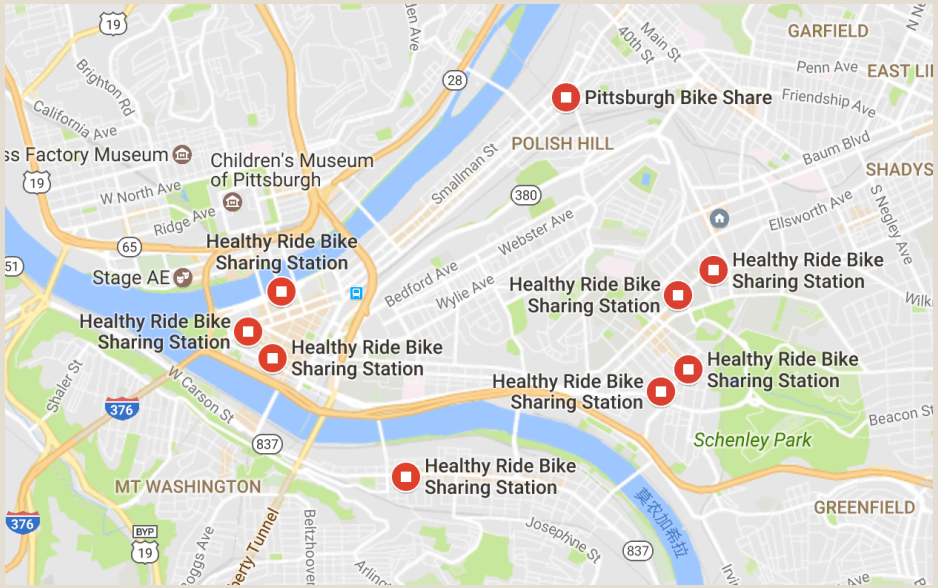
ENOUGH BIKES ALREADY FOR SHARING?

Predict the demand of the public bike sharing system in Pittsburgh

Fengtao Wu Yuwei Chen Zhendong Wang

Agenda

- A short introduction
  - Motivation
  - Data we used
- Method
  - Clustering
  - Classification
  - Regression
- Conclusion
  - Evaluation of our result
  - Future improvement
  - What we learned from this project



Motivation

- Our goal: predict the demand of the public bike sharing system in Pittsburgh
- To be more specific, we want to predict the bike renting count for a given station, for the last 10 days of a given month, for a given hour
- Benefits: bike provider can provide much better service and save money

RENTBIKE
EMPORDA

Data we used

- Weather Data:
  - Weather Condition (storming, raining, fog)
  - Temperature
  - Visibility
  - Wind speed
- Bike Renting Data
  - Bike renting count for a given time
  - Customer information
- Station Information
  - Station latitude and longitude
- Calendar Information

Data challenge

- Most station has zero rent count in most time

  (HARD to apply regression model directly)

- Different months have different ride patterns

  (NEED to build model month by month)

- Stations which have high rent count behave differently from those have low rent

  (BETTER to apply clustering first)

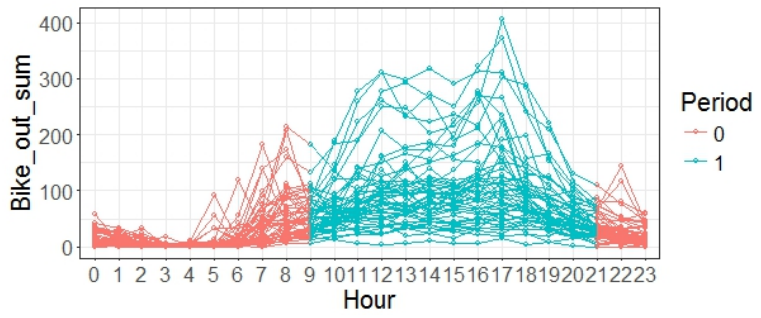
The Basic idea

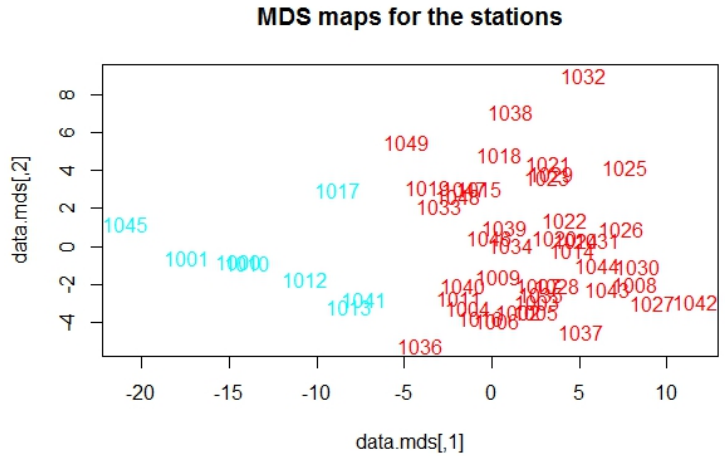
clustering + classification + regression

- Clustering :
  (station) + history data => high/low rent count groups

- Classification :
  (station, month, day, hour) + features + clustering result =>
  whether rent count is 1+/0

- Regression :
  (station) + features + clustering result + classification result =>
  rent bike count





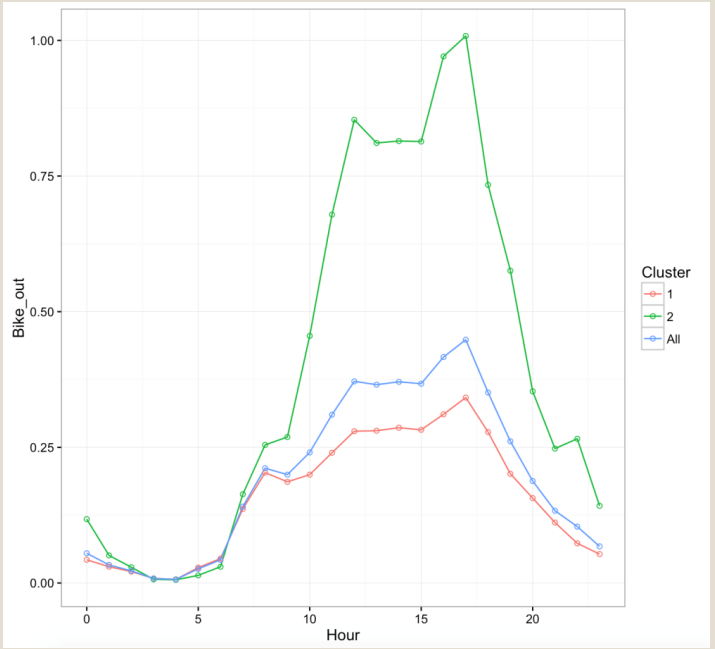
1. Station Clustering (why?)

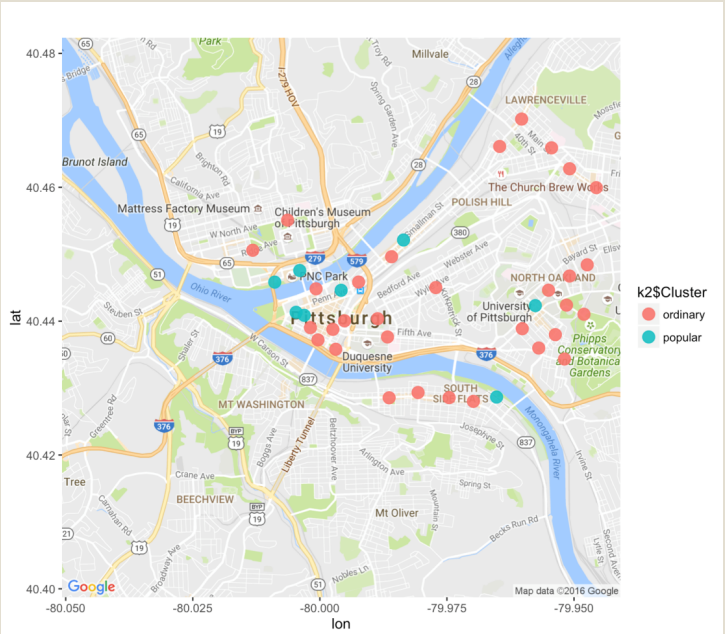
- Different station has different Temporal ride patterns in the course of day

1. Station Clustering (high/low groups)

- Method: k-means
- k=2 is optimal

	Davies-Bouldin	Dunn	Silhouette
k=2	**1.12**	**0.43**	**0.39**
k=3	1.76	0.33	0.16
k=4	1.74	0.27	0.14
k=5	1.88	0.32	0.12





Auto Regressive Model

Time & Calendar Data (TC)
Ride Data (R)
Weather Data (W)

$$B_t \sim \sum_i a \times TC_t + b \times R_{t-i} + c \times W_{t-i}$$

i = {1, 2}

T	B	TC	R	W
1	B_1	TC_1	R_1	W_1
2	B_2	TC_2	R_2	W_2
3	B_3	TC_3	R_3	W_3
4	B_4	TC_4	R_4	W_4

50 Stations
275 Days
24 Hours
330,000 Records

Ride Count	Percentage (%)
0	88.25
1	7.48
2	2.51
3	0.87
>3	0.89

2. Classification (why?)

- Different station has different Temporal ride patterns in the course of day
- Use Dataset of September 2015 as an example

2. Classification (1+/0 rent count)

- Method: Logistic Regression (BEST), SVM, Decision Tree, Naive Bayesian

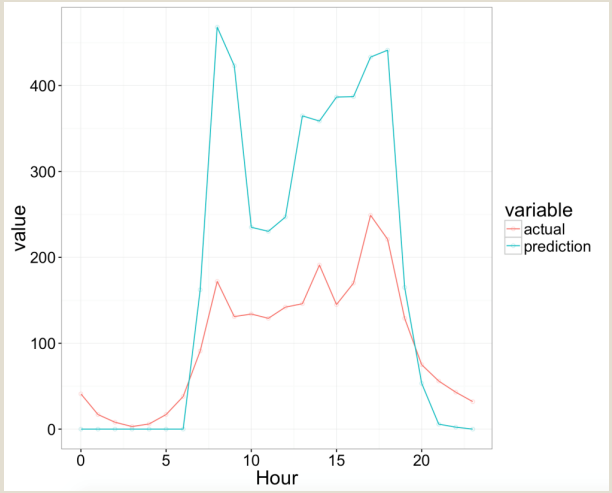
		cutoff=0.2	cutoff=0.5
cluster 1 (low rent count group)	accuracy	0.6840	0.8439
	sensitivity	0.6847	0.9981
	specificity	**0.6802**	**0.0121**
cluster 2 (high rent count group)	accuracy	0.6714	0.7938
	sensitivity	0.5939	0.9472
	specificity	**0.8918**	**0.3567**

3. Regression (Predict rent count)

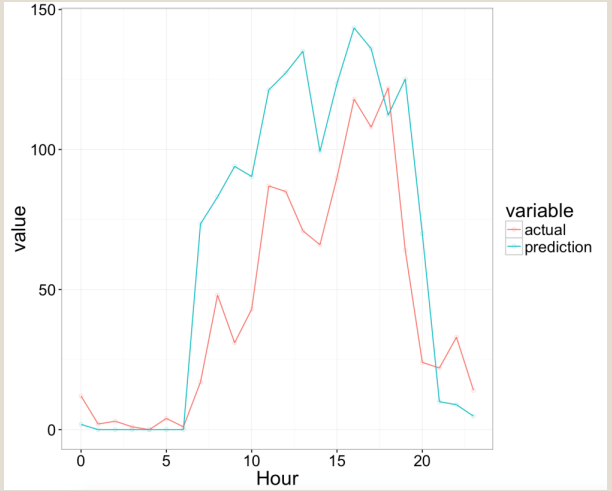
- Method: SVR (BEST), Linear Regression, Decision Tree

RMSE	Cluster 1	Cluster 2	Total
Origin	0.7939	1.2040	0.8725
Ceiling	1.1453	1.4263	1.1948
Flooring	**0.7443**	**1.1623**	**0.8255**
rounding	0.7625	1.2112	0.8504

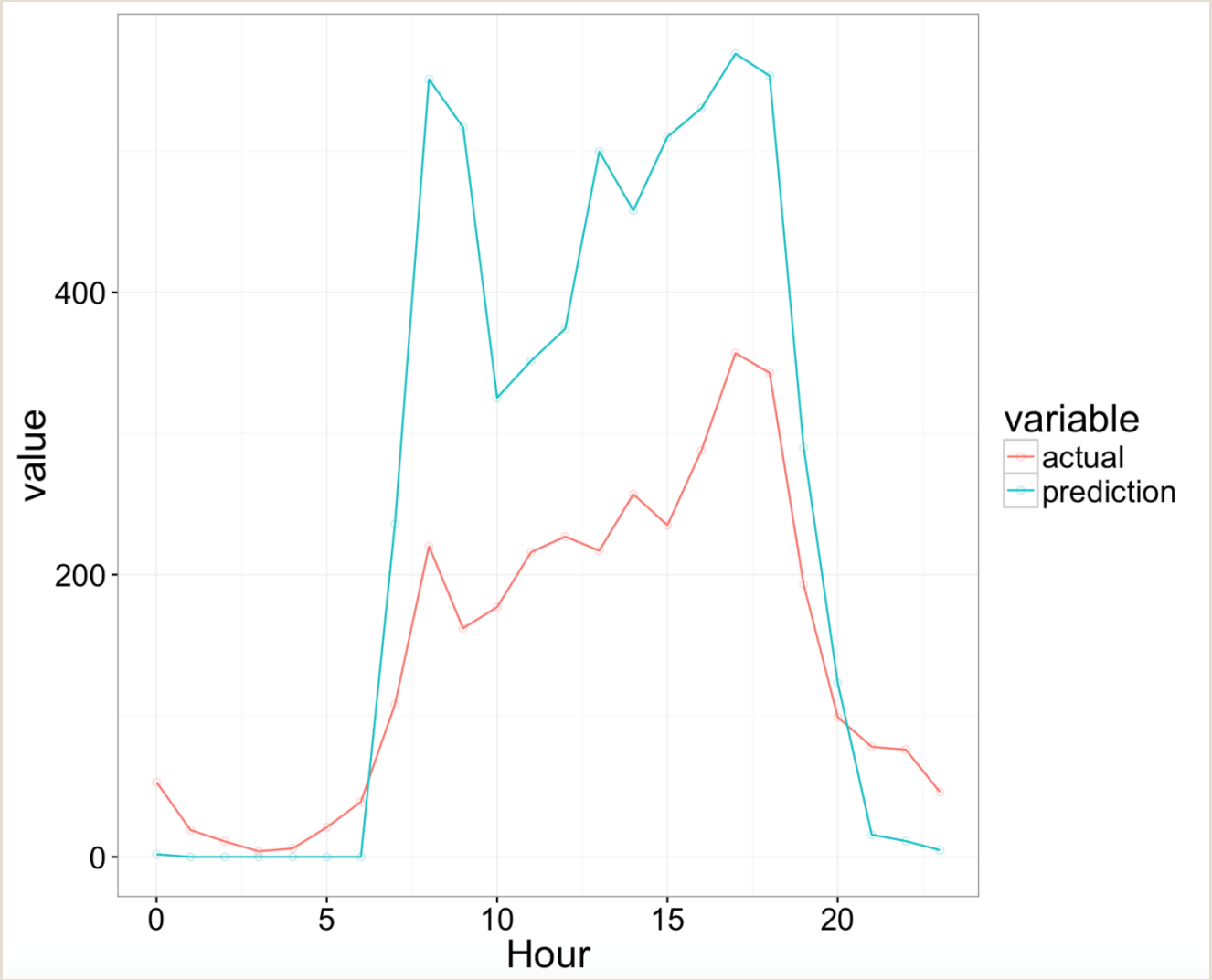
3. Regression (Predict rent count)



Low Group



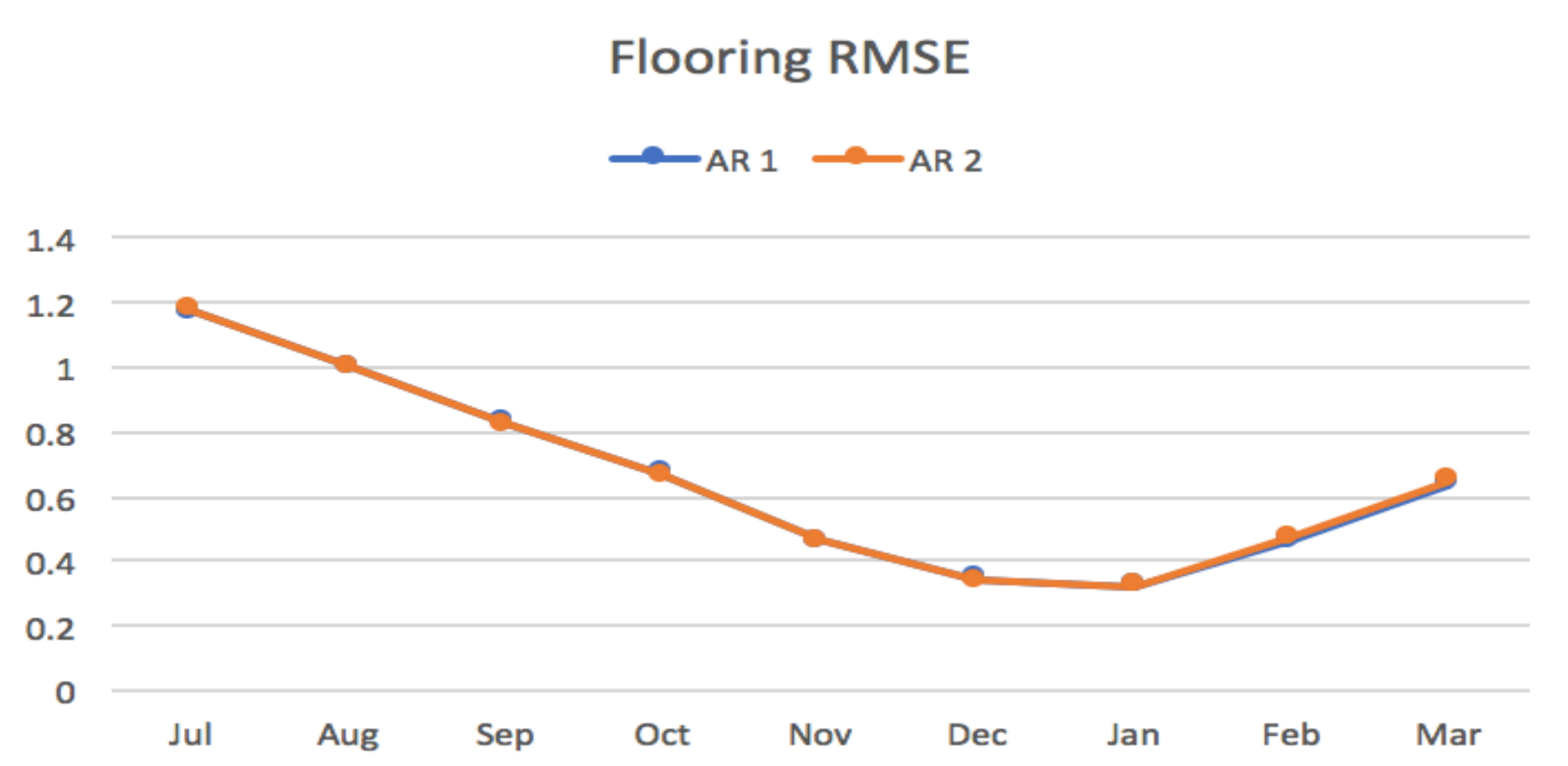
High Group



Combined

3. Regression (AR1 vs AR2)

	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar
AR 1	1.1724	1.0042	0.8255	0.6704	0.4635	0.3420	0.3219	0.4644	0.6369
AR 2	1.1766	1.0038	0.8247	0.6670	0.4631	0.3399	0.3187	0.4715	0.6499



Future Improvement

- Auto Regressive Model
- Classification Method

Auto Regressive Model Enhancement

- Current Issue: AR1 model and AR2 model have very similar result
- Potential Enhancement:
  - Select feasible lag
  - Use other time series analysis model
    - Moving-Average (MA) model
    - Gauge model

Classification Enhancement

- Current Issue: Parameter Tuning
- Potential Enhancement:
  - Logistic Regression: tune cutoff
  - SVM: reduce time-consuming
  - Decision Tree: prune branches

What we learned from this project

- Hybrid different DM method
  - Before: Apply regression directly and have terrible results
  - Now: Apply clustering and classification and then do regression

What we learned from this project

- Feature Selection
  - Before: We blindly try to use every feature we have
  - Now: We try to validate every feature to see if it has a significant on the final result. So we have a simpler and better model

What we learned from this project

- Clean data
  - Handle missing value
  - Encoding vague feature (weather condition)