Model Tuning

Session Plan

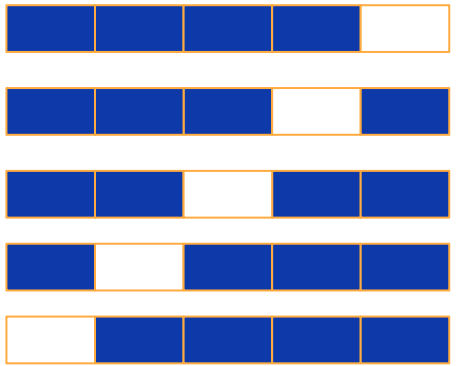
1. Introduction
2. Discussion Questions on the concepts
3. Hands-on Case study
4. Extended Discussions and QnA
5. Summary

Discussion Questions

1. What are the steps of cross-validation?
2. How to handle imbalanced data?
3. What is Data leakage?
4. How to deal with the situation where the model shows underfitting?
5. How to deal with the situation where the model shows overfitting?
6. Missing value imputation using KNN imputer

What are steps involved in cross-validation?

- Cross-validation is a technique used for evaluating models
- K fold cross-validation will divide data into k-folds
- Train model on k-1 folds and test its performance on the last fold
- K fold cross-validation will generate k models and k performance scores
- Instead of getting only 1 score, here we'll get k scores, which will give a better picture of the variance in model performance



How to handle imbalanced data?

- Datasets used in banking, health and market analytics usually have imbalance i.e. one class is in majority and one is in minority (less than 5%)
- During training on such datasets, the model gives more weightage to the majority class and gets biased
- To avoid such situations, we can use oversampling or undersampling techniques on data
- Oversampling will create artificial data points for the minority class
- Undersampling will remove data points from the majority class
- We can’t afford to lose data points in case of small data size, so oversampling is preferred in such cases

Data leakage

- Data leakage is the situation where the model, while it is being created, is influenced by test data
- Due to data leakage, model performance on test data is not trustworthy as the sanctity of test data is compromised
- Data leakage can happen in multiple ways.
    - Standardizing data before splitting into training and testing data. For e.g. using z-score
    - Imputing missing values for the entire data before splitting into training and testing data
    - Hyper parameter tuning to improve performance on test data
- **Best way to avoid data leakage** is to keep a portion of the sample data away before doing any processing

What is underfitting?

- We say a model is underfitting when it is not performing well on the train set
- This situation arises when a model is not able to learn from the train set

Reasons for underfitting

Small data size with a large number of features

Less model complexity

Irrelevant features

Imbalanced data

Dealing with underfitting

Increase model complexity, i.e. if you were using only a linear combination of features then try using a non-linear combination

In case of imbalanced data, use oversampling or undersampling

In the case of small-sized datasets with a large number of features, use features that seem important as per the need

What is overfitting?

- We say a model is overfitting when it performs good on train data but not good enough on test/unseen data
- This situation arises when the model starts learning the noise and inaccurate data entries

What could be the reason for this?

- High model complexity
- Small dataset
- Noisy data

Train accuracy = 98.01 !!

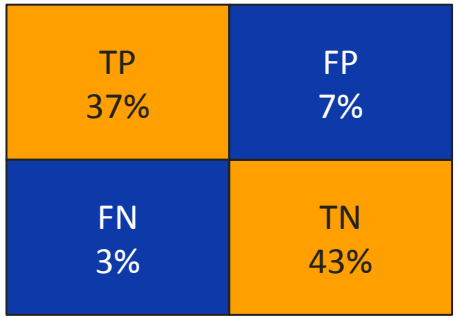
Test accuracy = 55.87

How to detect overfitting?

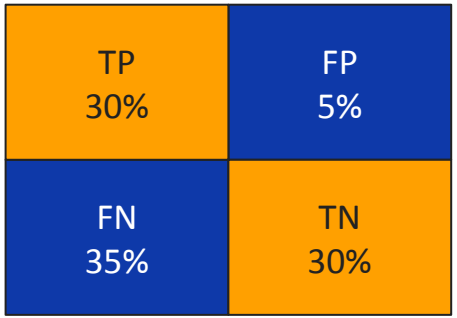
- Check model performance on train set and test set - if there is a huge difference in both, then we can say that model is overfit
- But sometimes we might get a biased train-test split i.e., train data has different distribution as compared to the test set
- So to confirm if we truly have overfitting or not, one must check model performance via cross validation

Dealing with overfitting

- Regularization
- Train with more data
- Remove irrelevant features
- Decrease model complexity



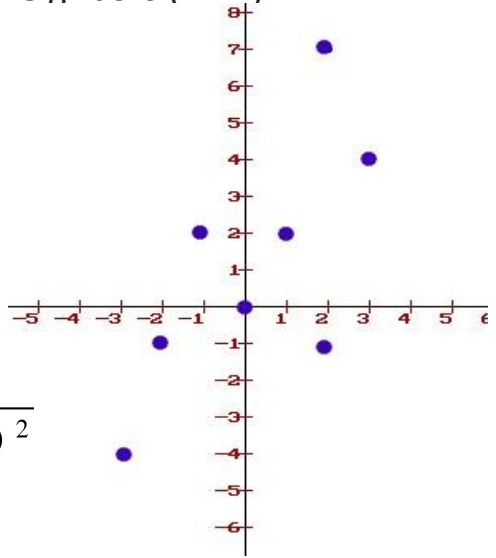
Confusion matrix on train data



Confusion matrix on test data

K- nearest neighbours

Before understanding how KNN Imputer works, let's understand what are K-nearest neighbors (KNN)



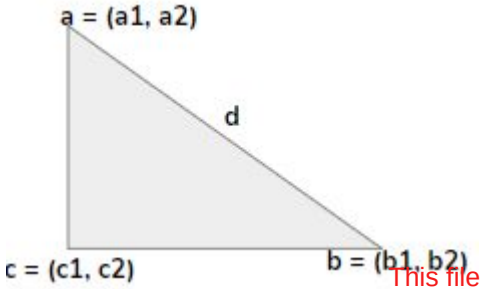
Looking at the graph what are the 4-nearest neighbors of (0, 0)?

1. Find the euclidean distance of (0, 0) from all other points
2. 4 Points with the least Euclidean distance will be the 4-nearest neighbors

How to calculate Euclidean distance?

The Euclidean distance between **a** and **b** is **d** and d is:

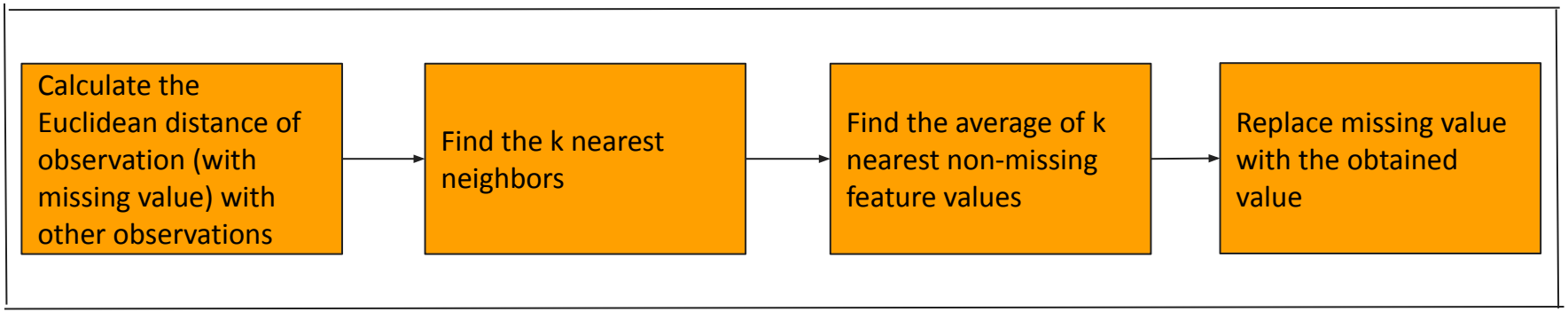
$$\sqrt{(a1-b1)^2+(a2-b2)^2}$$



KNN Imputer

How does KNN Imputer works?

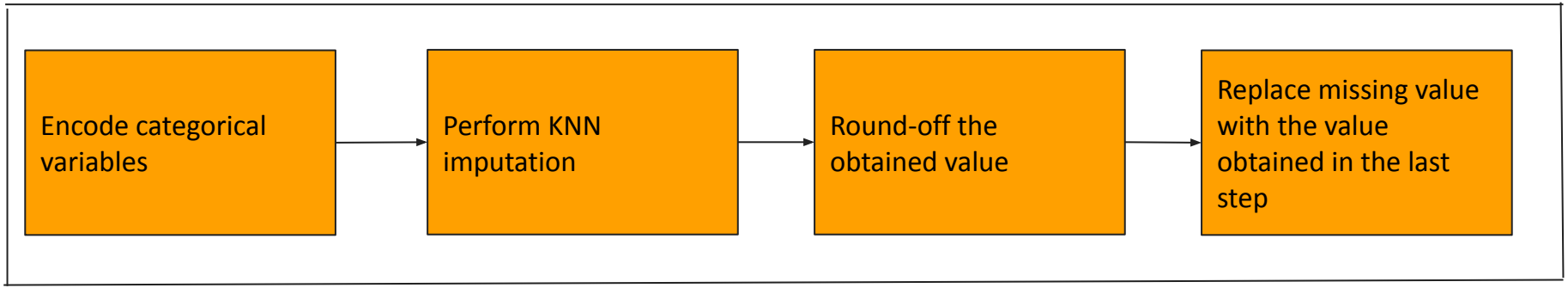
KNN imputer replaces missing values using the average of k nearest non-missing feature values (k needs to be decided by us)



KNN Imputer for categorical data

Since missing values are getting replaced with average values - How to do imputation in the case of a categorical variable?

Missing values in categorical data should be replaced with the nearest integer obtained via KNN imputer

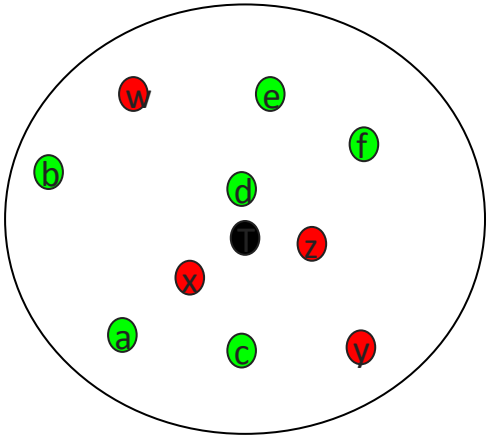


KNN Imputer for categorical data - example

- Looking at the image, assume red and green be 2 different categories of a variable
- And the black ball ‘T’, has this variable missing and we want to impute that

The process to calculate missing category of ‘T’

1. Encode green as 0 and red as 1
2. Consider k = 3, find out 3 nearest neighbors
3. We can see that d, x and z are 3 nearest neighbors of T
4. d has the category encoded as 0 and x, z as 1
5. So the average value is (0+1+1)/2 = 0.66
6. Rounding-off 0.66 we get 1, so the category assigned to ‘T’ is 1
7. Reverse encode the categories, so ‘T’ will be assigned the red category



greatlearning
Power Ahead

Happy Learning !