*Summary of the paper “When is Nearest Neighbour Meaningful“ by Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan and Uri Shaft*

In this paper the authors analyzed how dimensionality influences the outcome and the effectiveness of the nearest neighbour problem. The main hypothesis of the paper asserts, that the higher the dimensionality gets, the less significant the difference between the two distances, from the query point to its nearest data point and from the query point to any other data point, becomes.

A possible definition of the Nearest Neighbour problem would be the following: disposing of a collection of data in an m-metric space, with a specific query point being of interest, find the one data point which is nearest to the specified query point.

One of the comprehensions of this paper claims that the technique of linear scan is much more effective than the techniques proposed on the presented workloads and reveals a better performance, even in case of relatively high dimensionality (10-15).

As specified in the upper part of the summary, there exists a case, under certain conditions, with increasing dimensionality, in which the distance to the nearest neighbour and the distance to the farthest neighbour lose contrast, until they become indistinguishable. In such cases the nearest neighbour heuristic is rather not recommended.

An advancement of the nearest neighbour problem is represented by the k-nearest neighbour variant, where the observer does not only take interest in the finding of a single nearest data point, but is rather in search of the first k data points, which are nearest.

The paper tries to provide an adequate answer to the question in the title by showing the empirical results of various simulated workloads and also of some real data.

The authors suggest that, before one uses the nearest neighbour algorithm, they should make sure, especially when dealing with higher dimensionality, that there exists a clear dissociation between the nearest neighbour and the one which is farthest away.

However the writers emphasize the fact that there are also cases in which the nearest neighbour can still be meaningful, even in high dimensions and that the results of the experiments should not be treated as generally affirmative, because there still exist some situations in which they can be proven wrong. Furthermore the authors encourage the choice of a meaningful workload when evaluating a nearest neighbour technique.

The authors also defines an unstable query for a given ε. We classify a query as being unstable, when the distance from the query point to most of the data points is less than (1+ ε) times the distance from the query point to its nearest neighbour. An unstable query usually results in a meaningless differentiation between the nearest neighbour and the rest of the data points.

The paper also includes some formulas and mathematical notations of the scenarios, when the nearest neighbour algorithm becomes meaningless and with the help of these formulas the hypothesis stating that the nearest neighbour approach becomes useless with increasing dimensionality, has been tested. Basically what the formulas are representing suggests that if the distances are distributed in a certain way and if the distributions behave in a certain way with the increasing of dimensionality, all query points are converging to the same distance from the query point.

Another part of the paper concentrates on answering the question: starting with which dimension does nearest neighbour become meaningless? The authors have analyzed a set of simulations in order to be able to give an adequate answer to this question.

Furthermore the paper presents some experiments and cases, where the nearest neighbour method was proven to be meaningless and meaningful, respectively. The conditions of these scenarios are discussed, as well as the conclusions. One of these conclusions states that nearest neighbour continues to be meaningful even with increasing dimensionality in some scenarios, for instance in the case of clustered data. In this case it is sufficient, that the nearest cluster to the query point is found. This comprehension is of much importance, because it shows, among other things, that there also exist scenarios, where nearest neighbour is still useful even in higher dimensions.

But other than this exceptional case, the authors come to the conclusion, that the nearest neighbour algorithm often becomes meaningless starting with as few as 10-20 dimensions.

Considering the k-nearest neighbour approach, the authors state that it’s quality can be determined by the number of data points which are within a factor of (for instance) 3 of the distance to the nearest neighbour: with the increasing of k, the percent of data points within the named area increases as well.

Another suggestion by the authors refers to the comparison between the method of linear scans and the nearest neighbour algorithm as a sanity check.

The process of the performance getting worse while dimensionality increases is also knows as “dimensionality curse”.

The authors also define the “boundary effect” as taking place, when the query region is mostly outside the hyper-cubic data space (visual representation of hyper-dimensionality). In such a scenario, where the boundary effect is not taken into consideration, the actual cost can be very different than the expected one and much higher.

As a conclusion of the paper the authors advise us to make sure that the distribution of the distances allows for sufficient contrast, so that the workloads would be meaningful. By that it is meant, that the differences between the distance to the nearest neighbour and the distance to the other data points should be clearly and significantly distinguishable. They furthermore suggest that the chosen nearest neighbour algorithm should present better performance than the banal sequential scan.