Fingerprinting Raw Accelerometry Data - A Functional Regression Approach

Andrew Leroux

October 24, 2019

The Set Up

- The “Big" goal:
  - Predict whether a new time series comes from a particular individual
- A more (potentially?) tractable goal:
  - Predict whether a labelled time series (e.g. walking) comes from a particular individual
- A regression based approach:
  1. Start simple: predict subject-specific features (age, sex, bmi, etc.)
  2. Go all-in: predict subjects

Problem Set Up

- Assume
  1. We have well labeled accelerometry data for model building
  2. We predict using similarly well labeled data observed at a different time
  3. Device worn on the same part of the body for all subjects (left wrist)
- Do not assume
  1. Devices are oriented the same
  2. We have landmarked features

Approach: Model

- Choose l time length
- Sample $j = 1, \ldots, J$ non-overlapping intervals of length l
- Denote each interval as $X_{ij}(t), t \in [0, l]$
- Let $[Y_{ij}]_{j=1,\ldots,J} = \mathbf{1}_{J\times 1} Y_i$
- Consider two different models form $g(E[Y_{ij}]) = \eta_{ij}$ where

$$\text{Model 1:} \eta_{ij} = \beta_0 + \int_0^l f(X_{ij}(u))du$$

$$\text{Model 2:} \eta_{ij} = \beta_0 + \int_{u=0}^l \int_{s=0}^u F(X_{ij}(u), X_{ij}(s), u-s)dsdu$$

Approach: Some Model Intuition

$$\text{Model 1:}\eta_{ij} = \beta_0 + \int_0^I f(X_{ij}(u))du$$

$$\text{Model 2:}\eta_{ij} = \beta_0 + \int_{u=0}^I \int_{s=0}^u F(X_{ij}(u), X_{ij}(s), u - s)dsdu$$

- Model 1
  - Gait sped – number of "peaks", duration of troughs, etc.
  - Overall magnitude of peaks
- Model 2
  - Cyclic patterns, rates of decrease/increase in acceleration
  - Consistency of peaks in timing

Approach: Prediction

- For each interval we obtain $g^{-1}(\hat{\eta}_{ij})$
- We average over intervals to obtain a single prediction for each participant

$$\hat{Y}_i = \frac{1}{J}\sum_{j=1}^{J} g^{-1}(\hat{\eta}_{ij})$$

- Alternatively, could also average on the linear predictor scale

Model Parameters

- Split subjects' walking data into non-overlapping $I = 1$ second intervals
- Each subject has at least 380 such intervals
- Split subjects' intervals into training and test data
  - Training data: First $J = 200$ intervals (400 seconds total)
  - Test data: Last $J = 180$ intervals (360 seconds total)
- Fit models on 3 different outcomes (age, height, sex)

Predicting subject-specific Characteristics

Outcome	Linear Predictor	
	Model 1	Model 2
Age	$\hat{R}^2 = 33.6\%$	$\hat{R}^2 = 47.0\%$
Height	$\hat{R}^2 = 5.4\%$	$\hat{R}^2 = 30.7\%$
Gender	$\hat{\text{AUC}} = 0.69$	$\hat{\text{AUC}} = 0.92$

- Permuting outcomes completely erases predictive power of the model
- Model performance is evaluated out-of-sample
- Do we really believe these patterns are specific to age, height, or gender?

Predicting Subjects: Method

- Initial idea: Mutlinomial model
- More feasible: Separate logistic regression (one-vs-rest classification). Denote model fit using subscript $k = 1, \ldots, N$
  - Estimate $\hat{\eta_{ijk}} = \log(\Pr(Y_i = k)/\Pr(Y_i \neq k)$
  - Obtain

$$\widehat{\Pr}(Y_{ij} = k) = \frac{\exp(\hat{\eta}_{ijk})}{\sum_{m=1}^{N} \exp(\hat{\eta}_{ijm})}$$

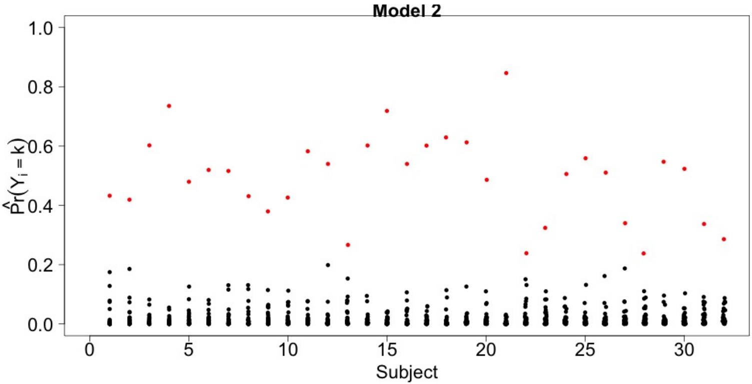
  - Finally

$$\widehat{\Pr}(Y_i = k) = \frac{1}{J} \sum_{j=1}^{J} \widehat{\Pr}(Y_{ij} = k)$$

- Classify subjects as $\hat{Y}_i = \arg\max_k \widehat{\Pr}(Y_i = k)$

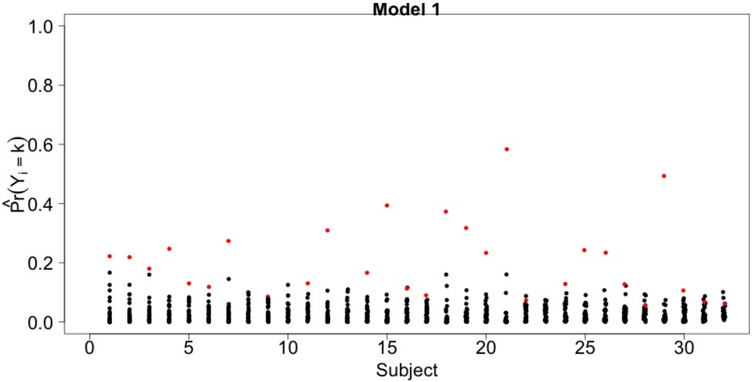
Predicting Subjects: Results

- 100% classification accuracy with Model 2

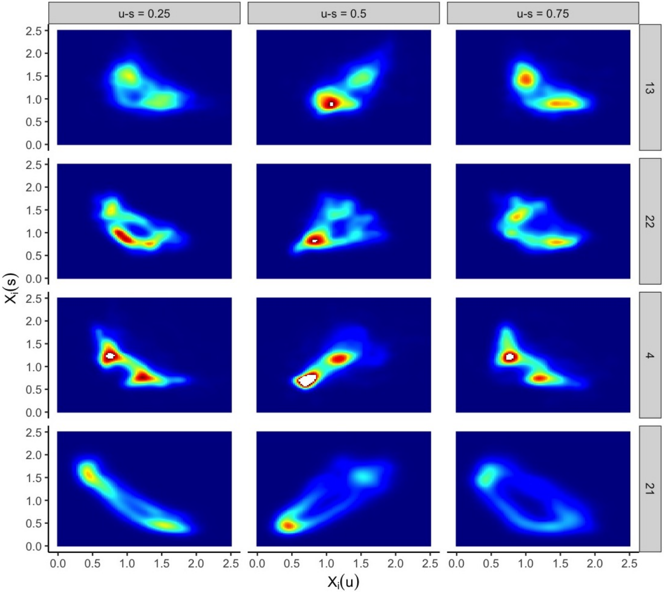


Predicting Subjects: Results

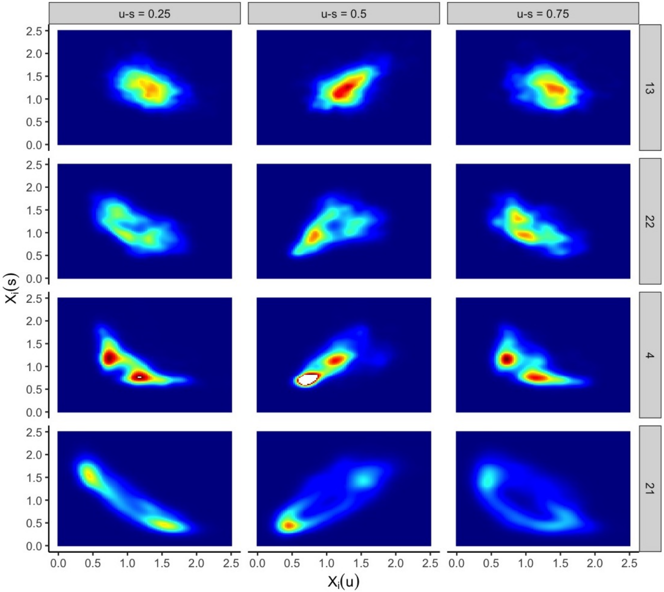
- 75% classification accuracy with Model 1



Predicting Subjects: Training Data



Predicting Subjects: Test Data



Some Thoughts

- Limitations
  - Requires labelled training and test data
  - Need to choose interval length (computational and sample size concerns)
  - Requires a definined population for comparison
  - Computationally not expected to scale well
  - Lab walking vs "in-the-wild" walking
- Proof of concept! Extracting patterns seems to hold some signal
- Alternatives to functional regression
  - Estimate 3-d densities (or 2-d conditional on s-u), establish some sort of thresholding
  - Machine learning on the pairwise difference vectors