

Predict the Success of Bank Telemarketing

MLP Project T32024

About this Competition

The data is related to direct marketing campaigns of a banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required in order to determine if the product (bank term deposit) would be subscribed (*yes*) or not (*no*).

Your goal is to build a model (e.g., a Multilayer Perceptron) to predict whether a client will subscribe to a term deposit based on their demographic and historical interaction data.

Files

- **train.csv** – The training set
- **test.csv** – The test set
- **sample_submission.csv** – A sample submission file in the correct format

Input Variables

#	Variable	Description
1	last contact date	Last contact date
2	age	Age of the client (numeric)
3	job	Type of job
4	marital	Marital status (categorical: "married", "divorced", "single"; note: "divorced" means divorced or widowed)
5	education	Education level (categorical: "unknown", "secondary", "primary", "tertiary")
6	default	Has credit in default? (binary: "yes", "no")
7	balance	Average yearly balance in euros (numeric)

#	Variable	Description
8	housing	Has a housing loan? (binary: "yes", "no")
9	loan	Has a personal loan? (binary: "yes", "no")
10	contact	Contact communication type (categorical: "unknown", "telephone", "cellular")
11	duration	Last contact duration in seconds (numeric)
12	campaign	Number of contacts performed during this campaign for this client (numeric, includes last contact)
13	pdays	Number of days that passed by after the client was last contacted from a previous campaign (numeric; -1 means not previously contacted)
14	previous	Number of contacts performed before this campaign for this client (numeric)
15	poutcome	Outcome of the previous marketing campaign (categorical: "unknown", "other", "failure", "success")

Output Variable (Target)

#	Variable	Description
16	target	Has the client subscribed a term deposit? (binary: "yes", "no")

Getting Started

1. Explore the data

- Load `train.csv` and examine distributions, missing values, and correlations.

2. Preprocess features

- Encode categorical variables (one-hot or label encoding).
- Scale numerical features if needed (e.g., `StandardScaler`).

3. Modeling

- Build a Multilayer Perceptron (MLP) using frameworks such as TensorFlow/Keras or PyTorch.
- Experiment with network architecture, activation functions, learning rates, and regularization.

4. Evaluation

- Use cross-validation on the training set to tune hyperparameters.
- Evaluate on a hold-out validation set or via k-folds.
- Metrics: accuracy, precision, recall, F1-score, ROC AUC.

5. Submission

- Train your final model on the full training set.
 - Generate predictions (`yes / no`) on `test.csv` .
 - Save results in the same format as `sample_submission.csv` and submit.
-