Machine Learning Engineer Nanodegree - Capstone Project

Andrea Rabinelli

PROBLEM STATEMENT

For the Machine Learning Engineer Nanodegree Capstone project, I decided to focus my effort on the healthcare sector. The recent outbreak of Covid-19 clearly showed the scarcity of our healthcare resources and forced us to switch our mindset and actions to be much more conscious of how our usage of those resources might affect others in need. While the pandemic has amplified the issue, it is not something new to healthcare professionals.

One of the facets of this issue is patients not showing to medical appointments, therefore not making use of slots that could have been dedicated to others in need. In the US only, **medical appointment no shows are believed to have an incidence between 20% and 30%** on the total number of appointments, with a study from 2006 estimating the yearly cost at 150 billion USD[1].

Scientific publications report that, when applying machine learning and data mining techniques to the studied context, these technologies outperformed the traditional management of no-shows (e.g. Srinivas, Sharan, and A. Ravi Ravindran (2018)[2])

GETTING THE DATA

Fortunately, Kaggle offers a great dataset to work with: the Medical Appointment No Show dataset[3]. This dataset is made up of over 110k rows and 14 variables, collecting anonymized data and a binary outcome (show/no-show) of medical appointments in public hospitals of Victoria, Brasil. Therefore, from a machine learning perspective, this is a **binary classification problem**, as the target variable to predict is whether or not a patient will show up to a medical appointment.

The dataset presents class imbalance: 79.8% of the records refer to appointments when the patient showed up. As no verified model exists for this dataset, the

benchmark prediction will be the “naive” case of always predicting that a patient will show up, as it currently happens in most clinics[2].

MEASURING SUCCESS

Given the class imbalance in the dataset, the first metric that comes to mind is the AUC ROC (Area under Curve of the Receiver Operating Characteristics). This metric is particularly fit for this task as it assesses the ability of the model to distinguish the two classes. However, **by definition of the AUC ROC metric, the naive model would score the lowest possible value** (0.5 - as it does not distinguish the two classes at all). This means that even a model that performs worse than the “naive” model in terms of our objective of identifying patients not showing up could have a higher AUC ROC compared to the baseline, which misses 1 patient out of 5. However, this metric can still be used to compare different models that will be explored.

Recall, precision, and F1 score are other relevant metrics for the task, however, when we try to predict who will NOT show up as the positive case, they assume a value of 0 or undefined (NaN). For this reason, it is more interesting to **define the problem as identifying which patients will show up**, and compare them with the baseline metrics (Accuracy = 79.8, Recall = 100%, Precision = 79.8%, F1 = 88.76%, AUC ROC=0.5). Finally, I’ll be monitoring the **False Discovery Rate** (False Positive/Predicted Positive) and **Negative Predictive Value** (True Negative/Total Negative) to identify how many no-shows will be missed by the model.

PROJECT OUTLINE

The initial focus will be on cleaning and exploring the dataset, using the obtained insights to engineer features that might be relevant for solving the task. As other notebooks on Kaggle have already focused on building random forests, linear regressions, and Xgboost classifiers[4], I will investigate the performances of a deep and wide network as proposed in the paper from Cheng, Heng-Tze, et. Al (2016)[5] as

a network capable of both generalization (e.g. birds can fly) and memorization (e.g. penguins don't fly). Initially suggested as an architecture fit for recommender systems, the combination of wide and deep networks has been proven useful in other contexts (e.g. electricity theft detection[6]). To assess the performances of this specific architecture, the model will be benchmarked with an Xgboost model sharing the same features. Both models will be optimized with hyperparameter tuning. The ability of the model to generalize to new cases will be assessed by performing a training-test-validation split.

Sources & Notes:

1. https://hbr.org/2010/03/how-behavioral-economics-can-h
2. Srinivas, Sharan, and A. Ravi Ravindran. "Optimizing outpatient appointment system using machine learning algorithms and scheduling rules: a prescriptive analytics framework." *Expert Systems with Applications* 102 (2018): 245-261.
3. https://www.kaggle.com/joniarroba/noshowappointments
4. Most of the model reviewed either didn't show major improvements over a naive model or showed potential sources of data leakage in the creation of the training and test sets. For this reason, they will not be considered as benchmark models.
5. Cheng, Heng-Tze, et al. "Wide & deep learning for recommender systems." Proceedings of the 1st workshop on deep learning for recommender systems. 2016.
6. Zheng, Zibin, et al. "Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids." IEEE Transactions on Industrial Informatics 14.4 (2017): 1606-1615.