Approach Document for Analytics Vidhya Data Engineering Competition

A brief on the approach, which you have used to solve the problem.

I've used Python to make a reusable and scalable code that would our Data science team, and in turn, help my client ComZ – to get a clean data source to be used for model input and training.

The code generally helps in data cleaning by using various methods like strings manipulation, variable imputation, and business logic implementations.

What data-preprocessing / data cleaning ideas really worked? How did you discover them?

Since this was a complex and dirty data, the path to the right solution was muddy. Overall, what really helped was to start thinking why the data is incorrect, and what all could work to make it better.

- For all strings variables, one thing which stood out was their case (upper/lower/elephant). First step was to make all variables in upper case.
- Secondly, the date time format was 2-fold – normal date time and Unix datetime. I cleaned it by separately treating Unix datetimes and then appending back with the main data.
- Third, there were a lot of missing values in Activity, ProductID and VisitDateTime columns.
- For Activity, this field couldn't be blank as a user had to do some activity when he has visited the website. For cases when ProdID equals prior ProdID, the used must have 'CLICK'ed. Else, Pageloads.
- Similarly, for ProdID, if the user is browsing from the same device on and around the same time, it has to be the first/prior non null value.
- And VisitDateTime could not be blank for customers who had a UserID and ProdID.

After all the data cleaning/imputations, I broke down the dataset to calculate all input features separately and, in the end, joined them all back together on UserID. This not only makes the code reproducible, but also makes sure it's easier to find problem in 1 feature rather than multiple features.

Which tools did you use to solve the problem?

I used Python with Pandas, Numpy, Time and Datetime libraries.