

応用心理学Ⅰ 課題小レポート

課題について

- 自分の班のインタビュー結果を対象に分析を実施し、結果をレポートとしてまとめ提出する
 - 分析内容：
 - 必須課題
 - 選択課題からひとつ以上選択
- 現実のデータなので結果がきれいにとは限らない。できたところまでで構わないので結果を報告すること
- 必須課題・選択課題いずれの課題においてもインタビュアーの発言は除外して分析すること

必須課題

- 登場した名詞の頻度表を作成する
 - 出現回数上位20件について図表化する（Excelなどを用いても可）
 - インタビューの情報を抽出できているか議論する

選択課題

下記のいずれかの課題を選択し、実施する（複数可）

- ①複数の班のデータを使って分析
 - 2つ以上の班のデータを使って分析・比較する
 - 分析の例：
 - それぞれのデータごとに単語の頻度を求め、上位の単語を抽出する
 - TF-IDFを算出し、キーワードを抽出する
- ②単語の共起関係を分析する
 - 単語の共起関係(バイグラム, bag of wordsなど)を使った分析を行う
 - 分析に使う品詞・出現回数の閾値については分析しながら自分で決めること
 - 実習では詳細に説明する時間がなかったが、授業で紹介した分析結果(e.g. ネットワークグラフ, 主成分分析)のコードはGitHub上ですべて公開してある(<https://github.com/satocos135/lecture2022shimane>)ので参考にしてもよい
- ③タグを使った分析
 - インタビュー入力時につけたタグとテキストマイニングの手法を組み合わせる分析する
- ④その他
 - 授業で紹介した手法、あるいは自分で調べた手法を自由に用いてインタビューデータを分析する
 - 計量的手法であること

いずれも、情報がうまく抽出できたかどうかについて議論すること

レポートについて

以下の形式で記述する

- 目的と方法
 - 分析の目的とデータのクリーニング・前処理の方法について記述する（引用文献の参照は不要）
 - 今回は小レポートなので、目的は数行でOK（序論のように大量に書く必要はない）
 - ストップワード・辞書を設定した場合はその内容についてレポート内で言及すること
 - 方言や表記ゆれなどにどう対応したか（あるいはしなかったか）の記述を含めること
- 結果
 - 課題について分析した結果を記述する
 - 選択課題については選択した内容についても記述すること
- 議論
 - 全体を振り返り、情報が抽出できたかについて議論する
 - 特に、うまくいかなかった場合は「なぜうまくいかなかったか」について考察すること

提出について

以下のファイルをまとめたzipファイルを提出する

- レポート本文
- データ
 - 生データ
- 辞書ファイル（使った場合）
 - コンパイル後のファイルと、コンパイル前の辞書csv
- 分析に用いたプログラムのソースコード
 - ソースコードの具体例についてはreport_exampleフォルダ内を参照のこと

ファイル名は 学生番号_氏名.zip の形式にすること

提出先： `sato.cos135@gmail.com`

- メール件名に「応用心理学 I レポート」という文字列を含めること
- CCに佐藤鮎美先生(`ayumisato@hmn.shimane-u.ac.jp`)を加えること

※提出24時間以内に受領した旨の返信が来ない場合は届いていない可能性があるので確認すること

提出期限： 7/25(月) 24:00

課題のヒント

クリーニング

以下の事実に留意する

- 残された時間ですべてのデータを完璧にクリーニングすることは不可能である
 - かりうじて 1 班分のデータぐらいなら人手でも可能かもしれない（試してみても構わない）
 - データが10倍、100倍あっても同じ方法をとるか（とりうるか）を考える
- データの大部分は分析と関係がない
 - とはいえ事前にどこが重要か知るのも難しい
 - Zipfの法則/Paretoの法則を思い出す
- データのクリーニングそのものは本来の目的ではない
 - あくまで分析が主目的であることを忘れない
 - 「無難なアプローチ」を思い出す

前処理

特定のデータを除外する

filter()を使ってそれ以外のものを抽出する

```
library('dplyr')
ToothGrowth %>% filter(supp != 'OJ') # supp列の値がOJではないものを抽出する
```

括弧の中身を除外する

str_replace()を使う

```
library('stringr')
text = 'この授業は「応用心理学 I : データ収集後探索的解析（テキストマイニング）」です'
text %>% str_replace('( [^ ]* )', '') # 正規表現にマッチした部分を除去する
# [^ ]* : 閉じ括弧以外の任意の文字の連続
```