Credit Card Number Plugin Documentation

Overview

The credit card number plugin analyzes a database column to calculate a confidence score for how likely it is the column contains credit card numbers. The plugin relies on two main checks to determine whether a column element is a credit card number, whether the column passes the Luhn algorithm check, and if it matches an expected length based on predefined credit card brands.

Dependencies

re - regular expressions module

Function Definitions

get_confidence_score()

The get_confidence_score() function takes in a *string* called col_name and a list of *strings* called *col_vals* and returns a *double*. Any null values within *col_vals* are removed from the column, with the remaining values being stripped of any delimiters typically found in credit card numbers (including any spaces). Scores for each element of the column are then calculated with the *get_elem_score()* function and an average for all scores is calculated and returned as a *double*. Columns with names that contain “ccn,” “credit,” or “card” receive a boost to their scores.

get_elem_score()

The *get_elem_score()* function takes in a *string* called *elem* and returns a *double*. *elem* is presumed to be an element from the dataset column. Any input which is not a digit is assumed to not be a credit card number because the function assumes all typical delimiters have been removed from *elem*. The function returns a score based on two checks run on the input, *check_valid_card_number()* and *matches_expected_length()*.

matches_expected_length()

The *is_expected_length()* function takes in a *string* called *elem* and returns a *boolean* which indicates whether the number is the expected length based on the fact that the length of a credit card number varies depending on what brand of credit card it is.

check_valid_card_number()

The *check_valid_card_number()* function takes in a *string* called *elem* and returns a *boolean*. The function checks whether a credit card number is valid. Since Luhn's algorithm can be applied to credit card numbers to check whether it is a valid number all credit cards should be able to be validated this way.

remove_delimiters()

The *remove_delimiters()* function removes any delimiters that would typically be found as part of a credit card number as well as leading and trailing whitespace for all elements of the dataset column.

remove_null()

The remove_null() function takes in a list of strings called col_vals. col_vals is presumed to be a dataset column that has been converted to a list of strings. The function uses list comprehension to return a list of where any None values have been removed and where any strings denoting a null value have been removed. The list of strings that denote values are noted as ['NA', 'N/A', 'na', 'n/a', 'Na', 'N/a'].

Implications for the future

Since credit card information is sensitive information, we would like to add a flag that denotes this specific data type as sensitive.