positivos, por lo que siempre estamos mirando una imagen de una o dos semanas atrás.

Por otro lado, vemos que los datos que se están reportando varían con respecto a la organización que hace el reporte, pues cada entidad define su formato para los datos, por lo que se debe hacer un proceso adicional de transformación para poder comparar la información. Adicionalmente, los datasets presentan información diferente entre sí, en cuanto a campos y cuanto a valores en algunos casos. Esto evidencia la dificultad que se tiene en estos casos para saber una cifra exacta que exprese la situación real.

Por último, respecto a las herramientas utilizadas para elaborar este trabajo, se puede decir que ofrecen robustez y calidad del servicio, con facilidad de uso y configuración. Adicionalmente se ofrecen diversos features para graficar que están bastante completos. Otro ventaja de estas herramientas, en el caso concreto de kibana, es que se puede tener

online con el servicio provisto por elastic. Sin embargo, comparándola con otras alternativas como Tableau, vemos que se queda corta en ciertas características y en variedad de diagramas, además la facilidad de uso de métricas y dimensiones es ciertamente más concreta en este último.