Fundamentals of Machine Learning - 2022

Practice 2 - Linear and Tree-based models
August 26 2022

This practice is intended for your own exercise and **does not** need to be turned in.

1. Questions

1. What are the two most common supervised tasks?
2. Can you name four common unsupervised tasks?
3. Can you identify which elements represent the task, the performance metric and the experience, for a linear regressor? And for the knn algorithm?
4. What is the purpose of a validation set?
5. What is the difference between a model parameter and a learning algorithm's hyperparameter?
6. If your model performs great on the training data but generalizes poorly to new instances, what is happening? Can you name three possible solutions?
7. What can go wrong if you tune hyperparameters using the test set?
8. What purpose does a Loss (or Cost, or Objective) function serve?
9. What is an optimizer?
10. What is the difference between precision and recall? Do you know a way to summary both in a single metric?
11. What could be the problem of severe class imbalance, and how can you try to mitigate it?

2. Problems

2.1. Data

Lets begin with our first ML Proyect, the end goal here is to predict California Housing Prices. We will start by loading and describing the dataset.

1. Take a first look at the dataset. Which type of features do you see? How many NonNull values are there for each one of them?.
2. Are there any categorical features? How many values do they take?.
3. Divide your dataset in two, a training set and a test set. From now on we will work on the training set and leave the test set to the end for model evaluation. REMEMBER NOT TO CHEAT.
4. Do some visulalization of the training set. Try to:
   - plot a histogram of each feature.
   - scatter points with prices over a map.
   - check for correlations.

   Don't restrain yourself from being creative.
5. (Optional) Find new features with non linear combinations of the originals. How can you check if they are useful?
6. Data Cleaning: removing NaNs. To address this issue we have some options, these are:
   - remove rows with incomplete information or NaNs.
   - remove those features with NaNs.
   - replace NaNs with some statistic of the Feature. Some possibilities are: mode, mean, median, some random value between the 40th and the 60th quantile, etc.

2.2. Linear models and Trees Regression

2.2.1. Preprocessing

We first have to prepare our training set for the ML training. In order to do this:

1. Apply an *Ordinal Enconder* and a *One Hot Encoder* to the categorical data. What is the difference between these two?.
2. Apply a standarization process to the numerical data. Namely, transform each feature such that it has zero mean and unit variance.
3. (Optional) Explore other options of scaling: MinMax, NonLinear, etc.

2.2.2. ML

We will work with the following regression algorithms: Linear Regressor, a Decision Tree Regressor and a Random Forest Regressor. For each one of them :

1. Apply the algorithm to the training set.
2. Plot some features along with their predicted values.

Can you forsee the performance of these algorithms by the plots of their predictions?

2.2.3. Metrics

1. In order to evaluate the performance of each algorithm calculate the *RMSE* and the *MAE* to their results.
   - Which one has the best score?
   - Is there a way to spot overfitting by their value?
2. Apply a 10-fold CrossValidation to each model and calculate the mean and standard deviation for the 10 folds as a whole. How can you interpret these statistics?
3. Select an algorithm due to its perfomance
4. Fine tuning over Crossvalidation:
   - apply a GridSearch over a 10-fold CV to find the best combination of Hyperparameters
   - Repeat the process with a random search over the 10-fold CV.
5. Train your model with the best hypter parameters to the whole training set. This is now your final model.
6. Evaluate the test set with the predictions from the final model. Compare this score with the one obtained in the training set.
7. Last but not least, export your final model so you can share it or eventually deploy it into prouduction.