

# RNN Based Telecom Customer Churn Prediction

## Introduction

Customer churn is a major problem and one of the most important concerns for large companies. Due to the direct effect on the revenues of the companies, especially in the telecom field, companies are seeking to develop means to predict potential customers to churn. Predictive analytics use churn prediction models that predict customer churn by assessing their propensity of risk to churn.

## Dataset

This dataset contains more than 7000 entries and 21 features. All entries have several features and of course a column stating if the customer has churned or not. There are 26,5% of the total amount of customer churn in the dataset. The dataset in CSV format can be downloaded from [1].

Here are the details of the features in the dataset:

Feature name	Data type	Description
class (target)	nominal	2 unique values (0 missing)
state	numeric	51 unique values (0 missing)
account_length	numeric	218 unique values (0 missing)
area_code	nominal	3 unique values (0 missing)
phone_number	numeric	5000 unique values (0 missing)
international_plan	nominal	2 unique values (0 missing)
voice_mail_plan	nominal	2 unique values (0 missing)
number_vmail_messages	numeric	48 unique values (0 missing)
total_day_minutes	numeric	1961 unique values (0 missing)
total_day_calls	numeric	123 unique values (0 missing)
total_day_charge	numeric	1961 unique values (0 missing)
total_eve_minutes	numeric	1879 unique values (0 missing)
total_eve_calls	numeric	126 unique values (0 missing)
total_eve_charge	numeric	1659 unique values (0 missing)
total_night_minutes	numeric	1853 unique values (0 missing)

total_night_calls	numeric	131 unique values (0 missing)
total_night_charge	numeric	1028 unique values (0 missing)
total_intl_minutes	numeric	170 unique values (0 missing)
total_intl_calls	numeric	21 unique values (0 missing)
total_intl_charge	numeric	170 unique values (0 missing)
number_customer_service_calls	nominal	10 unique values (0 missing)

## Mission

**The final objective is to build a classifier to predict customer churn among the customers in the data. The output of the system would be an insight for the companies about the important factors that accelerate the churn process.**

To do that, the following tasks need to be accomplished:

1. Understand the content that is available in the dataset.
2. Pre-process the data and do feature selection to extract the most important features.
3. Use a two fold cross-validation approach for evaluating all of the algorithms mentioned below.
4. Feed Forward based classification
  - a. Build a feed forward neural network to classify the records into one of the two categories.
  - b. Compare the performance of different activation functions and loss functions on the proposed task.
  - c. Analyze the outcomes.
5. Classification method of your choice
  - a. Use a classifier of your choice to classify the records into one of the two categories.
  - b. Analyze the impact of hyperparameters on the performance of the model on validation and test sets.
  - c. Analyze the outcomes.
6. Use the different packages of visualization explained during the course to visualize findings from both approaches.
7. Compare the results from both classifiers from the accuracy point of view.

## Deliverables

To carry out the assessment of the project, the group has to submit the following:

- A report using Google Doc and explaining the concept about the project solution and the expected division of the tasks regarding the components of the group. This document should not be longer than 10 pages (including cover, table of contents, etc).
- Collaborative work using Git with commit + push changes on a daily basis.
- A presentation explaining the thought process, your approaches and the reason for this choice, your findings and the real task division in your group at the end of the project. Every group has 15 minutes per presentation and there will be 5 minutes of questions.

[1] [https://www.openml.org/data/get\\_csv/4965302/churn.arff](https://www.openml.org/data/get_csv/4965302/churn.arff)