

訳者まえがき

本書は、“*Interactive Web-Based Data Visualization with R, plotly, and shiny*”の翻訳書であり、**plotly** パッケージを用いて R からインタラクティブな可視化を行うための入門書である。R は世界中のユーザが多様なパッケージを公開しており、そのおかげで様々なデータに対して多岐にわたる分析を行うことができ、かつ分析フローのすべてをカバーする汎用性の高い言語である。R からインタラクティブな可視化が行えるようになれば、データ分析のワークフローは大きく変わり、より高速に深くデータと対話できるようになる。つまり、インタラクティブな可視化は、データ分析の至るところで分析の質や早さを向上させる力を持っている。

たとえば、探索的なデータ分析やデータ理解の段階では、インタラクティブなグラフであれば、グラフを通してデータの分布や細かな数値の確認、セグメンテーションが行えるようになる。仮説をもとにデータ処理と可視化を繰り返す既存のワークフローは、単一のインタラクティブなグラフだけで完結できるものとなる。分析はより高速になり、そして、より気軽に仮説検証や数値確認が行える環境は、より深いデータ理解に繋がる。

さらに、データ分析者は結果の共有も求められる。R では **rmarkdown** や **shiny** パッケージを使って結果を共有することが多いだろう。**plotly** で作成したインタラクティブなグラフは、この両者とも連携することもできるし、もちろん単独でも共有できる。インタラクティブなグラフを共有することができれば、受け取った人はグラフからデータを探索できるようになる。データ分析者は共有結果に関する仲間からの問い合わせから解放されるだろう。分析の質という観点では、他者の仮説検証を促進させる効果もある。共有を例にとっても、データ分析の質と速さの向上に繋がることがわかるだろう。

このようなメリットがある **plotly** は、R ユーザであれば容易に利用できる点もまたメリットである。**ggplot2** ユーザであれば、**ggplot2** のグラフをインタラクティブな **plotly** グラフに簡単に変換することができる。簡単ゆえに限界もあるが、本書ではその限界やカスタマイズを行う方法を紹介している。**plotly** パッケージのみを使って可視化を行う場合でも、シンタックスが **ggplot2** と似ており、基本的なところは理解しやすいはずである。発展的な内容として、5 章では複数のグラフを連携させてインタラクティブな操作を実現させる方法を扱っている。

本書では、副題にあるように **shiny** と **plotly** を連携させ、より高度でインタラクティブ性の高いアプリケーションを構築する方法についても紙面を割いて扱っている。たとえば、本書ではグラフを通してドリルダウンやクロスフィルタが可能なダッシュボードの作成方法を紹介

している。plotly と shiny の連携方法以外にも、連携させたアプリを作成するために必要となる汎用的な実装テクニックも扱っている。高度なアプリケーションの実装にチャレンジしたい shiny ユーザにも一読いただきたい。なお、shiny の基礎的な内容は簡単には説明しているものの、shiny の入門書ではない点は補足しておく。

本書の原著はウェブ上で公開されており、各章のコードや操作例なども確認することができる。適宜参照してほしい。

<https://plotly-r.com/>

なお、翻訳にあたって一部コードを原著から修正している箇所がある。修正したコードは以下の訳者の GitHub リポジトリ上で公開している。本書に記載されているコードを参照する場合は問題ないが、原著ウェブサイトから参照する場合は、適宜確認してほしい。

https://github.com/t-kosshi/r-plotly_and-shiny_book_ja

コードに修正を行ったケースは、主に可視化のために必要なパッケージのバージョンアップによるところが大きい。可視化は様々な分析やデータと連携することで価値を発揮することから、本書では多数のパッケージを必要としている。たとえば、データのダウンロードにもいくつかのパッケージを利用している。地理空間データの可視化には地理空間データを扱えるパッケージが必要である。ggplot2 を拡張するパッケージとの連携も紹介している。それゆえに、多数のパッケージをインストールする必要があるが、R のバージョンや OS によってはインストールに戸惑うかもしれない。1 章で必要なパッケージをまとめてインストールする方法を紹介しているが、一度にやろうとすると躓いてしまうかもしれない。サポートサイトで各章ごとに必要なパッケージをインストールするコードをまとめているので、必要な場合は参照して少しずつ本書を読み進めていってほしい。

2022 年 4 月

興石 拓真（訳者を代表して）

目次

1 導入	1
1.1 なぜ R とインタラクティブなグラフなのか	1
1.2 本書で学ぶこと	5
1.3 本書で学ばないこと	9
1.3.1 ウェブ技術	9
1.3.2 d3js	9
1.3.3 ggplot2	10
1.3.4 可視化によるデータ分析	10
1.3.5 データ可視化のベストプラクティス	10
1.4 読み進めるうえでの前提条件	10
1.5 サンプルコードの実行環境	10
1.6 困った時やさらに学習したい時	11
1.7 謝辞	12
1.8 本書のデータに関する情報	13
第 I 部 グラフの作成	15
2 概要	16
2.1 plotly() の紹介	16
2.2 plotly.js の紹介	23
2.3 ggplotly() の紹介	26
3 グラフ作成の基礎	33
3.1 マーカー（散布図）	37
3.1.1 アルファブレンディング（透明度）	37
3.1.2 カラー（色）	38
3.1.3 シンボル（形状）	41
3.1.4 輪郭線（ストローク）とその幅（スパン）	44
3.1.5 サイズ	44
3.1.6 ドットプロットとエラーバー	46

3.2	折れ線グラフ	47
3.2.1	線の種類	47
3.2.2	セグメント	49
3.2.3	密度プロット	53
3.2.4	平行座標プロット	54
3.3	ポリゴン	56
3.3.1	リボン	58
4	地図	59
4.1	標準マップ	59
4.1.1	概要	59
4.1.2	コロプレス図	65
4.2	カスタムマップ	68
4.2.1	sf パッケージ	68
4.2.2	カートグラム	73
5	棒グラフとヒストグラム	77
5.1	複数の数値変数の分布	79
5.2	複数の離散変数の分布	80
6	箱ひげ図	83
7	2次元ヒストグラム	86
7.1	plotly.js でビン計算を行うアプローチ	86
7.2	R でビン計算を行うアプローチ	87
7.3	カテゴリ変数軸	88
8	3次元グラフ	90
8.1	マーカー（散布図）	90
8.2	パス（データの並び順に点を繋げる）	91
8.3	ライン（ x 軸の値順に点を繋げる）	91
8.4	軸の編集	93
8.5	曲面	94

第 II 部 グラフのパブリッシュ	95
9 はじめに	96
10 HTML への保存・埋め込み	97
11 静止画の保存	99
11.1 コードによる静止画の保存	99
11.2 ブラウザを使った静止画の保存	100
11.3 静止画のサイズ変更	101
12 静止画とその保存形式の編集	102
第 III 部 複数のグラフを用いた表現	107
13 複数グラフの配置	108
13.1 plotly オブジェクトの配置	108
13.1.1 subplot の再帰的な結合	111
13.1.2 その他のアプローチと subplot の応用	113
13.2 htmlwidgets の配置	118
13.2.1 flexdashboard	118
13.2.2 Bootstrap のグリッドレイアウト	119
13.2.3 CSS フレックスボックス	120
13.3 グループ数の多いスモールマルチプルへの対応	122
14 アニメーショングラフ	124
14.1 アニメーション API	124
14.2 アニメーション化が可能なグラフ	127
第 IV 部 グラフの連携	131
15 はじめに	132
16 クライアントサイドでのグラフの連携	134
16.1 グラフィカルクエリ	134
16.2 ハイライト vs フィルタ	141

16.3	アニメーショングラフの連携	145
16.4	グラフィカルクエリの実践例	148
16.4.1	スモールマルチプル	148
16.4.2	統計的トレースタイプの活用	151
16.4.3	地理空間データ	156
16.4.4	他の <code>htmlwidgets</code> との連携	159
16.4.5	一般化ベアプロット	162
16.4.6	診断プロット	162
16.4.7	リスト列をキーにしたグラフィカルクエリ	164
16.5	グラフィカルクエリの限界	167
17	shiny によるサーバーサイドを使ったグラフの連携	168
17.1	shiny に plotly を組み込もう	169
17.1.1	shiny の基礎	169
17.1.2	<code>renderPlotly()</code> の詳細 (非表示, サイズ調整など)	172
17.2	plotly 入カイベントの活用	173
17.2.1	ドラッグイベント	174
17.2.2	3次元グラフでのイベント	175
17.2.3	編集イベント	176
17.2.4	レイアウトとトレースを変更するイベント	181
17.2.5	イベントソースの特定	183
17.2.6	イベントによるリアクティブ式の再実行条件	185
17.2.7	離散的な軸への対応	187
17.2.8	イベントから得たデータの保持と管理	188
17.3	アプリのパフォーマンス改善	191
17.3.1	plotly グラフの部分的な変更	192
17.3.2	部分的な変更の活用例	195
17.4	shiny を使った発展的なアプリケーション	199
17.4.1	ドリルダウンの実装	199
17.4.2	クロスフィルタの実装	210
17.4.3	ドラッグ可能な範囲選択	217
17.5	終わりに	217

第 V 部	JavaScript によるイベントハンドリング	221
18	はじめに	222
19	JSON の基礎	225
19.1	JavaScript での代入・抽出・一括処理	225
19.2	R オブジェクトから JSON への変換	227
20	イベントハンドラを自作する	230
21	customdata 属性の活用	232
22	ウェブ技術の活用	244
22.1	基本的なウェブ技術の利用	244
22.2	モダンな JavaScript と React の活用	246
第 VI 部	その他のトピック	253
23	plotly は無料かつ安全に使えるのか？	254
24	パフォーマンスを向上させる	255
25	ツールチップの制御	257
25.1	plot_ly() のツールチップ	257
25.2	ggplotly() のツールチップ	261
25.3	ツールチップのスタイルを変更	265
26	モードバーの制御	267
26.1	モードバー全体の削除	267
26.2	plotly ロゴの削除	267
26.3	指定したボタンの削除	268
26.4	カスタムボタンの追加	268
26.5	指定したボタンのみの表示	269
26.6	静止画ダウンロードボタンの詳細設定	269

27	色の指定方法	271
28	マーカーの形状とグリフ	273
29	グラフに画像を埋め込む	276
30	言語の切り替え	279
31	LaTeX 形式の描画	281
31.1	MathJax の注意点	283
32	パイプ演算子による処理フローの拡張	284
33	ggplotly() の改善	289
33.1	レイアウトの変更	290
33.2	データ (トレース) の変更	291
33.3	集計後データの活用	293
34	カスタム geom_*() の plotly での利用	297
	参考文献	301
	索引	309