

CS 215 Project

Chandra Maloo

130050009

chandra@cse.iitb.ac.in

Utkarsh Kumar

130050022

utkarsk@cse.iitb.ac.in

Dibyendu Mondal

130050046

dibyendu@cse.iitb.ac.in

November 16, 2014

Working

1.Introduction

Working

- 1.Introduction
- 2.Working

Working

- 1.Introduction
- 2.Working
- 3.Conclusion

Introduction

This report is created as a part of our CS 215 project to easily convey the internal working of the project.

In the coming slides, you will find a detailed description of the project.

Happy Reading!

Working

- 1 Defined classes 'Attribute', 'Country' and 'Sentence' along with their data members and data functions to help organise and simplify the code.

Working

- 1 Defined classes 'Attribute', 'Country' and 'Sentence' along with their data members and data functions to help organise and simplify the code.
- 2 Defined three dictionaries (which are basically maps):

Working

- ➊ Defined classes 'Attribute', 'Country' and 'Sentence' along with their data members and data functions to help organise and simplify the code.
- ➋ Defined three dictionaries (which are basically maps):
 - 2.1 cc : A dictionary that maps a country code to the corresponding index number, in the order in which they are received the first time.

Working

- ❶ Defined classes 'Attribute', 'Country' and 'Sentence' along with their data members and data functions to help organise and simplify the code.
- ❷ Defined three dictionaries (which are basically maps):
 - 2.1 cc : A dictionary that maps a country code to the corresponding index number, in the order in which they are received the first time.
 - 2.2 cn : A dictionary that maps a country name to the corresponding index number, in the order in which they are received the first time.

Working

- ❶ Defined classes 'Attribute', 'Country' and 'Sentence' along with their data members and data functions to help organise and simplify the code.
- ❷ Defined three dictionaries (which are basically maps):
 - 2.1 cc : A dictionary that maps a country code to the corresponding index number, in the order in which they are received the first time.
 - 2.2 cn : A dictionary that maps a country name to the corresponding index number, in the order in which they are received the first time.
 - 2.3 p : A dictionary that maps an attribute to the corresponding index number, in the order in which they are received the first time.

Working

- ❶ Defined classes 'Attribute', 'Country' and 'Sentence' along with their data members and data functions to help organise and simplify the code.
- ❷ Defined three dictionaries (which are basically maps):
 - 2.1 cc : A dictionary that maps a country code to the corresponding index number, in the order in which they are received the first time.
 - 2.2 cn : A dictionary that maps a country name to the corresponding index number, in the order in which they are received the first time.
 - 2.3 p : A dictionary that maps an attribute to the corresponding index number, in the order in which they are received the first time.

Working

- 1 On running the code, main function is called by default, which:

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads from "countries_idmap.txt" and populates 'country_array'

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads from "countries_idmap.txt" and populates 'country_array'
 - 1.2 Reads from "selected_indicators" and populates "attributes_array_main"

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads from "countries_idmap.txt" and populates 'country_array'
 - 1.2 Reads from "selected_indicators" and populates "attributes_array_main"
 - 1.3 Runs a 'for' loop that adds this attribute-list to each of the countries already listed above.

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads from "countries_idmap.txt" and populates 'country_array'
 - 1.2 Reads from "selected_indicators" and populates "attributes_array_main"
 - 1.3 Runs a 'for' loop that adds this attribute-list to each of the countries already listed above.
 - 1.4 Reads from "kb-facts-train_Sl.csv" and fills up the attribute-list for each country using this data.

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads from "countries_idmap.txt" and populates 'country_array'
 - 1.2 Reads from "selected_indicators" and populates "attributes_array_main"
 - 1.3 Runs a 'for' loop that adds this attribute-list to each of the countries already listed above.
 - 1.4 Reads from "kb-facts-train_Sl.csv" and fills up the attribute-list for each country using this data.
 - 1.5 Calls regression() function for each of the countries, which:

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads from "countries_idmap.txt" and populates 'country_array'
 - 1.2 Reads from "selected_indicators" and populates "attributes_array_main"
 - 1.3 Runs a 'for' loop that adds this attribute-list to each of the countries already listed above.
 - 1.4 Reads from "kb-facts-train_Sl.csv" and fills up the attribute-list for each country using this data.
 - 1.5 Calls regression() function for each of the countries, which:
 - 1.5.1 Finds the sample mean and variance for each attribute of the country

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads from "countries_idmap.txt" and populates 'country_array'
 - 1.2 Reads from "selected_indicators" and populates "attributes_array_main"
 - 1.3 Runs a 'for' loop that adds this attribute-list to each of the countries already listed above.
 - 1.4 Reads from "kb-facts-train_Sl.csv" and fills up the attribute-list for each country using this data.
 - 1.5 Calls regression() function for each of the countries, which:
 - 1.5.1 Finds the sample mean and variance for each attribute of the country
 - 1.5.2 Uses linear regression and finds the regression coefficients

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads from "countries_idmap.txt" and populates 'country_array'
 - 1.2 Reads from "selected_indicators" and populates "attributes_array_main"
 - 1.3 Runs a 'for' loop that adds this attribute-list to each of the countries already listed above.
 - 1.4 Reads from "kb-facts-train_Sl.csv" and fills up the attribute-list for each country using this data.
 - 1.5 Calls regression() function for each of the countries, which:
 - 1.5.1 Finds the sample mean and variance for each attribute of the country
 - 1.5.2 Uses linear regression and finds the regression coefficients

Working

- 1 On running the code, main function is called by default, which:

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads "sentences.tsv" and converts each row of input to a corresponding Sentence class object and calls the function doAll() on that Sentence object.

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads "sentences.tsv" and converts each row of input to a corresponding Sentence class object and calls the function doAll() on that Sentence object.
 - 1.1.1 The doAll() function uses the country-name passed with the Sentence object and goes on removing the last letter until it finds a match with one of the countries in the country_array list (or the country-name becomes empty)

Working

- ❶ On running the code, main function is called by default, which:
 - 1.1 Reads "sentences.tsv" and converts each row of input to a corresponding Sentence class object and calls the function doAll() on that Sentence object.
 - 1.1.1 The doAll() function uses the country-name passed with the Sentence object and goes on removing the last letter until it finds a match with one of the countries in the country_array list (or the country-name becomes empty)
 - 1.1.2 For calculation of score, the function uses the normlised exponential formula

$$score = 100 * \exp - (x - u)^2 / (2(s^2)) / (\sqrt{2 * \pi} * s),$$

s being the standard deviation of the sample values of attribute in consideration.

Conclusions

This is our project as a part of the CS 215 course. Hope you enjoyed going through the report as much as we enjoyed making it. Any other reviews and/or suggestions are most welcome. Looking forward to hearing from you. Thank You :)