CSDA1020

Big Data Analytics Tools

Project 3, ELK (Elasticsearch, Logstash, Kibana)

Ioana Moca

**Problem Background**

You are the data analyst at a large investment firm that is contemplating to invest in a used car business. Your task is to install and configure ELK stack in your GCP platform. Successful completion includes configuring Logstash\_cars.config file and firing Logstash to ingest the cars.csv data into Elasticsearch and use Kibana to analyze and visualize the results. Project 3 grade is Pass or Fail based on the above. Bonus marks for Project 3 involves using Kibana completing the questions below.

**Solution Part 1: Running Elastic Search, Kibana and Logstash**

Following the configuration of the cars logstash file, the following codes in SSH successfully open Kibana with the cars.csv dataset loaded as the cars\* index.

**Note: Please see last page for Part 2 of the solution, visualizations**.

First we increase virtual memory:

sudo sysctl -w vm.max\_map\_count=262144

Next we run elasticsearch from an SSH browser window:

elasticsearch/bin/elasticsearch

We run the logstash file to ingest the cars dataset:

logstash/bin/logstash -f /home/imoca\_econ/logstash\_cars.config

We run Kibana:

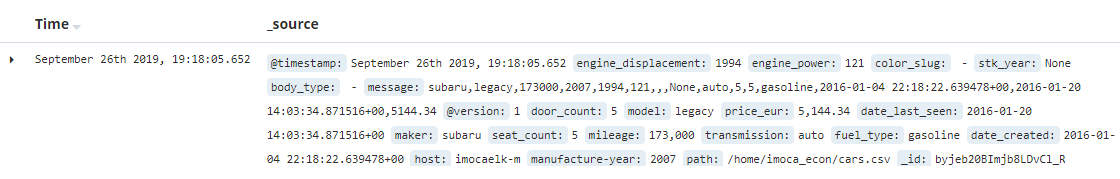
kibana/bin/kibana

**Bonus Analytical Questions**

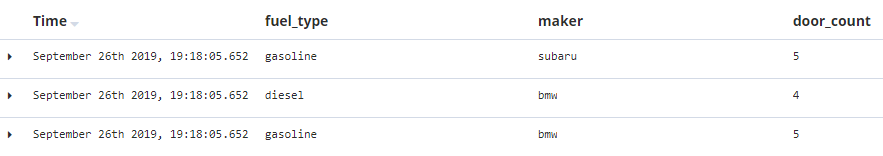
After ingesting cars data set into ES using Logstash, use Kibana to answer the following questions

1. **Read and display all the documents in the cars index**

To see our logs, we can select the “Discover” option in the left bar menu. This displays documents. A sample is shown below.



We can also filter to generate more streamlined, tabular information by document. Below, I have filtered by fuel type, maker and door count. We see that only those variables are displayed for each document.

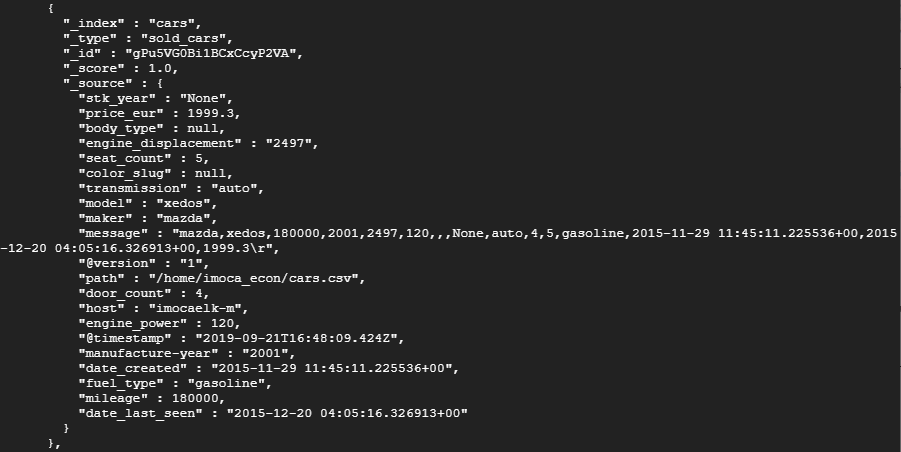


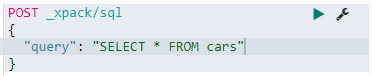
1. **Query the cars index to display all documents**

There are two ways to do this. One is using the following code and querying elastic search as follows:

curl -XGET http:// http://34.74.220.4:9200/cars\*/\_search?pretty=true&q=\*:\*

We get the following output. Below is a sample of one of the documents displayed.



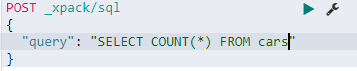
 We can also use the Dev Tools option. The following sql code displays all documents:

1. **Display the total documents being indexed**

As above we can query Elastic Search with the following code, and get the total documents being indexed:



In Kibana Dev Tools, we can execute the same command as follows:



It returns the same number of rows as Elastic Search:



**Solution Part II: Visualizations**

The next two pages contain a sample dashboard made with the cars index. It includes:

* A pie chart showing the top ten makers with respect to the count of cars the dataset in descending order, with BMW making up 12% and Mercedez benz making up 11%, respectively.
* A bar chart illustrating price percentile by maker
* A data table illustrating the count of cars by manufacture year for the top manufacture years
* A heat map illustrating average price by maker and fuel type

