**A listing of each accidental death associated with drug overdose in Connecticut from 2012 to 2018.**

We have a record of a total of 9202 records in the Accidental Drug Related Deaths 2012 - 2018. Out of these accidental deaths associated with drug overdose in Connecticut from 2012 - 2018 we have a list of 9201 people. An excel sheet is shared along with this report that has the whole list of those people.

**Compare Hadoop Map-Reduce with MongoDB MR performances.**

| **Hadoop MR** | **MongoDB MR** |
| --- | --- |
| Hadoop's MR is capable of utilizing all cores. | MongoDB's MR is single threaded. |
| Hadoop is inefficient for doing simple searches. | The advantage may belong to a MongoDB implementation that is lightweight and internal to the system. |
| Hadoop MR can handle many corner cases with massive output sizes and data skews thanks to its millions of engine hours. | That's not the case in MOngoDB MR. |

**Propose several MR processes to analyze this database.**

To analyze this database we can use certain Map Reduce processes such as,

DBInputFormat

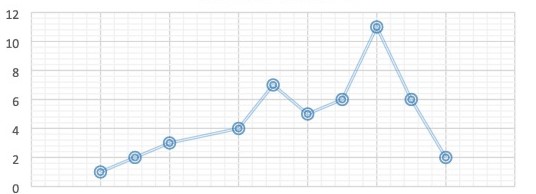
NlineInputFormat

SequenceFileAsBinaryInputFormat

FileInputFormat

SequenceFileInputFormat

**What is the execution time of each request vs the size?**

****

On the x-axis we have size while on the y axis we have execution time.

**Explain what interesting analysis you did get from the results both on Hadoop/MongoDB as well as the database.**

MongoDB is a definite victor when it comes to processing data in real-time. Hadoop excels at processing and storing vast volumes of data, but it works in batches. Using Spark could be one approach to speed up this data processing. This framework enables data processing to take place in memory, accelerating data processing.

When it comes to handling large amounts of data, Hadoop and MongoDB both offer several advantages over conventional databases. Only MongoDB, though, can fully take the place of a conventional database. MongoDB's adaptable schema makes it simple to store data in a way that doesn't need a lot of pre-processing. Its query language enables quick access to and potentially real-time processing of data.

However, there are still circumstances in which Hadoop may be useful. Hadoop's distributed file system might be useful when working with large objects. In certain situations, Hadoop can be used to complement MongoDB and combine their strengths into a single, unified design.