TEST (2025) 34:488–489
https://doi.org/10.1007/s11749-025-00964-y

DISCUSSION

Comments on: Exploratory functional data analysis

Jeff Goldsmith[1]

Received: 7 January 2025 / Accepted: 12 January 2025 / Published online: 5 February 2025

Abstract
Exploratory FDA is a critical step in understanding functional data. In this article, the authors present a range of compelling real-data examples to illustrate best practices in visualization and initial analyses. My comments include suggestions to incorporate covariates in exploratory approaches and note a new software package to facilitate these techniques.

Keywords Functional data · Visualization · Software

Many commentaries—probably most, and perhaps all—follow a familiar structure: they begin by enumerating points of agreement, add some complementary suggestions, and close with allusions to the commentor's own work. This will not be an exception to that format.

It is my pleasure to say that I have many points of agreement with the authors and find a great deal to praise in this clear review of exploratory functional data analysis (EFDA). The authors' selection of example datasets illustrates the breadth of functional data types and the scientific settings in which they arise. This includes data observed over sparse and dense grids, which can be the same or differing across study units, as well as univariate and multivariate functional data. Visualizations, as the authors show beginning with Fig. 1 and continuing in more detail in Section 3, are a critical tool in EFDA. Indeed, in my experience, visualization builds intuition for data generating mechanisms in FDA much more easily than in other settings with high-dimensional data. With the right plot, it is easy to understand the path of a hurricane or to imagine your own hip and knee angles during a gait cycle. The intuition this provides makes it possible to move from EFDA to more formal analyses using plausible model structures.

Using depth-based approaches to add summary statistics facilitates a nuanced understanding of functional datasets. Identifying representative curves from the center out enables both rainbow plots, where the spectrum of depths can be represented through color, and boxplots, in which median curves, central regions, and outliers can be identified visually. Clustering methods, meanwhile, serve to uncover latent but interpretable subgroups from observed curves. Several methods for functional clus-

✉ Jeff Goldsmith
ajg2202@cumc.columbia.edu

1 Columbia University, New York, USA

```
chf_df %>%
  group_by(day) %>%
  summarize(mean_act = mean(activity)) %>%
  mutate(smooth_mean = tf_smooth(mean_act)) %>%
  ggplot(aes(tf = mean_act, color = day)) +
  geom_spaghetti(alpha = .2) +
  geom_spaghetti(aes(tf = smooth_mean), size = 2)
```

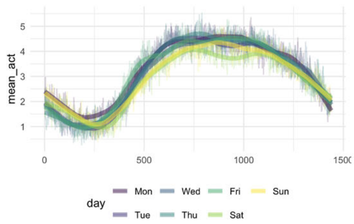

Fig. 1 Example of `tidyfun` code and output

tering are reviewed here, together with the good advice to select the method based on the research goal; it also helps to provide an illustration of how the selected method can affect the result using an accessible dataset on hurricane path.

My suggestion for an EFDA method that complements those presented by the authors is the incorporation of additional (non-functional) variables in visualizations. For continuous variables, the result can be similar to a rainbow plot; for categorical variables, using color or separate panels can be effective. Adding this information directly facilitates comparisons as continuous covariates change, or across groups. For example, we often use visualizations like these to begin to understand how physical activity behaviors measured continuously over 24 h using wearable devices are affected by age or gender. Assessing the statistical significance of any observed differences, however, can involve regression models that are beyond the scope of any review focused on EFDA.

Lastly, I will take the liberty to note that the R package `tidyfun` can make many kinds of EFDA quick and easy (Scheipl et al. 2024). The conceptual innovation in this package is that FDA considers complete functions as the basic unit of observation, and so complete functions should be stored as single observations in data frames. After developing infrastructure to allow this for a range of real-data settings, `tidyfun` then allows `tidyverse`-influenced operations (like grouping and summarizing) as well as pasta-themed plots through `ggplot`. Figure 1 shows both an example code chunk and the resulting plot.

My sincere congratulations to the authors of this interesting and timely review of EFDA.

References

Scheipl F, Goldsmith J, Wrobel J (2024) tidyfun: tools for tidy functional data. R package version 0.0.98 https://tidyfun.github.io/tidyfun/