**Classifying age categories of Abalones using physical features**

**What was done:**

Using a classic k-nearest-neighbours classification system, we approximated the age categories of the abalone in the given data set. A two-class classification system was chosen (Abalone-2) as it made calculating performance metrics much simpler in terms of determining which class was the positive or negative class. In this case, the “young” class (rings 10) was assigned as 0, and the “old” class (rings 11) assigned as 1.

Through some observation, it was found that the sex of an abalone alone is a poor indicator of determining the number of rings of an abalone has, and does little to differentiate a “young” and “old” abalone. Furthermore, the length, which is the most intuitive attribute when thinking of a way to determine the age of an abalone, was not alone indicative of the number of rings. Thus the other attributes had to be considered as well to create a useful measure of distance or difference between young and old.

In practice, it was found that the sex is useful in distinguishing infant abalone and “adult” abalone, as Infants were much more likely to be small and thus have fewer rings, this was fairly obvious but worth noting. “Adult” abalone were considered to be any abalone that was Male or Female. The “What was found” or results section will discuss in more detail what this means for the classifier.

Using the k-fold cross validation method, the original data set was split into k partitions, 1 partition was used as the test set, and the remaining k-1 sets were used as the training set. This partitioning was repeated k times so each partition had a chance to shine as the test set while the others took a seat as the training set. Each data point in the test set was assessed, and a list of its k nearest neighbours, sorted by score, was retrieved from the training set. Its predicted class was decided mostly by a majority vote, and in the case of a draw, the first item in the list of neighbours is drawn.

**What was found:**

Several similarity/distance metrics were used to determine the nearest neighbours, including Manhattan distance, Euclidean distance, and cosine (dis)similarity. Cosine dissimilarity was determined as 1-cosine similarity, for the sake of allowing 1 sorting method to sort all types of metrics regardless of means of calculation since we want a lower score to mean greater similarity, and cosine similarity works exactly opposite to that.

Of the available metrics, perhaps the least useful was cosine dissimilarity, as it seemed like most data points lie in approximately the same “line”. Thus many of the data points that “point” in the same “direction” but have different magnitudes will be seen to be the same from the perspective of the cosine dissimilarity. This is supported by intuition, as an abalone grows, given the same conditions, you would expect it to grow proportionally in all dimensions.

In preliminary testing, it was found that completely excluding the sex of an abalone in determining an abalone’s age resulted in somewhat acceptable accuracy, with approximately a 5% decrease in the accuracy of the classifier. Upon converting the sex into a 0 or 1 depending if it was an Infant or Male/Female (or Adult for short) and re-including it in the metric, the accuracy on average improved greatly but became slightly more varied, but not significant enough to be a concern.

Likewise, the Manhattan distance as a metric of accuracy was not as effective and results varied too unreliably to be used. The exact cause for this is unknown. Perhaps it is due to the attributes being all of slightly different scales, and thus a somewhat insignificant difference in one attribute can be seen as a large difference in the overall comparison of two attributes. While the exact reason for the unreliability is unknown, the results proved too unreliable to make confident estimations.

In the end and by process of elimination, the metric settled for was the Euclidean distance between two vectors as this was found to be both the most reliable and most accurate.

Selecting the k value for k-fold cross validation and (a different) k for k-nearest neighbours was simply a matter of re-running the test multiple times with varying values for both k’s. Ultimately it was found that k = 19 for k-fold with k = 7 for k-nearest neighbours worked to obtain the most accurate results. The results of which can be perused in the Appendix at your pleasure, with the appropriate grid highlighted to show the most accurate k values obtained of the values tested.

As such, the recommendations to be made based of the experimentation is to use k-fold with a value of = 19, using k-nearest neighbours with a value of = 7, and a distance metric of Euclidean distance because it provided the most reliable results and also the other metrics produced slightly more unreliable results.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 7 | 11 | 19 | 21 |
| 5 | Splits: 7 nn 5  acc: 0.7337008628954937  err: 0.26629913710450626  pre: 0.7070457354758962  sen: 0.39557399723374825  spe: 0.913059427732942 | Splits: 11 nn 5  acc: 0.731590309426721  err: 0.268409690573279  pre: 0.7030075187969925  sen: 0.3887733887733888  spe: 0.913059427732942 | Splits: 19 nn 5  acc: 0.7325162220620043  err: 0.2674837779379957  pre: 0.7001223990208079  sen: 0.3972222222222222  spe: 0.9099595736861448 | Splits: 21 nn 5  acc: 0.7385762385762386  err: 0.26142376142376145  pre: 0.704  sen: 0.4265927977839335  spe: 0.9045689019896831 |
| 7 | Splits: 7 nn 7  acc: 0.7370565675934804  err: 0.26294343240651963  pre: 0.6922222222222222  sen: 0.43173943173943174  spe: 0.8984976181751557 | Splits: 11 nn 7  acc: 0.7371072199568242  err: 0.26289278004317584  pre: 0.7285902503293807  sen: 0.3832293832293832  spe: 0.9244314013206163 | Splits: 19 nn 7  acc: 0.7433309300648883  err: 0.25666906993511174  pre: 0.7328339575530587  sen: 0.4073560027758501  spe: 0.9213235294117647 | Splits: 21 nn 7  acc: 0.7251082251082251  err: 0.27489177489177485  pre: 0.6817625458996328  sen: 0.38680555555555557  spe: 0.9043414275202355 |
| 11 | Splits: 7 nn 11  acc: 0.7404122722914669  err: 0.2595877277085331  pre: 0.734375  sen: 0.3908523908523909  spe: 0.9252473433492122 | Splits: 11 nn 11  acc: 0.7335092348284961  err: 0.26649076517150394  pre: 0.7133592736705577  sen: 0.3819444444444444  spe: 0.9190179552949799 | Splits: 19 nn 11  acc: 0.7317952415284787  err: 0.26820475847152125  pre: 0.7055137844611529  sen: 0.3898891966759003  spe: 0.9135075450864925 | Splits: 21 nn 11  acc: 0.7301587301587301  err: 0.2698412698412699  pre: 0.7275320970042796  sen: 0.35392088827203333  spe: 0.9297018770702982 |
| 17 | Splits: 7 nn 17  acc: 0.7281879194630873  err: 0.27181208053691275  pre: 0.7048748353096179  sen: 0.370242214532872  spe: 0.9178584525119179 | Splits: 11 nn 17  acc: 0.7169585032381867  err: 0.28304149676181334  pre: 0.6890156918687589  sen: 0.3342560553633218  spe: 0.919970631424376 | Splits: 19 nn 17  acc: 0.7317952415284787  err: 0.26820475847152125  pre: 0.7256267409470752  sen: 0.36180555555555555  spe: 0.9276001470047777 | Splits: 21 nn 17  acc: 0.7318422318422318  err: 0.26815776815776815  pre: 0.7363112391930836  sen: 0.3541233541233541  spe: 0.932596685082873 |

**Appendix and Raw numbers**

Note: Positives were considered Old abalone, Negatives Young. Therefore Specificity (spe) represents the percentage of times the model predicts young abalone out of the times it predicts a young abalone, sensitivity is likewise same as the above but for old abalone.