If you want to take the `Data Science Laboratory Course` in summer 2024, you need to solve this task. If you don't want to participate in the practical course, you can use this task as an additional exercise. However, we won't publish a solution.

Scenario. We modeled a simultaneous-multi-round auction with four bidders and six products as a business process. Next, we used a model checker to verify whether certain outcomes of the auction are possible or not, e.g., if a certain bidder can win a certain product at a certain price. (Understanding how the auction or the verification procedure works isn't necessary to solve the task.) An outcome of the auction consists of a `price` for a `product`, and sometimes a `winner` (can be zero if only the price is verified). Verification also depends on the `capacities` of the four bidders. Capacities tell how many products a bidder is allowed to win. Our goal is to find out if the `verification_result` becomes true or not, depending on the outcome and the capacities.

Task. Create a `Jupyter Notebook` to analyze the given scenario. We provide training data (feature values and target variable) and test data (only feature values). Your notebook needs to contain the following steps:

1. **Exploring** the data, i.e., computing statistics and creating plots.
2. **Training** at least one baseline and at least one proper prediction model.
3. **Evaluating** the model(s) you chose with `sklearn.metrics.matthews_corrcoef()`.
4. **Predicting** on the test data and creating a file with the predictions.

Assessment. We will assess your solution according to the following four criteria:

1. **Methodological quality**, i.e., using suitable data-science techniques and interpreting all results appropriately.
2. **Code quality**, but not quantity.
3. **Prediction quality** on the test data, but only to some extent: Your solution should be better than a baseline, but we don't expect hyperparameter tuning. In fact, we should be able to run your notebook in under a minute on a modern consumer laptop.
4. **Reproducibility** and adhering to the prescribed solution format.

Submission Format. Send your solution to federico.matteucci@kit.edu until Tuesday, 29 February 2024. Submit three files individually, i.e., do not put them in a zipped folder: A `Juypter Notebook`, an HTML export of the notebook, and a prediction file. Name your files like `NAME_solution.ipynb`, `NAME_solution.html`, and `NAME_prediction.csv` (for example, `Marco_Heyden_prediction.csv`). **You need to work on the solution for this task on your own.**

The notebook needs to contain your entire code and interpretation of results. Also, execute and save it before submission, so it contains all output. Make sure it runs in sequential order, i.e., from start to end, without any manual intervention necessary. Remove code snippets that don't have a purpose anymore or whose results you don't interpret. If your solution uses packages not contained in the exercises' `requirements.txt`, please provide a requirements file.

Assume the input data are stored in a directory `data/`, located in the same directory as your notebook. Save the prediction file to `data/` as well. The format of the prediction file needs to be the same as for the `target` file of the training data, i.e., a one-column CSV (so actually no commas), the same column name, correct number of data objects, Boolean class labels, and no quotes around values.