Predict Gender using Names

Submitted by: Surabhi Singhal
Date: 13th May, 2018

Highlights:

Software used: Python 3, Jupyter Notebook

Key impact maker: Feature engineering using first name and last name

Model Used: Random Forest

R^2 on Train = 0.91

R^2 on Test = 0.48

Steps Followed to Identify Gender

1. Importing all the required packages in python
2. Data Loading: importing all 8 csv files in python
3. Data Cleaning:
    a. Getting same column names and removing spaces from column names to get uniformity for all 8 files
    b. Appending all 8 files to 1 : all_data
4. Data Visualization: to know the key variables, distribution pattern, outliers
    a. Scatter plot to find NA entries
    b. Distribution of Male Female in the data : 20% female, 80% male
    c. Distribution of Male Female within each race
5. Data Preprocessing/Feature generation: Based on inputs from visualization, doing the next step of creating features
    a. Data Split to test & train: manual splitting done to ensure that 20-80 proportion of male female is restored based on gender and race both.
    b. Changing Gender from Female as 1 and Male as 0
    c. Dropping Race since it's not used as in input (reference: example in the email by Shashank)
    d. Dropping all NA entries since models can not be implemented if NA values exists
    e. Removing spaces from first name and last name; converting all of them in lower case. (First name has 1st character blank and hence removing this)
    f. New feature generation:

    i. first name: First letter, First 2 letters, First 3 letters, Last letter, Last 2 letters, Last 3 letters, length of the string
    ii. last name: First letter, First 2 letters, First 3 letters, Last letter, Last 2 letters, Last 3 letters, length of the string
6. **Modelling: Decision Trees** was used but the model didn't show good results. R^2 on train: 0.99; R^2 on test: 0.37. clearly over fit. Also, predictions were Booleans (0 or 1), no float values which seemed a little against expectations. Hence tried a different model
7. **Modelling: Random Forest showed better results with respect to Decision trees.** R^2 on train: 0.91; R^2 on test: 0.48. predictions are not in float values which makes sense. Still an over fit though.

Key Problems Identified and tackled

1. Size of the data sets and proportion of male female is vast. The random split can't happen properly. Hence a **manual split to ensure male female** is captured in adequate proportion as it is present in the complete data set.
2. **Spaces removed** from first name and last name for better feature generation

Further scope of improvement

1. Last Name is not present for Indians. Which means all values NaN. Hence it can't be used in the model. Therefore, I had to drop all Indian observations. If **SNU: SURNAME UNKNOWN** can replace NaN values, observation loss will not happen.
2. Replace first and last names written in Hindi script to English.
3. Split Indian names between first name and last names. The cases were very few in number and hence I didn't do it.
4. I didn't use Race in the model because I wasn't sure that will it be given at the deployment end? If it is given at deployment, it can be included provided we know in which format is it given.
5. The model is still an over fit and doing steps 1-4 of improvement can be helpful in reducing the same.