

RとQuartoではじめるデータサイエンス《2026》

#8 プレゼンテーションとループ処理

荻谷千尋

Wed, 3, Jun, 2026

目次

1. 前回の振り返り
2. プレゼンテーション
3. グループワーク
4. ループ処理
5. レポート課題の相談

0. 本日の目標

1. プレゼンテーションを頑張る
2. 他の受講生のよい点を発見する（できれば真似する）
3. 授業を振り返り、自分ができるようになったことを確認する
4. （時間に余裕がある人は）ループ処理に挑戦する

2. プレゼンテーション

3つのポイント

- 分析の目的を伝える
 - 何を知りたくて分析したのかを説明する
- グラフの読み取り
 - グラフの内容を説明するだけでなく、「何がわかったのか」を中心に伝える
- 結論につなげる
 - 図ごとの説明で終わらず、目的を意識した上で、全体として何が明らかになったのかを説明する

3. グループディスカッション

テーマ1. 他のプレゼンテーションから学んだこと

- 印象に残ったプレゼンテーションはどれでしたか？
- どのような点がわかりやすかったですか？
- 自分も真似したいと思った工夫はありましたか？

テーマ2. この授業でできるようになったこと

- この授業を通して、新しくできるようになったことは何ですか？
- データ分析やグラフ作成に対する理解はどのように変わりましたか？
- 次に挑戦してみたい分析や可視化はありますか？

4. ループ処理

map() 関数

- purrr (「パー」と発音) パッケージに含まれる
 - purrrはtidyverseに含まれているので、追加インストール不要
- ベクトルやリストの要素を順に取り出し、同じ処理を行う

主な使い方

- 指定した列名ごとに同じ処理を繰り返す
 - 例: body_mass, bill_len, flipper_len
- グループごとにデータを分けて同じ処理を繰り返す
 - 例: スポーツ種目ごとにグラフを作成
 - split() や nest() を利用

利点

- コードが短くなり、ミスを減らせる
- 多くの変数やグループを効率よく分析できる
- 特定の結果だけでなく、データ全体を網羅的に探索できる
 - 自分の興味のあるデータだけを恣意的に選ぶことを防ぐ

vars() 関数

- 指定した列名ごとに同じ処理を繰り返す

nest() 関数

- グループごとのデータを表として管理する
- 計算結果やグラフを列として追加できる

split() 関数

- データをグループごとに分けて同じ処理を繰り返す
- データフレームをグループごとに分割する
- 分割したデータに対して同じ処理を実行する
- グラフの一括作成やファイルの連続出力に向く

walk() 関数

- map() と同様に繰り返し処理を行うための関数
- 計算結果を利用するよりも、処理の実行を目的とする
- グラフの表示やファイル出力でよく使う

宿題

授業の感想

- 回答先：Google Forms
- 締め切り：6月7日（日）23時59分

学期末授業アンケート

- 回答先：Google Forms
- 締め切り：6月7日（日）23時59分

レポート課題（再提出したい人のみ）

- ファイル：ZIPファイル
 - ファイル名：どこかに氏名を特定できる文字を入れて下さい
 - 例：kariya.zip
- 提出先：dropbox
- 締め切り：6月7日（日）23時59分（延長しています）