Doktar

Data Engineer Case Study

1-) Data Reporting

You are provided two datasets: raw dataset and final report of the grain harvest prediction in Turkey. You are expected to reach the final report using the raw dataset by taking the following steps:

i. In the raw dataset, the date columns (GrainHarvestAreaDATE) are not ordered. Reorder the columns based on the dates.

ii. If "***GrainHarvestAreaDATE*** " == "***GrainHarvestArea20211231***", it implies that the weather is too cloudy to get a proper satellite image. Therefore, you should change the name of the column accordingly. This column corresponds to the "***Bulut Engeli***" in the final report.

iii. If "***GrainHarvestAreaDATE*** " == ***"GrainHarvestArea20220101***", it implies that there will be no harvest soon. Therefore, you should change the name of the column accordingly. This column corresponds to the "***Yakinda Hasat Yok***" in the final report.

iv. Create a column for total grain area by the sum of all columns you have, and name it accordingly. This column corresponds to the "***1. Urun Danelik Misir***" in the final report.

v. Create columns for the dates below and each column should show the cumulative sum of the harvested areas until that date.

27/09/21, 10/10/21, 17/10/21, 24/10/21, 30/10/21, 07/11/21, 14/11/21, 21/11/21, 28/11/21

vi. When you complete these steps, you will reach the following column structure which corresponds to the "Final_Dataset" sheet on the final report:
CityName, DistrictName, Placename, Total Area, Cloud Barrier, NoHarvest, and the cumulative sum columns for the dates above.

Now, for each city, you should sum over it's all districts and places to reach city-level report which corresponds to the "Summary" sheet. Notice that the "Summary" also includes a regional-level analysis (like Cukurova) but it is not needed for you now.

At the end you should have the following column structure:
CityName, Total Area, Cloud Barrier, NoHarvest, and the cumulative sum columns for the dates above.

vii. Create a docker container that implements the above steps automatically in a local file system.

Expected command line*: docker run–it your_docker_image –v local_path:docker_container_path python harvest_report.py --input harvest_input.csv -output output_name.csv*

Expected files*: Dockerfile, harvest_report.py and readme.md*

2-) Suppose you have a binary classification problem with a large enough dataset and a high number of variables. You have run logistic regression, XGBoost, and neural networks algorithms. Initial results show that XGBoost has the best accuracy metrics, and you reported the best results to the data team lead. However, your data team lead provided feedback that interpretability is a crucial factor for this problem. Can you explain the trade-off that you face? Can you shortly explain what is the main intuitive advantage of the XGBoost algorithm in general? If you choose logistic regression for interpretability, what could be potential problems? How do you deal with them?

This question is optional. You can try to answer if you are also interested in data science. There is not a correct answer, try to be creative.