Model Tuning

Session Plan

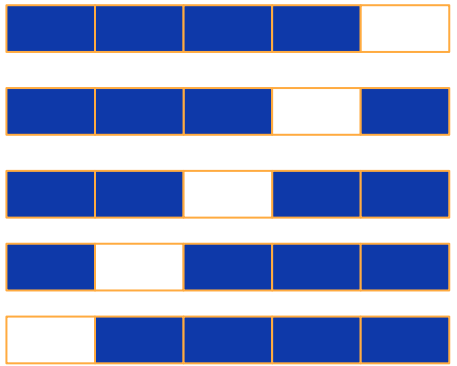
1. Introduction
2. Discussion Questions on the concepts
3. Hands-on Case study
4. Extended Discussions and QnA
5. Summary

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 underfitting and overfitting?
5. What is hyperparameter tuning?
6. Different types of hyperparameter tuning

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 !!

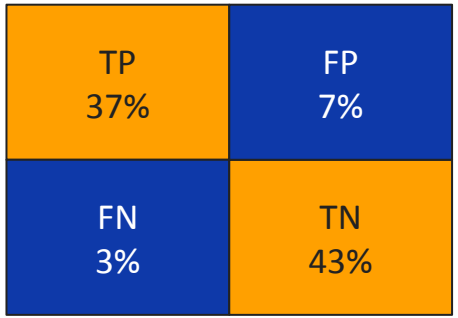
Test accuracy = 55.87

How to detect overfitting?

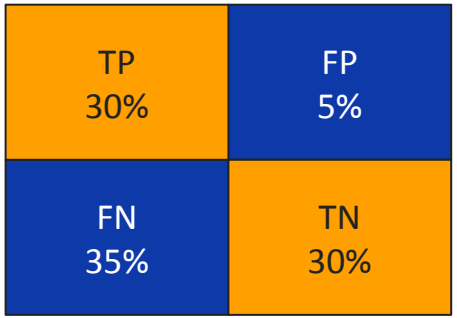
- Check model performance on train set and test set - if there is a huge difference in both, then we can say that model is overfit
- But sometimes we might get a biased train-test split i.e., train data has different distribution as compared to the test set
- So to confirm if we truly have overfitting 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

What is hyperparameter tuning?

- Hyperparameters are the parameters that govern the entire training process
- Their values are set before the learning process begins
- They have a significant effect on the model's performance
- The process of finding optimal hyperparameters for a model is known as hyperparameter tuning
- Choosing optimal hyperparameters can lead to improvements in the overall model's performance and can help in reducing both overfitting and underfitting

Different types of hyperparameter tuning

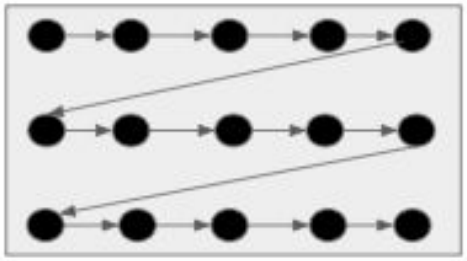
- Some models consist of a huge number of hyperparameters, and finding the optimal set of hyperparameters can be a very time-consuming process
- To make the process efficient, we'll look at 2 of the most common methods available in sklearn:
  - GridSearchCV
  - RandomizedSearchCV
- Grid search is best used when we have small search space, while Random search is best used when we have large search space
- We can use grid search to get the best possible results when we don't have any time constraints, but when we have time constraints, it's better to go with the random search
- Randomized search is known to give better results as compared to grid search

Grid Search

Grid search is a technique used to find the optimal set of hyperparameters for a model from the provided search space

Let's understand how grid search works, with an example

- Let this grey box be set of all possible hyperparameters
- Let these black circles indicate the search space
- Grid search will iterate over all black circles in a sequence
- And finally gives the best set of hyperparameters based on the best score obtained

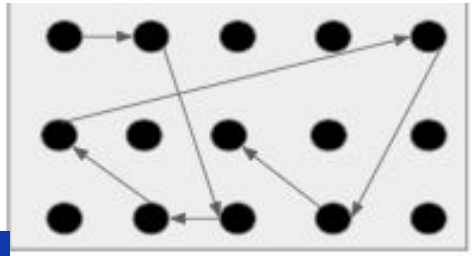


Doesn't work well on large search spaces

It will find the best set of hyperparameters but with high cost

Random Search

- Random Search is another technique to find the best set of hyperparameters which takes lesser time than grid search
- Random search is very similar to grid search, the difference is that in the random search
  - we define **'n_iter' -** not all parameter values are tried out, but rather a fixed number of parameter settings is sampled from the specified distributions. The number of parameter settings that are tried is given by n_iter.
  - the set of hyperparameters is not searched sequentially
  - We can pass a range here instead of just numbers
- Out of the entire search space of hyperparameter, only n_iter number of set of hyperparameters will be checked **randomly**



Works well on large search spaces
Gives better results than grid search

It doesn't guarantee finding the best set of hyperparameters

greatlearning
Power Ahead

Happy Learning !