Naïve Bayes was compared with K Nearest Neighbors within the context of binary image classification while leveraging the pictures of fresh and rotten fruit. (Nayak, 2023) The goal was to identify possible pros and cons of using Naïve Bayes versus K Nearest Neighbors.

Performance metrics such as speed, memory and accuracy are often great metrics to use when deciding to deploy a solution. If the model is being deployed in a medical capacity erroneous result can be life threatening. In addition to preventing the blowback from delivering an incorrect result, a low error rate can result in less retraining and rework thus better performance. For a solution like autonomous driving a high-speed model can result in decisions being made in a timely manner that might prevent an accident. While a model that does not consume a lot of resources like memory can provide cost savings when processing large volumes of data.

Naïve Bayes leverages probability. Based on prior knowledge or training Naïve Bayes will use the conditions or attributes to deliver the probability of an outcome. It can be represented as:

* P(A|B) = (P(B|A) \* P(A)) / P(B)
* P(A|B) probability of event A given condition B
* P(B|A) probability of condition B given event A
* P(A) probability of event A
* P(B) probability of condition B

K Nearest Neighbors classifies based on similarities. When attempting to identify a fruit as fresh or rotten it will try and find the example that most closely resembles the item that needs to be classified. One way to re

The presumption based on a variety of articles is that Naïve Bayes will be faster and more accurate. A few notable quotes:

*Naive bayes is much faster than KNN due to KNN’s real-time execution* (Varghese, 2018)

*Naive Bayes … tends to be faster when applied to big data. In comparison, k-nn is usually slower for large amounts of data, because of the calculations required for each new step in the process Naive Bayes is highly accurate when applied to big data. Don’t discount K-NN when it comes to accuracy though* (Glen, 2019)

The data set *Fresh and Rotten Classification* (Nayak, 2023) was images of fruit that were divided into folders for fresh or rotten obtained from Kaggle.

The code employs scikit-learn’s GaussianNB for Naïve Bayes KNeighborsClassifier module for K Nearest Neighbors. Prior to running the data sets through the models the images were transformed into hitograms that essntiall compile vlaues based on the colors in the images. The training was wrapped in functions that capture the time just prior to training and the time immediately after triaing. Thow activities were wrapped in functions that started a memory trace to measure ram utilization during the models running and stopped immediately after the stotrasin stop time. The function records the lowest and highest ram utilization. The code the records the train start time, train stop time, memory high menoy low and accuracy in an excel workbook.

The result:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Classifier/Model** | **Train Start** | **Train End** | **Train time [h]:mm:ss.0** | **Memory Start (KB)** | **Memory Peak (KB)** | **Memory Difference (KB)** | **Accuracy** |
| Naive Bayes | 8/5/2023 16:03:13.928 | 8/5/2023 16:03:13.942 | 0:00:00.014 | 40325 | 4570089 | 4529764 | 61% |
| k-nearest neighbors(3) | 8/5/2023 15:53:02.995 | 8/5/2023 15:53:02.999 | 0:00:00.004 | 205745 | 1746405 | 1540660 | 96% |

Why scikit learn KNN runs faster than Naive Bayes?

Size of data set?

KNN algorithm?

KNN tree depth?

Summary

Perhaps more attributes are need to understand the performance benefits of Naive bayes. Fruits both fresh and rotten are more than the their colovlaues represented in an histogram.

# References

Christopher, A. (2021, Feb 2). *K-Nearest Neighbor*. Retrieved from medium.com: https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4

Gandhi, R. (2018, May 5). *Naive Bayes Classifier*. Retrieved from medium.com: https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c

Glen, S. (2019, Jun 19). *Comparing Classifiers: Decision Trees, K-NN & Naive Bayes*. Retrieved from datasciencecentral.com: https://www.datasciencecentral.com/comparing-classifiers-decision-trees-knn-naive-bayes/

*Naive Bayes*. (n.d.). Retrieved from scikit-learn: https://scikit-learn.org/stable/modules/naive\_bayes.html

Nayak, S. S. (2023, Jun). *Fresh and Rotten Classification*. Retrieved from kaggle.com: https://www.kaggle.com/datasets/swoyam2609/fresh-and-stale-classification

*Nearest Neighbors*. (n.d.). Retrieved from scikit-learn: https://scikit-learn.org/stable/modules/neighbors.html

Varghese, D. (2018, Dec 6). *Comparative Study on Classic Machine learning Algorithms*. Retrieved from medium.com: https://towardsdatascience.com/comparative-study-on-classic-machine-learning-algorithms-24f9ff6ab222#:~:text=A%20general%20difference%20between%20KNN,whereas%20KNN%20is%20non%2Dparametric.