greatlearning

Feature Engineering

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Learning Objectives

- Introduction to Feature engineering
- Part 2 of feature engineering and model tuning
- How to tune the models or improve performance
- Concept of upsampling and downsampling
- Case study on Feature engineering

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Introduction to Feature engineering

- **Feature engineering** is the process of using domain knowledge of the data to create features that make machine learning algorithms work.

- It is fundamental to the application of machine learning, and is both difficult and expensive.

- One reason why feature engineering is so important is that defining and/or learning higher level domain specific feature is actually one way to deep learning.

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Part 2 of feature engineering and model tuning

- Feature engineering is a set of those activities which performs a discipline we follow, to ensure that you finally feed the right data to your algorithms, in which the target variable is very strongly influenced by the independent variables.

- The independent variables do not have any significant outliers, all the missing values have been taken care of.

- As a part of data preparation, we have 2 techniques such as Upsampling and Downsampling

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

How to tune the models or improve performance

- To increase performance of any model, increase model capacity.

- The tuning process is more empirical than theoretical. We first observe the model whether is overfit or underfit in the first run then change the values of the parameters involved in the model so that we gradually move towards a good fit.

- We repeat the iterations until the accuracy improvement is diminishing and no longer justify the drop in the training and computation performance

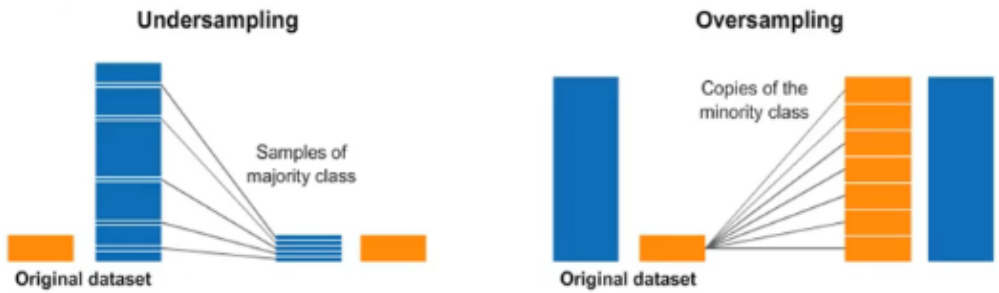
| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Concept of upsampling and downsampling

greatlearning

- We can handle the imbalanced dataset cases to minimize the Type II errors by balancing the class representations.

- To balance the classes we can –
  - Decrease the frequency of the majority class
  - Increase the frequency of the minority class



| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Imblearn techniques

- Python imbalanced-learn module – provides more sophisticated resampling techniques.
- In over-sampling, instead of creating exact copies of the minority class records, we can introduce small variations into those copies creating more diverse synthetic samples.
- SMOTE consists of synthesizing elements for the minority class, based on those that already exist, works randomly picking a point from the majority class and computing the K-nearest neighbours.

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Case study on Feature engineering

greatlearning

Context:

The dataset contains 1000 entries with 20 categorial/symbolic attributes prepared by Prof. Hofmann. In this dataset, each entry represents a person who takes a credit by a bank. Each person is classified as good or bad credit risks according to the set of attributes. The link to the dataset can be found below.

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Attribute information

- Age (numeric)
- Sex (text: male, female)
- Job (numeric: 0 - unskilled and non-resident, 1 - unskilled and resident, 2 - skilled, 3 - highly skilled)
- Housing (text: own, rent, or free)
- Saving accounts (text - little, moderate, quite rich, rich)
- Checking account (numeric, in DM - Deutsch Mark)
- Credit amount (numeric, in DM)
- Duration (numeric, in month)
- Purpose (text: car, furniture/equipment, radio/TV, domestic appliances, repairs, education, business, vacation/others)

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Steps to follow

1. Import the necessary libraries
2. Get the data.
3. Perform the summary of the dataset and compare values.
4. Randomly select 50% data for this use case
5. Lets build a Ensemble model but need to modify the dataset first
6. Check for highly correlated variables
7. Split Train/Test data 70:30 ratio
8. Build RF Model
9. calculate Confusion Matrix
10. View a list of the features and their importance scores
11. Calculate test and train accuracy.
12. Perform K-fold cross validation

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

Additional steps

1. Perform k-fold cross validation with stratification (Stratified cross-validation)
2. Bootstrapping (Create a model with each bootstrap sample and validate it with the test set)

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

HappyLearning

Questions?

| Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited