UPDATE: An Analysis of a KNN classifier with Scaling, (versus a Decision Tree?)

*Dirk Dosch*

[*dirk.dosch@trojans.dsu.edu*](mailto:dirk.dosch@trojans.dsu.edu)



Figure 1. Illustrate your task and/or approach

**1. Task**

Set up a k-nearest-neighbors model to classify car models. Base the model on a dataset of Ford cars.

Build a decision tree classifier and compare the results. Difficulties of the KNN model: Some attributes are categorical and will likely need to be converted to numerical form to be useful. Other attributes may impact the model negatively or interact with other attributes and must be sorted out carefully. Data will likely need to be scaled to prevent uneven weighting for different attributes.

**2. Approach**

Scale the data between 0 and 1 to deal with uneven weighting for attributes. Clean the data and make sure attributes are defined for all records. Replace attribute entries listed as ‘other’ with NaN (or perhaps remove the records with those entries.) Use KNN functions from the scikit learn library to build the classifier. Use the default metric (cartesian distance) for building the KNN model. Use k-fold cross validation to train and assess the classifier’s performance.

Most of the data analysis and training will be done with Jupyter notebooks. Python libraries include but are not limited to sklearn, pandas, NumPy, matplotlib, and IPython. Due to the likelihood of poor boundary separation, the KNN model will involve multiple dimensions and will not be easily visualized. If applicable, principal component analysis may be used to improve the performance of the model.

**3. Dataset and Metric**

The KNN model will be built on a [preprocessed dataset](https://www.kaggle.com/datasets/adityadesai13/used-car-dataset-ford-and-mercedes?select=ford.csv) for Ford cars. The dataset consists of records for online listings of used Ford cars. Each record consist of the attributes: *model*, *year*, *price*, *transmission*, *mileage*, *fuel type*, *miles per gallon*, and *engine size*. The target attribute is *model*. A total of 17965 records are provided.

For final analysis, the data will be broken up into a training set of 13474 records and a test set of 4491 records. The training set will also be broken into 10 folds for cross validation. Model performance will be evaluated on classification accuracy, calculated from the ratio of correctly classified models in the test set.

For example, 6 out of 10 correctly classified models yields an accuracy of 0.60. Accuracy of the final model will be compared to KNN without weights or scaling.

**4. Preliminary Results**

Talk about the progress you have made so far, specifically, what code implementation has been completed, what preliminary results you obtained on your dataset. Note, if you proposed data collection, the dataset should be ready to use by now, not still being collected/labeled.

**5. Detailed Timeline and Roles**

Each team mate should be assigned some non-trivial coding task.

|  |  |  |
| --- | --- | --- |
| **Task** | **Deadline** | **Lead** |
| Implement X | MM/DD/YY | Team Mate1 |
| Implement Y | MM/DD/YY | Team Mate2 |
| Run tests Z | MM/DD/YY | Team Mate1 |
| ... |  |  |
| Prepare report and presentation | 12/07/17 | all |

NOTE: All progress updates must follow this exact formatting, including fonts, and must not exceed 5 pages