Dynamic Prediction of Non-Guassian Outcome with fast Generalized Functional Principal Analysis

Ying Jin Andrew Leroux

March 15, 2023

Introduction

Biomedical investigators are often interested in predicting future observations of subjects based on their historical data, typically referred to as dynamic prediction. Traditionally, this type of data has been modeled using marginal models (generalized estimating equations) or conditional models (mixed effect models) (Laird and Ware 1982; LIANG and ZEGER 1986; Lindstrom and Bates 1990; "Nonlinear models for repeated measurement data" 2003), and predictions are made based on the correlation between repeated measures from the same subject, and/or covariates that can be either fixed or time-varying. However, these methods are limited in terms of flexibility of correlation structure and the ability to handle out-of-sample prediction. When sample size is large or the density of repeated measures is high, they also tend to cause severe computational burden (Rizopoulos 2022).

To address these problems, one may turn to functional mixed effect models instead when measures are dense across the domain. Such methods accommodate more flexible correlation structure by modeling subject-specific random effects as a function, and non-parametric smoothing (Scheipl et al. 2014) can be incorporated to speed up the computation, such as spline basis functions or eigenfunctions from functional principal component analysis (fPCA). The introduction of basis functions also makes out-of-sample prediction more straightforward. Instead of estimating subject-specific random effects of new observations, we can simply estimate coefficients/loadings on the basis function used for smoothing. Existing research on dynamic prediction with functional data analysis methods has been focusing on continuous/Gaussian outcomes, modelling subject-specific random effects with FPCA (Chiou 2012; Goldberg et al. 2014; Shang 2017). Kraus (2015) has used this approach to predict missing observations in partially observed function tracks, and Delaigle and Hall (2016) achieved similar goals using Markov Chains. While methods mentioned above used only partial observations for prediction with an intercept-only model, Leroux et al. (2018) proposed Functional Concurrent Regression (FCR) framework which can incorporate the effect of subject-specific predictors. However, little extension was made on prediction of non-Gaussian functions, such as binary and count outcomes.

Unfortunately, fewer papers have focused on its extension to non-Gaussian data, such as series of binary or count outcomes. Existing methods also tend to be very computationally intensive. For example, Chen et al. (2013) proposed approaches to fit marginal functional models that is compatible to multi-level, generalized outcomes. Goldsmith et al. (2015) established a model framework that takes into account the fixed effect of time-invariant covariates, with parameters estimated with Bayesian method in *Stan*. Gertheiss et al. (2016) identified bias introduced by directly applying FPCA methods to generalized functions, and proposed to address this problem using a two-stage, joint estimation strategy. Linde (2009) used an adapted Bayesian variational algorithm for FPCA of binary and count data. In terms of implementation, Wrobel et al. (2019) proposed a fast, efficient way to fit GFPCA on binary data using EM algorithm, accompanied by the an open source R package *registr*.

In this paper, we aim to develop a fast, scalable method for dynamic prediction of discrete function tracks based on functional mixed effect model with fPCA smoothing. Section 2 presents the procedure of the proposed method. In Section 3, we illustrate the performance and efficiency of our proposed method in a simulation study. In Section 4, we apply this method to a real-world dataset. Section 5 presents a discussion of davantages and limitation of the proposed method.

Method

The observed data for a single subject i is (t, $Y_i(t)$), where t consists of dense, discrete points along the functional domain, and $Y_i(t)$ is the non-Gaussian outcome observed at t. We assume that the outcome $Y_i(t)$ can be characterized by a latent continuous function $\eta_i(t)$. That is, at a specific t, $Y_i(t)$ follows a exponential family distribution such that:

$$g[E(Y_i(t))] = \eta_i(t) = \beta_0(t) + b_i(t)$$

where g is a appropriate link function, $\beta_0(t)$ is the population mean of latent function, and $b_i(t)$ is a subjects-level random effect function follows a zero-mean Gaussian process.

While $b_i(t)$ is not observed, it can be approximated under the FPCA framework $b_i(t) = \sum_{k=1}^{K} \xi_{ik}\phi_{ik}(t) + \epsilon_i(t)$. Here $\phi_{ik}(t)$ is a set of orthogonal eigenfunctions that explains the most variation in $b_i(t)$, and ξ_{ik} are subject-specific PC scores (or loadings) on each eigenfunction. Additionally, ξ_{ik} are mutually independent obver both subject (i) and eigenfunctions (k). That is, each ξ_{ik} follows normal distribution $N(0, \lambda_k)$ where λ_k is the kth eigenvalues: $\int \phi_k^2(t)dt = \lambda_k$. $\epsilon_i(t)$ here is a residual function that accounts for the variation not explained by the first K eigenfunctions from FPCA. We assume it follows a zero-mean Gaussian process. At a specific point t, $\epsilon_i(t) \sim N(0, \sigma^2)$.

Based on the problem set up above, we propose the following algorithm for PFCA on the unobserved latent process $\eta_i(t)$:

1. Bin the observed outcomes in to small, non-overlapping, equal length intervals. We hereafter index the bins by their midpoints s.
2. Fit a local Generalized Mixed Model at every bin. Specifically, at bin s, we fit $g[E(Y_i(t))] = \beta_0(s) + b_i(s)$ for all t in bin s. From this series of models we can get estiamtes of population mean $\hat{\beta}_0(s)$ and subject-level random effect $\hat{b}_i(s)$, thus estimate of the individual latent functions at every bin: $\hat{\eta}_i(s) = \hat{\beta}_0(s) + \hat{b}_i(s)$.
3. Fit FPCA on the estimated latent functions $\hat{\eta}_i(s)$, and obtain estimates of basis functions $\mathbf{\Phi} = \{\phi_1(s), ..., \phi_k(s)\}$, eigenvalues $\hat{\lambda}_1 ... \hat{\lambda}_k$, population mean $\hat{\beta}_0(s)$ and residual variance $\hat{\sigma}^2$.

4. With components extracted above, calculate the maximum likelihood estimate (MLE) of the subject-specific PC scores $\hat{\xi}_{ik}$ of new samples based on their partially observed data. Then the value of latent functions at unobserved points can be estimated as $\hat{\eta}_i(s) = \hat{\beta}_0(s) + \sum_{k=1}^{K} \hat{\xi}_{ik}\phi_k(s)$

Following the algorithm above, predictions of individual latent functions are made on the binned grid based on partially observed non-Gaussian functions tracks of new subjects. Since the bins are set up to be small in length, the binned grid would still be dense enough. However, in situations where we need predictions on the original, un-binned grid instead, linear interpolation turns out to be a fast, convenient way with good performance for prediction at points between bins.

Precision of prediction, usually quantified by the variance of prediction error $Var(\hat{\eta}_i(s) - \eta_i(s))$, is also straightforward under this framework. In step 4 we calculated the MLE of $\hat{\xi}_{ik}$. Based on likelihood theory, its variance can be estimated with observed information $I(\hat{\xi}_{ik})$, which is essentially the second derivative of likelihood at $\hat{\xi}_{ik}$. Therefore, the variation of prediction interval is:

$$Var(\hat{\eta}_i(s) - \eta_i(s)) = \mathbf{\Phi}(s) I(\hat{\boldsymbol{\xi}}_i) \mathbf{\Phi}^T(s) + \hat{\sigma_\epsilon}^2$$

Where $\mathbf{\Phi}(s) = (\phi_1(s)...\phi_K(s))$ and $\hat{\boldsymbol{\xi}}_i = (\hat{\xi}_{i1}, ..., \hat{\xi}_{iK})$.

Simulation

In this section, we illustrate the predictive performance and computational efficiency of the proposed method through a simulation study. We simulated 50 datasets, each with 500 subjects. For every subject, we generate 1000 binary outcomes $Y_i(t) \in (0, 1)$ across functional domain $t \in [0, 1]$, where the distribution of outcome is characterized by a continuous latent function. The data generation mechanism can be expressed as follows:

$$Y_i(t) \sim Binomial(\frac{exp(\eta_i(t))}{1 + exp(\eta_i(t))})$$
$$\eta_i(t) = f_0(t) + \xi_{i1} sin(2\pi t) + \xi_{i2} cos(2\pi t) + \xi_{i3} sin(4\pi t) + \xi_{i4} cos(4\pi t)$$

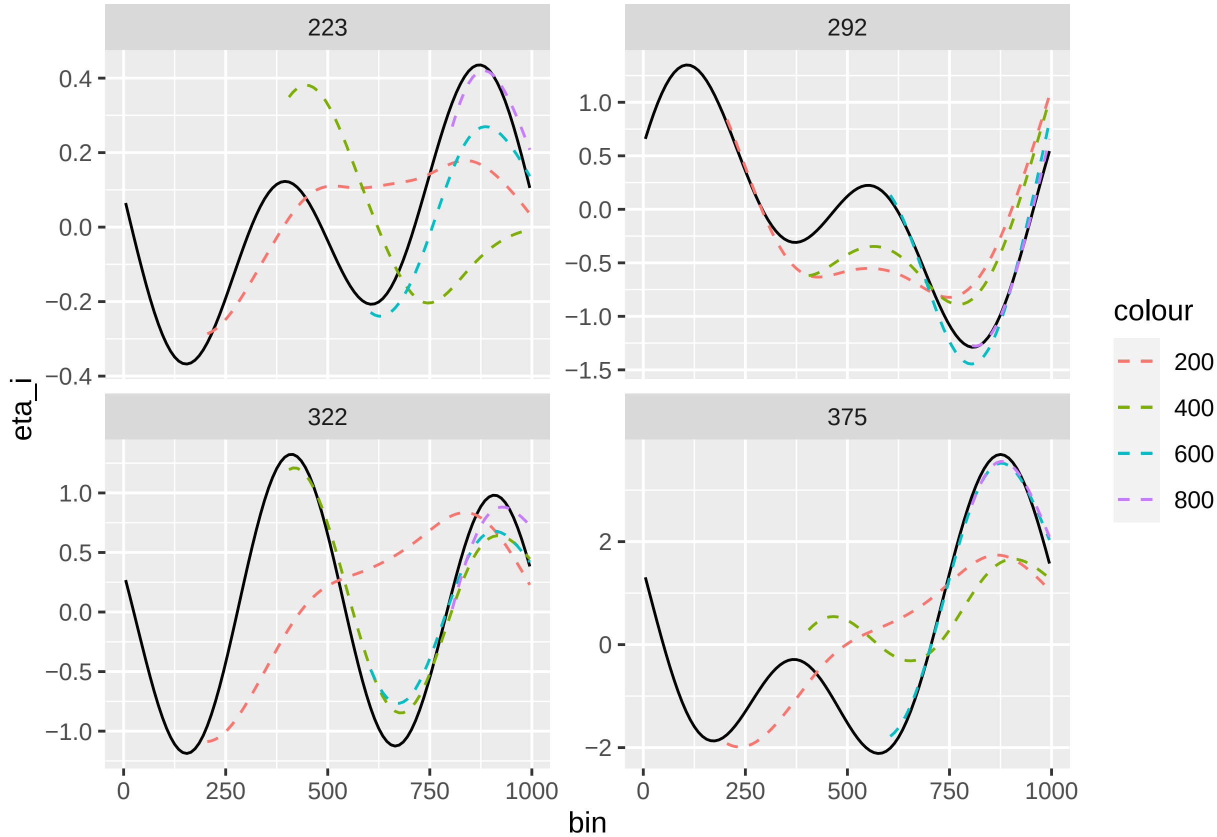
In this simulation, we set $f_0(t) = 0$. ξ_{ik} are mutually independent normal random variables $\xi_{ik} \sim N(0, \gamma_k)$. Here we set the values of γ_k to be 0.5^{k-1}, $k \in (1, ..., 4)$. In addition, for simplicity of presentation, we generate data on a regular grid, which means observations points are equally distributed across $[0, 1]$ and are the same for all subjects.

We use two metrics to evaluate the out-of-sample predictive performance: integrated squared error (ISE) and Area-Under-the-Receiver-Operator-Curve (AUC). ISE assess the prediction accucay of latent continuous function. It is evaluated on the binned grid at midpoints of each unobserved bin. If the entire functional domian has S bins, but we have observations up to the mth bin, then ISE is defined as $\frac{1}{N}\sum_{i=1}^{N}\sum_{s=m+1}^{S}(\hat{\eta}_i(s) - \eta_i(s))^2$.

The second metric, AUC, focuses on evaluation of prediction of the binary outcome. Since the binary outcomes are generated on the original, un-binned grid, we evaluated AUC on this grid as well and estimated values of latent functions between bins with linear interpolation. Just like ISE, the AUC report is also the average value across the whole sample.

As a reference method, we compare our method to Generalized Linear Mixed Models using Adaptive Gaussian Quadrature (GLMMadaptive). This is one of the fastest existing method developed for dynamic prediction of repeated generalized outcomes. Just like many mixed models, this method is very limited in terms of flexibility. For example, the model used for prediction of our simulated datset would simply be an linear model with one covariate indicating observation time: $g(E(Y_i)) = \beta_0 + \beta_1 t + b_{i0} + b_{i1} t$. While the flexibility of this mixed model can be increased using spline functions, the dimension of spline basis is also restricted by computational ability, and is unfeasible to implement under the scale of our simulated data or the complexity of our proposed method.

Result




```
## [1] 4.943948
```

```
##           t200     t400     t600     t800
##  [1,] 66.24273 22.54873 3.542797 1.240280
##  [2,] 67.78446 23.41331 3.877125 1.361210
##  [3,] 72.87894 23.66563 3.765386 1.425343
##  [4,] 69.76188 23.70570 3.795493 1.550903
##  [5,] 73.70884 23.58198 3.725293 1.391400
##  [6,] 65.76142 22.99703 3.503089 1.344802
##  [7,] 65.40496 22.13871 3.736834 1.498508
##  [8,] 64.26236 23.28330 3.619222 1.329766
##  [9,] 64.02149 23.53012 3.522818 1.259254
## [10,] 67.85612 22.87279 3.663146 1.255833
```

```
## [11,] 68.28494 22.23799 3.705315 1.275184
## [12,] 69.28752 23.86819 3.485495 1.222193
## [13,] 68.23590 23.65544 3.541065 1.216607
## [14,] 69.01748 22.64045 3.524419 1.292440
## [15,] 68.79349 22.06252 3.287550 1.184345
## [16,] 66.64831 22.52152 3.464968 1.279325
## [17,] 62.74778 21.33425 3.551274 1.412606
## [18,] 67.86389 22.45924 3.612767 1.249918
## [19,] 66.17925 23.08272 3.650364 1.403443
## [20,] 65.40744 22.71652 3.335048 1.251820
## [21,] 63.34640 22.76221 3.582341 1.282970
## [22,] 70.11289 23.31881 3.513332 1.233090
## [23,] 68.80965 21.25318 3.542332 1.330683
## [24,] 69.36756 22.73243 3.270891 1.275345
## [25,] 64.95044 19.84767 3.360214 1.196233
## [26,] 65.51981 22.85911 3.193166 1.198874
## [27,] 69.85064 22.08971 3.705718 1.271486
## [28,] 69.16139 23.61882 3.510015 1.267455
## [29,] 67.16257 21.46370 3.365553 1.228573
## [30,] 61.96033 22.93042 3.470349 1.333235
## [31,] 66.51524 22.07824 3.582422 1.195378
## [32,] 66.91172 21.75199 3.553637 1.361830
## [33,] 66.29053 21.94500 3.418139 1.204997
## [34,] 67.75195 24.49674 3.508264 1.255619
## [35,] 68.96053 23.65381 3.383380 1.232158
## [36,] 71.06036 23.01888 3.613351 1.314470
## [37,] 66.73265 22.97031 3.774147 1.414883
## [38,] 68.63432 23.83645 3.627263 1.206341
## [39,] 65.90022 21.55775 3.510041 1.345698
## [40,] 68.45275 22.15695 3.724797 1.372378
```

```
## [41,] 64.52251 20.46636 3.347395 1.232621
## [42,] 70.15392 23.06395 3.718334 1.466085
## [43,] 71.57331 22.62605 3.549136 1.260566
## [44,] 65.92148 22.24961 3.815144 1.399057
## [45,] 78.56107 24.14073 3.552019 1.262953
## [46,] 71.65332 23.34037 3.480720 1.406164
## [47,] 69.35799 24.58813 3.481430 1.328244
## [48,] 70.58539 23.91938 3.463617 1.240770
## [49,] 67.64045 24.01861 3.483040 1.353263
## [50,] 64.00190 21.22364 3.258786 1.241889
```

```
##            t200      t400      t600      t800
##  [1,] 0.6831917 0.7072828 0.6962669 0.6216755
##  [2,] 0.6881236 0.7011135 0.6886562 0.6203330
##  [3,] 0.6903756 0.7055576 0.6949729 0.6219840
##  [4,] 0.6869610 0.6983328 0.6946955 0.6178570
##  [5,] 0.6857926 0.7086377 0.6983496 0.6223322
##  [6,] 0.6932462 0.7029830 0.6939192 0.6262192
##  [7,] 0.6845848 0.7084870 0.6963884 0.6227765
##  [8,] 0.6864029 0.7031216 0.6957261 0.6210430
##  [9,] 0.7019075 0.7089013 0.6992221 0.6225649
## [10,] 0.6930817 0.7096679 0.6982366 0.6265394
## [11,] 0.6898522 0.7119879 0.6986544 0.6209836
## [12,] 0.6878123 0.7018091 0.6955114 0.6210767
## [13,] 0.6833768 0.6994038 0.6881579 0.6194239
## [14,] 0.6855302 0.7066976 0.6960071 0.6285915
## [15,] 0.6922020 0.7088249 0.6899797 0.6212294
## [16,] 0.6818066 0.7035349 0.6882206 0.6113030
## [17,] 0.6851981 0.6932127 0.6777413 0.6117117
## [18,] 0.6937411 0.7060218 0.6890442 0.6189389
```

```
## [19,] 0.6813624 0.6984211 0.6908621 0.6203759
## [20,] 0.6831171 0.6996776 0.6940369 0.6232097
## [21,] 0.6899041 0.7031550 0.6952849 0.6248322
## [22,] 0.6838072 0.7102512 0.6923645 0.6152336
## [23,] 0.6876191 0.7095309 0.6952974 0.6225838
## [24,] 0.6757380 0.7029618 0.6881554 0.6193086
## [25,] 0.6817890 0.7081914 0.6955291 0.6212917
## [26,] 0.6964732 0.7066309 0.6966653 0.6234079
## [27,] 0.6728153 0.7007066 0.6943636 0.6281670
## [28,] 0.6820400 0.7006211 0.6877082 0.6183698
## [29,] 0.6855178 0.7010634 0.6839710 0.6164009
## [30,] 0.6834277 0.6966205 0.6941087 0.6172431
## [31,] 0.6902815 0.7106810 0.6997837 0.6236359
## [32,] 0.6815398 0.7073760 0.6929314 0.6204267
## [33,] 0.6878528 0.7039848 0.6912621 0.6196722
## [34,] 0.6785079 0.6996903 0.6900174 0.6188339
## [35,] 0.6902678 0.7116401 0.6970651 0.6271642
## [36,] 0.6773438 0.7014910 0.6953198 0.6294785
## [37,] 0.6998593 0.7111832 0.6973497 0.6246476
## [38,] 0.6943192 0.7106729 0.6956372 0.6206847
## [39,] 0.6676717 0.6921172 0.6837725 0.6194201
## [40,] 0.6846974 0.7089066 0.6912748 0.6273155
## [41,] 0.6900183 0.6996615 0.6901512 0.6217254
## [42,] 0.6830576 0.7073767 0.6927547 0.6301514
## [43,] 0.6838424 0.7087791 0.6966853 0.6314348
## [44,] 0.6980983 0.7061690 0.6944230 0.6294428
## [45,] 0.6892633 0.7142045 0.7062987 0.6322929
## [46,] 0.6905329 0.7085717 0.6925943 0.6220946
## [47,] 0.6922206 0.7064544 0.6933813 0.6238599
## [48,] 0.6770491 0.7012942 0.6921313 0.6226142
```

```
## [49,] 0.6864164 0.7075723 0.6958503 0.6218655
## [50,] 0.6884789 0.7061650 0.6954227 0.6214933
```

```
## [1] 1
```

Data application

Discussion

- Grid
- Score bias: cannot demonstrate without repeat simulation

References

Chen, H., Wang, Y., Paik, M. cho, and Choi, H. A. (2013), "A marginal approach to reduced-rank penalized spline smoothing with application to multilevel functional data," *J Am Stat Assoc.*, 108, 1216–1229. https://doi.org/10.1080/01621459.2013.826134.

Chiou, J.-M. (2012), "Dynamical functional prediction and classification, with application to traffic flow prediction," *The Annals of Applied Statistics*, Institute of Mathematical Statistics, 6, 1588–1614. https://doi.org/10.1214/12-AOAS595.

Delaigle, A., and Hall, P. (2016), "Approximating fragmented functional data by segments of markov chains," *Biometrika*, 103, 779–799. https://doi.org/10.1093/biomet/asw040.

Gertheiss, J., Goldsmith, J., and Staicu, A. (2016), "A note on modeling sparse exponential-family functional response curves," *Comput Stat Data Anal*, 105, 46–52. https://doi.org/10.1016/j.csda.2016.07.010.

Goldberg, Y., Ritov, Y., and Mandelbaum, A. (2014), "Predicting the continuation of a function with applications to call center data," *Journal of Statistical Planning and Inference*, 147, 53–65. https://doi.org/https://doi.org/10.1016/j.jspi.2013.11.006.

Goldsmith, J., Zipunnikov, V., and Schrack, J. (2015), "Generalized multilevel function-on-scalar regression and principal component analysis," *Biometrics*, 71, 344–53. https://doi.org/10.

1111/biom.12278.

Hall, P., Müller, H.-G., and Yao, F. (2008), "Modelling sparse generalized longitudinal observations with latent gaussian processes," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70, 703–723. https://doi.org/https://doi.org/10.1111/j.1467-9868.2008.00656.x.

Kraus, D. (2015), "Components and completion of partially observed functional data," *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, [Royal Statistical Society, Wiley], 77, 777–801.

Laird, N. M., and Ware, J. H. (1982), "Random-effects models for longitudinal data," *Biometrics*, [Wiley, International Biometric Society], 38, 963–974.

Leroux, A., Crainiceanu, C. M., and Wrobel, J. (n.d.). "Fast generalized functional principal component analysis."

Leroux, A., Xiao, L., Crainiceanu, C., and Checkley, W. (2018), "Dynamic prediction in functional concurrent regression with an application to child growth," *Statistics in medicine*, 37, 1376–1388.

LIANG, K.-Y., and ZEGER, S. L. (1986), "Longitudinal data analysis using generalized linear models," *Biometrika*, 73, 13–22. https://doi.org/10.1093/biomet/73.1.13.

Linde, van der (2009), "A bayesian latent variable approach to functional principal components analysi with binary and count data," *A StA Adv Stat Anal*, 307–333. https://doi.org/10.1007/s10182-009-0113-6.

Lindstrom, M. J., and Bates, D. M. (1990), "Nonlinear mixed effects models for repeated measures data," *Biometrics*, [Wiley, International Biometric Society], 46, 673–687.

"Nonlinear models for repeated measurement data: An overview and update" (2003), [International Biometric Society, Springer], 8, 387–419.

Rizopoulos, D. (2022), *GLMMadaptive: Generalized linear mixed models using adaptive gaussian quadrature*.

Scheipl, F., Staicu, A.-M., and Greven, S. (2014), "Functional additive mixed models," *J Comput Graph Stat*, 24, 447–501. https://doi.org/10.1080/10618600.2014.901914.

Shang, H. L. (2017), "Functional time series forecasting with dynamic updating: An application to intraday particulate matter concentration," *Econometrics and Statistics*, 1, 184–200. https://doi.org/https://doi.org/10.1016/j.ecosta.2016.08.004.

Suresh, K., Taylor, J. M. G., Spratt, D. E., Daignault, S., and Tsodikov, A. (2017), "Comparison of joint modeling and landmarking for dynamic prediction under an illness-death model," *Biom J*, 59, 1277–1300. https://doi.org/10.1002/bimj.201600235.

Wrobel, J., Zipunnikov, V., Schrack, J., and Goldsmith, J. (2019), "Registration for exponential family functional data," *Biometrics*, 75, 48–57. https://doi.org/https://doi.org/10.1111/biom.12963.