May 14th, 10:00 AM - 2:00 PM

Time Series Modeling of Baseball Performance

Kyle Andelin
Western Washington University

Sam Kaplan
Western Washington University

Time Series Modeling of Baseball Performance

Kyle Andelin, Sam Kaplan and Brian Hutchinson

Computer Science Department, Western Washington University

Overview

Motivation: Predicting upcoming player performance is vital to team management and a hot topic in sports media

Goal: Greater understanding of recent trends impacting future outcomes and increased accuracy of predictions

Approaches:

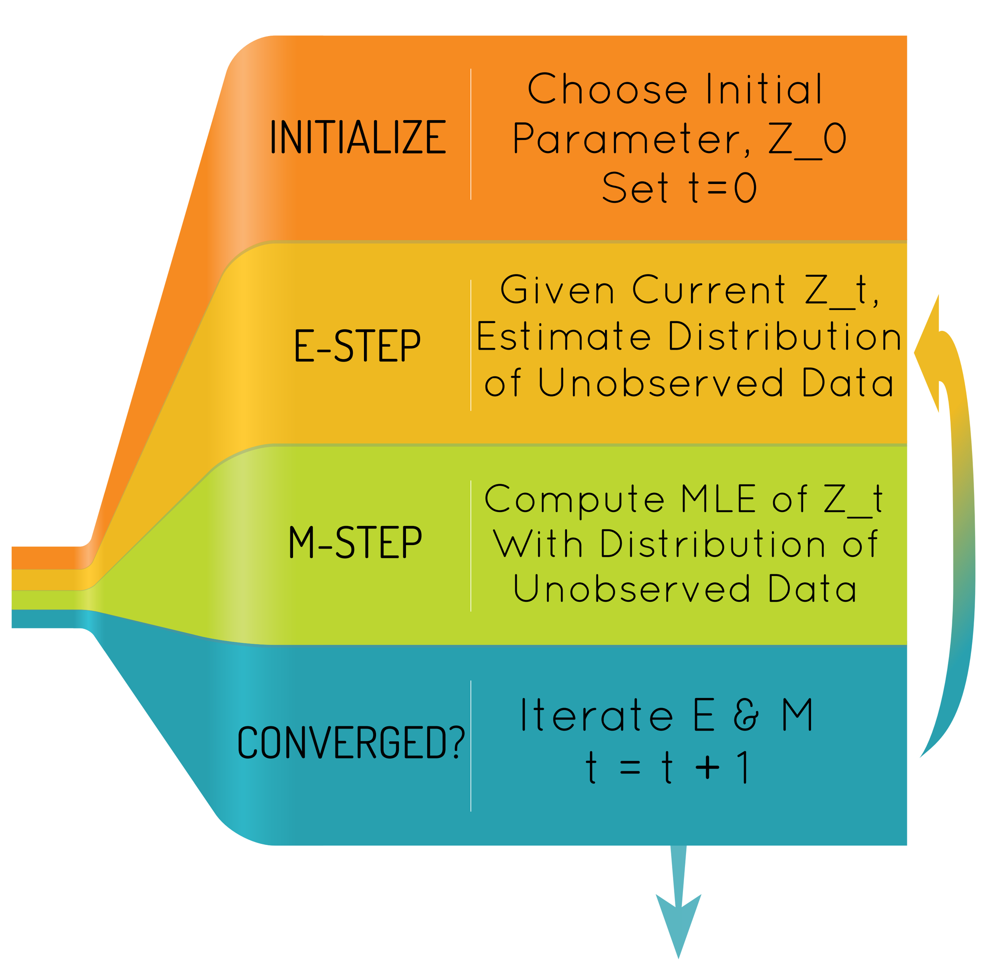
I. Use expectation maximization (EM) to identify most predictive past time periods

II. Predict next game performance based on season history using a recurrent neural network (RNN)

Background

Expectation Maximization

- The EM algorithm is a general iterative method to perform maximum likelihood estimation (MLE)
- Find MLE of mixture density parameters via EM



EM Process

- Unobserved data is a variable indicating mixture component membership
- Terminate when changes in estimates are insignificant

Our Model

Mixture Model

- Future performance as a function of past performance periods:

$$\boldsymbol{P_i(x)} = \boldsymbol{w_1} \underbrace{\boldsymbol{P_{i,1}(x)}}_{\text{Past}} + \boldsymbol{w_2} \underbrace{\boldsymbol{P_{i,2}(x)}}_{\text{Early}} + \boldsymbol{w_3} \underbrace{\boldsymbol{P_{i,3}(x)}}_{\text{Recent}}$$

where

$$\boldsymbol{P_{i,j}(x) = (1-\alpha)\tilde{P}_{i,j}(x) + \alpha\delta_j(x)}$$

$\boldsymbol{\tilde{P}_{i,j}}$ - player $\boldsymbol{i}$ empirical PMF for period $\boldsymbol{j}$

$\boldsymbol{\delta_j}$ - league average PMF for period $\boldsymbol{j}$

$\boldsymbol{\alpha}$ - interpolation coefficient of league average PMF, $\boldsymbol{0 \le \alpha \le 1}$

$\boldsymbol{w_j}$ - mixing weight for time period $\boldsymbol{j}$

$$\boldsymbol{0 \le w_j \le 1}, \quad \boldsymbol{w_1 + w_2 + w_3 = 1}$$

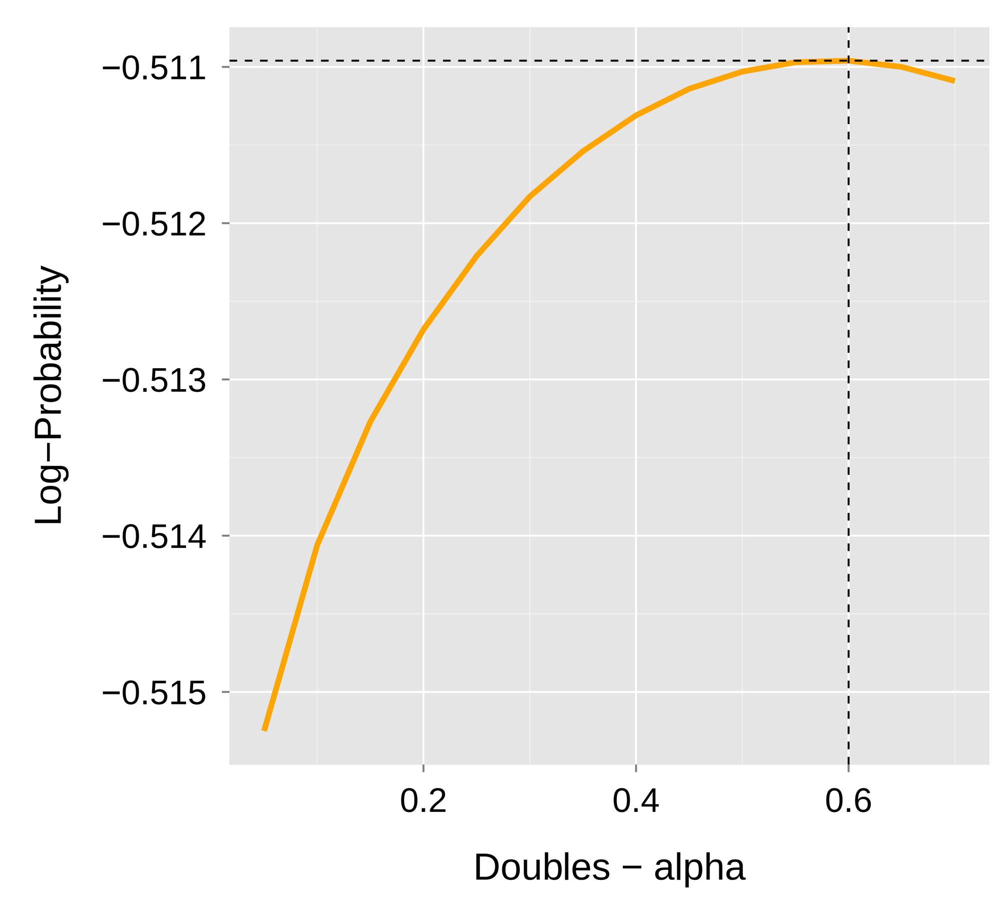
Experiments

Data

- Play-by-play data from Retrosheet.org
- 250+ players per season, years 2000-2013
- 6 statistics: strikeouts (K), walks (BB), singles (1B), doubles (2B), triples (3B), homeruns (HR)

Training

- Tune $\boldsymbol{\alpha}$ to maximize log-likelihood on a held out data set
- Use EM to learn appropriate $\boldsymbol{w_j}$ weights to best predict future outcomes



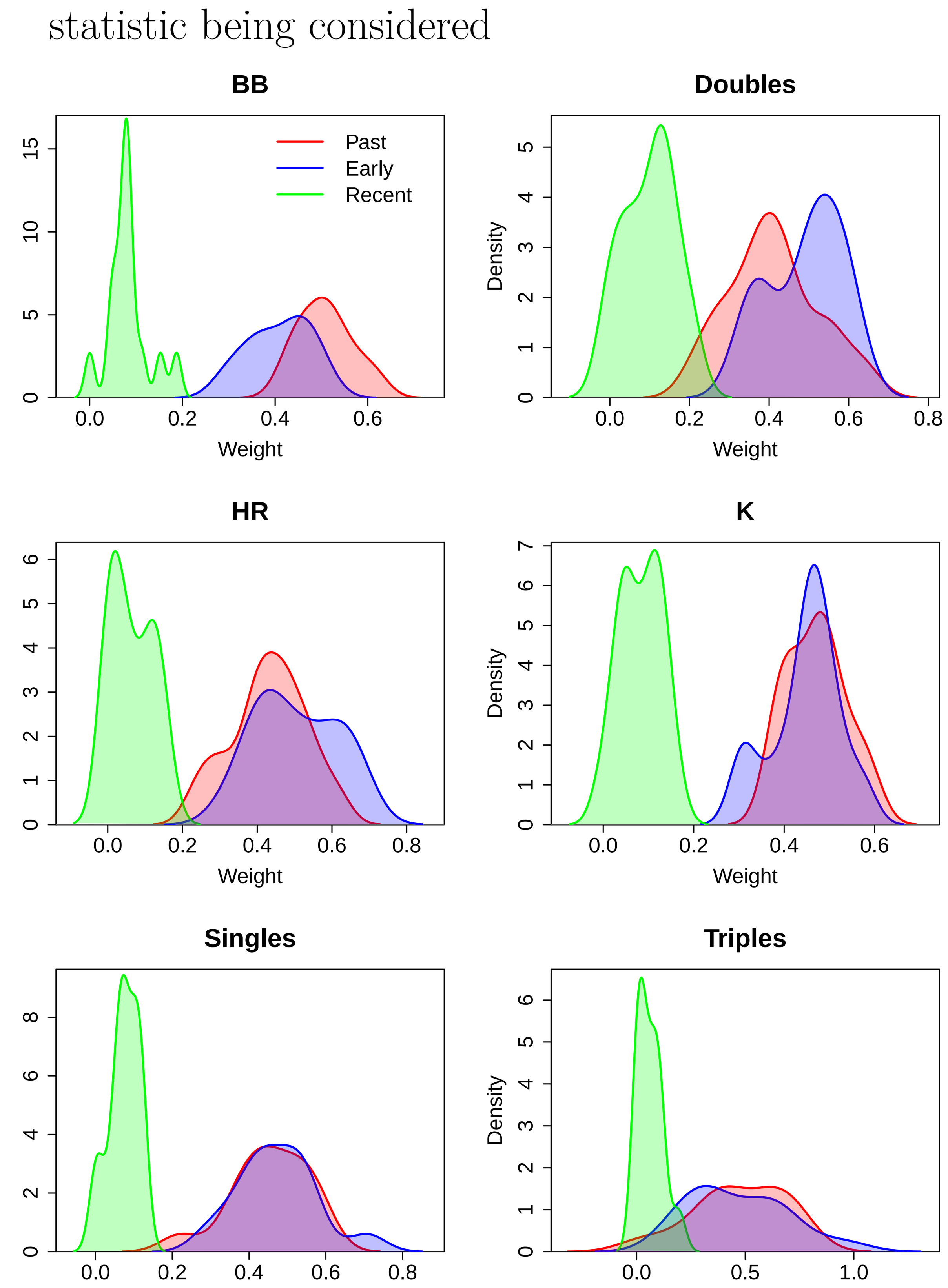
Log-Probability as a function of $\boldsymbol{\alpha}$

Results

Log-Probability of Optimal $\boldsymbol{\alpha}$ Model vs League Average Model

	K	BB	1B	2B	3B	HR
Optimal $\boldsymbol{\alpha}$	0.2	0.25	0.35	0.6	0.45	0.1
Optimal LogP	-1.016	-0.747	-1.002	-0.510	-0.092	-0.332
$\boldsymbol{\alpha}$ = 100%	-1.042	-0.765	-1.011	-0.512	-0.094	-0.347

- The optimal $\boldsymbol{\alpha}$ is highly dependent on the statistic being considered



Learned Weights Density

Recent: previous two weeks

Past: last season

Early: start of season up until Recent

RNN

Data

- Stats per player, per game, over a season:

<s> 1 1 1 1 2 0 ... 0 0 1 1 0 0 </s>

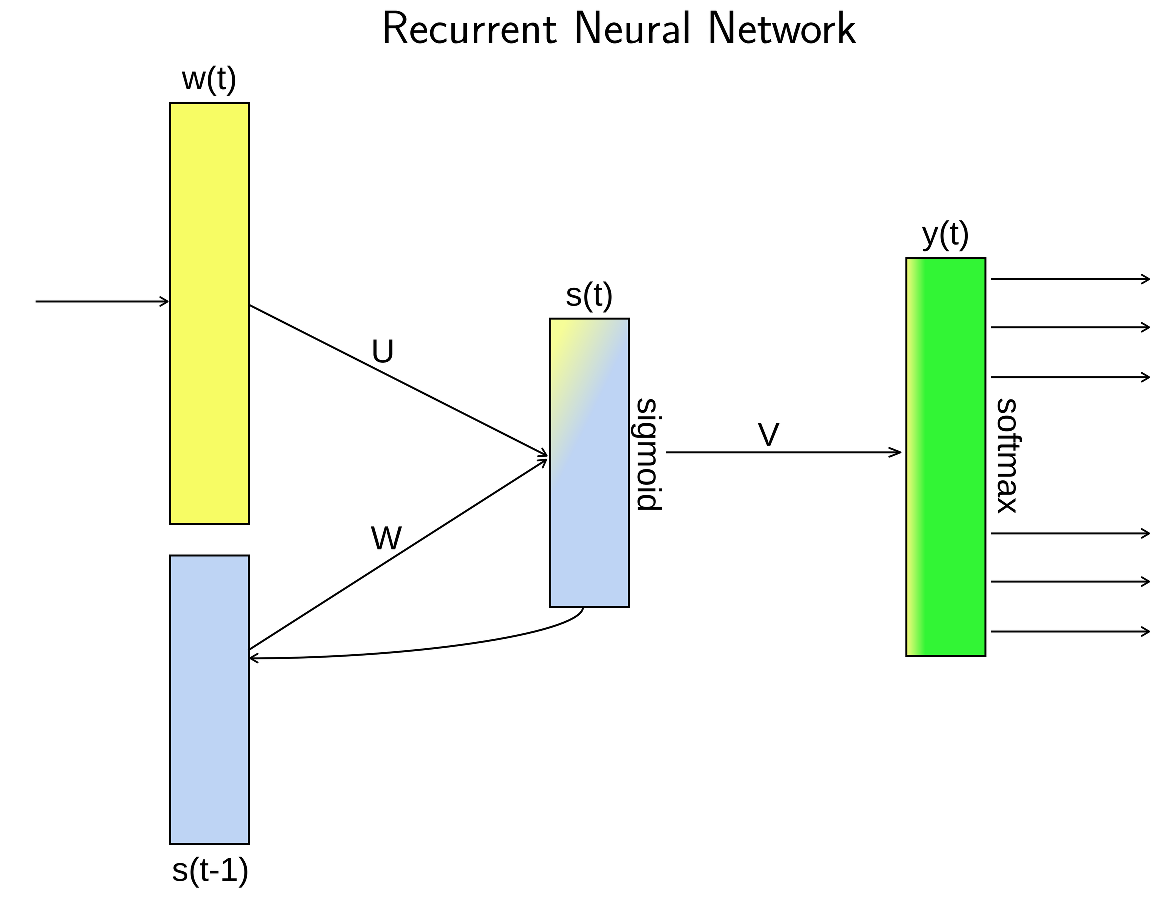
<s> 0 0 0 0 0 1 ... x 0 x 1 x 0 </s>

<s> 1 0 1 0 1 0 ... 1 0 0 0 0 1 </s>

  - x = did not play | </s> = end of season

Model

Recurrent Neural Network



$$\mathbf{s}(t) = f(\mathbf{U}\mathbf{w}(t) + \mathbf{W}\mathbf{s}(t-1))$$
$$\mathbf{y}(t) = g(\mathbf{V}\mathbf{s}(t))$$
$$f(z) = \frac{1}{1+e^{-z}}, \quad g(z_m) = \frac{e^{z_m}}{\sum_k e^{z_k}}$$

- Hidden layer, $\mathbf{s}(t-1)$, represents history
- Tune on:
  - Size of hidden layer
  - Number of time steps to backpropagate error

Training

- Train on 60% of data - Tune on 20% - Test on remaining 20%
  - Learn optimal $\mathbf{U}, \mathbf{W}, \mathbf{V}$ matrices
  - Trained using backpropagation through time

Results

- To evaluate our methods, we feed held out data to our model in the above form and compute the metrics in the following table

Early Measures of Model Performance

	K	BB	1B	2B	3B	HR	Hits
RNN MSE	0.64	0.29	0.53	0.16	0.02	0.09	0.75
Lg Avg MSE	0.66	0.29	0.54	0.16	0.02	0.09	0.77
Run Avg MSE	0.67	0.36	0.64	0.20	0.03	0.12	0.84
RNN MAE	0.67	0.43	0.62	0.28	0.03	0.16	0.69
Lg Avg MAE	0.68	0.44	0.64	0.28	0.03	0.17	0.71
Run Avg MAE	0.66	0.44	0.64	0.29	0.05	0.18	0.72
RNN % Cor	0.48	0.75	0.57	0.85	0.98	0.92	0.45

- Ongoing work: we anticipate more results soon

Evolution of model over several games