UCLA Extension – Introduction to Data Science

COM SCI X 450.1

Name: \_\_\_\_\_\_\_\_\_\_

Instructor: \_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_

[Final Project Title]

What are the differences and similarities between K-Means and K-NN? What are the differences and similarities between MongoDB and MySQL?

What are the differences and similarities between K-Means and K-NN? What are the differences and similarities between MongoDB and MySQL?

*Introduction*:

Machine learning is broken down into two types of algorithms, supervised and unsupervised. The defining factor if one were to simplify in what makes supervised learning is the data set has labels or outputs whereas unsupervised has no labels or outputs. We’ll inspect K-NN a supervised algorithm and K-Means an unsupervised algorithm.

Database (SQL) stores have been around for quite a while compared to newer database technologies (NoSQL) stores. The previous works well with transactional data and grew in popularity to create a variety of options from open source (MySQL) to production/licensed technologies (i.e. Oracle), the latter NoSQL space has no shortage of technologies although there are a few more prominent choices. We’ll analyze MongoDB, a document store and compare it to MySQL.

*K-means compared to K-NN*:

To start out with similarities, both K-means and K-NN leverage a Euclidean distance measure (which is quite common in optimizations looking to reduce ‘distances’ between errors, expected values, etc.). They both have a concept of setting an initial parameter, a K. For the most part those are the greatest similarities.

To move on to differences, K-means is an unsupervised learning algorithm whereas K-NN is supervised. As we mentioned earlier the previous algorithm knows only of the inputs whereas the latter algorithm knows about the inputs and outputs. The latter allows K-NN to perform predictive qualities and the K determines the number of neighbors able to vote, or contribute to the determination of the output based on some input. K in K-means is the number of centroid to be defined. K-mean clusters various inputs whereas K-NN predicts classifications based on inputs. Finally in terms of applications K-means is useful when there is no output data so for things such as market segmentation recommendation (i.e. knowing which markets to target with which products and or services), an application for K-NN can be used for image classification (i.e. search engine image categorization).

*MongoDB compare to MySQL*:

MongoDB is a NoSQL solution that is a document store, it is used to store any type of JSON in its native format BSON (which isn’t far from JSON). Legal documents can be stored for example. MySQL is a SQL based solution and a relational database. Transactions can be stored for example, or customer information in a relational data store. Both MongoDB and MySQL are open-source and both have the concept of a database, a table in MySQL can be thought of equivalent to a collection in MongoDB, a row in MySQL can be thought of equivalent to a document in MongoDB and there are some aggregation and join functionalities in MongoDB although much more limited than MySQL

*Visualization*

**Figure 1:** MongoDB versus MySQL

|  |  |  |
| --- | --- | --- |
|  | **MongoDB** | **MySQL** |
| ACID Transactions (supported)? | Yes | Yes |
| Best used for? | Storing document type data | Storing transactional data |
| Open Source | Yes | Yes |
| Language | Query Document Language | SQL |