# Machine Learning Engineer Nanodegree

## Predicting Telstra Network Disruptions

Jeremias Binder
May 28th, 2017

---

## Domain Background

Telstra is australias biggest telecommunications and media company with an annual revenue of over 20 billion USD (A$27.1 billion).
In a recent Kaggle competition, Telstra challenged people to predict the severity of service disruptions on their network. Using a dataset of features from their service logs, the task was to determine whether a disruption was a momentary glitch or a total interruption of connectivity.

The competition ended at 2/29/2016, the winner was a Kaggle user called 'Mario Filho'. Some participants uploaded their kernels on the Kaggle forum, but the winning solution is not available there. In total, 974 teams participated in the competition.

One of the reasons i choose this project as my capstone project for the Machine Learning Nanodegree at Udacity, is that the company i am currently working at, the sovanta AG in Heidelberg, Germany, has some projects running in the domain of predictive maintenance.

## Problem Statement

The problem we are about to solve, is to minimize the reaction time to interruptions in Telstras network. Long reaction times lower customer satisfaction in the long run and can therefore be costly to Telstra. A model, that can predict accurately the network failures in advance, would be very valuable to Telstra and its customers.

The problem right now is, that Telstra has no good estimation on when one of their nodes will fail. They might have clues (certain nodes reporting an error), but no further conclusions are drawn from this data.
Our model should change that: The data provided by the nodes will be used to create a model, that will accurately predict failures on the network.
Since the input information is digitally obtained, each error message can be put in a certain category and is distinct. Since it's a future prediction, it's easily verifiable: After the event is predicted, the actual time and place can be observed and the degree to which the prediction is correct can be verified.

## Datasets and Inputs

The goal of the problem is to predict Telstra network's fault severity at a time and a particular location based on the log data available. Each row in the main dataset (train.csv, test.csv) represents a location and a time point. They are identified by the "id" column, which is the key "id" used in other data files. Fault severity has 3 categories: 0,1,2 (0 meaning no fault, 1 meaning only a few, and 2 meaning many). Different types of features are extracted from log files and other sources: event_type.csv, log_feature.csv, resource_type.csv, severity_type.csv.

Note: "severity_type" is a feature extracted from the log files (in severity_type.csv). Often this is a severity type of a warning message coming from the log. "severity_type" is categorical. It does not have an ordering. "fault_severity" is a measurement of actual reported faults from users of the network and is the target variable (in train.csv).

*File descriptions*

train.csv - the training set for fault severity
test.csv - the test set for fault severity
sample_submission.csv - a sample submission file in the correct format
event_type.csv - event type related to the main dataset
log_feature.csv - features extracted from log files
resource_type.csv - type of resource related to the main dataset
severity_type.csv - severity type of a warning message coming from the log

(Telstra Network Disruptions. (2016). Retrieved June 1, 2017, from the competitions page).

## Solution Statement

A solution to the above problem is a testset, which includes as many accurate predictions of issue severity as possible. The testset will eventually be evaluated by Kaggle (after submitting the prediction).
If the the submitted set scores a sufficient low logloss and is generalizable on new data, the problem can be considered solved for Telstra, since the model could be used to predict other data as well.

## Benchmark Model

One (or several) benchmark models are provided in the discussion forum of the competition, since it already finished and some of the participants published their solution. Also, the leaderboard provides a good check on the quality of the model: All the models are evaluated with the below described multi-class logarithmic loss metric and are therefore directly comparable with each other.

## Evaluation Metrics

The thing Telstra is interested in, is when and where their nodes are likely to fail. The quality of the prediction therefore matters:
To determine the quality of such a model the multi-class logarithmic loss is used. Each data row has been labeled with one true class, which represents the severity of the incident (an incident with label '0' means, there is no issue). For each row, a set of predicted probabilities is submitted (one for every fault severity). The formula is then,

$$logloss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} log(p_{ij}),$$

where N is the number of rows in the test set, M is the number of fault severity classes, $log$ is the natural logarithm, $y_{ij}$ is 1 if observation $i$ belongs to class $j$ and $0$ otherwise, and $p_{ij}$ is the predicted probability that observation $i$ belongs to class $j$.
Or in layman terms: The lower the sum of wrongly predicted severities, the lower the logloss. Participants with a lower logloss on the testset get ranked higher on the public leaderboard.
Scoring a low logloss on the testset is important, since the model might overfit on the trainset.

## Project Design

*Data Exploration*
To start the Project, a proper understanding of the problem and the data is key:
To achive an in-depth understanding, the available data needs to be visualized (matplotlib, seaborn, ggplot) and discussed thoroughly. The first step will be a univariant analysis of each feature to really grasp the single components our model is going to consist of.
After that, the data exploration to then conclude with a multivariate analysis, to see what the relation between the features is. After this, i hope to have a better insight on which algorithm might work well and what are the actual upcoming challenges in the project.

*Preprocessing/Feature Engineering*
After the data is visualized and i got a more detailed understanding, the data needs to be preprocessed. I will consider the following steps to preprocess the data in the first place:
- Data cleaning (Remove outliers, smooth out noisy data)
- Transform data (One hot encoding, normalization when necessary)
- Data reduction (Some features might not be necessary, binning, maybe principal component analysis)

I might extend the list and do some feature engineering when necessary, but as of now, it seems to me, that the data provided by Telstra is quite clean and thought-out.
Furthermore, i will create a custom test- and validation set (since test.csv is not an actual testset, but a set to be submitted to Kaggle and checked on its accuracy).
Also, since the Telstra dataset is distributed in several files, these files have to be put in one coherent panda data frame.

*Model creation and evaluation*
Finally, a model will be applied. Since we have labeled data, a supervised kernel can be applied (like a decision-tree, boosting, naive bayes, or random forests). In order to experiment with different algorithms and techniques, a prediction pipeline has to be set up, including a prediction on the (personal) testset and evaluation with the multi-class logarithmic loss.

I will try different algorithms and tweak the parameters until a satisfying logloss is reached.
Also, an ensemble learner might help out here. Finally, predictions will be made on the personal validation set, then on the test.csv and are finally submitted to the Kaggle competition.