Predicting Bikeshare Demand

Robert Kraig

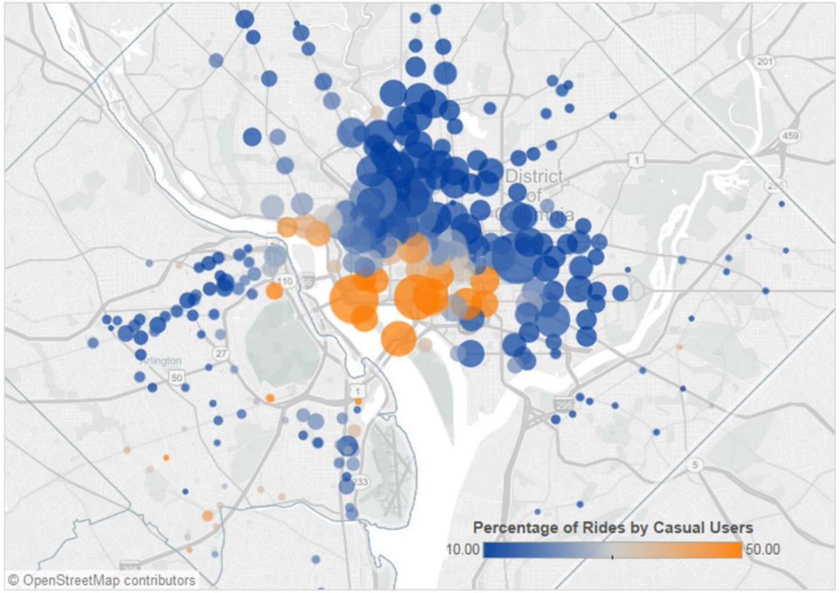
August 02, 2016

Capital Bikeshare (CaBi)

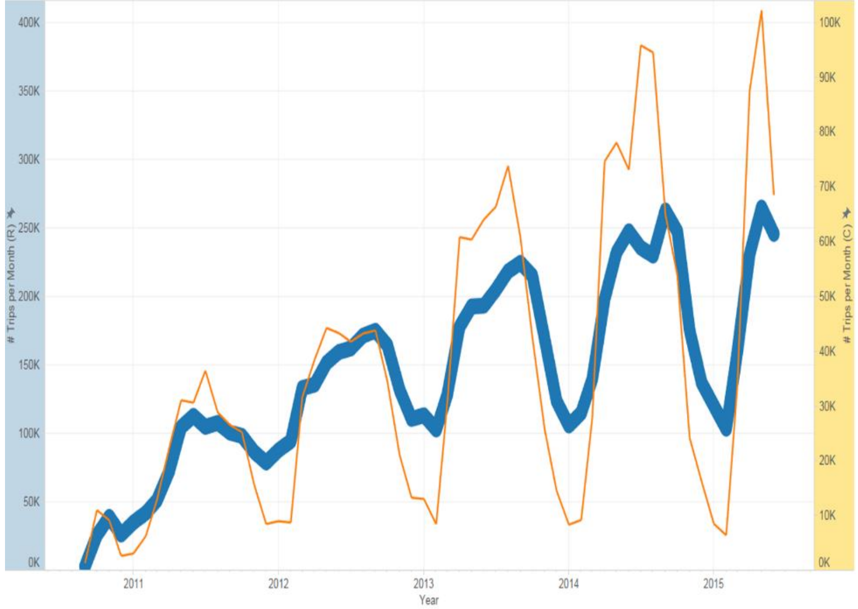
Self-guided, self-powered public transit

- DC / Arlington / Alexandria / MoCo
- First station: Crystal City, Sep 2010
- 360 Stations (10-45 docks each) used in 2014 and 2015
    - Average = 3-4 blocks between stations
- Member Types: Regular and Casual

Relative Ridership Quantity, 2014-2015

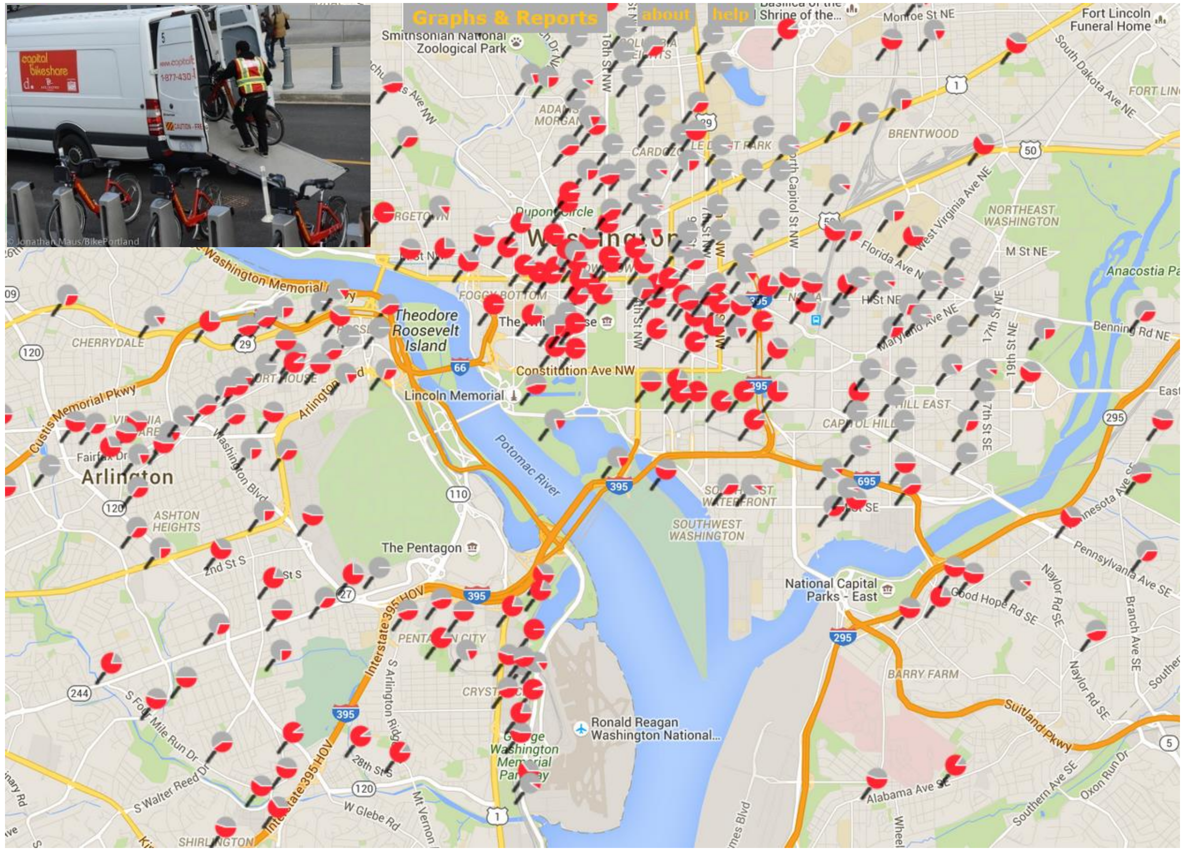

District of Columbia
Arlington
Percentage of Rides by Casual Users
10.00
50.00
© OpenStreetMap contributors

Rides per month, 2010-2015



Challenge: System Capacity

- Problem: “Dock-blocking”
    - No bikes or no docks available
- Mitigation Strategies
    - Rebalancing
    - Corrals
    - Infill stations
    - Alternative pricing schemes?



Visualization by Daniel Gohlke (CaBi Tracker)

- Predictive Modeling
    - Can demand be predicted? How well?
    - What data can be used for prediction? Which predictors are informative?
    - How should demand be binned or quantified?
    - Which method works best?

Data and Problem Formulation

Available Data

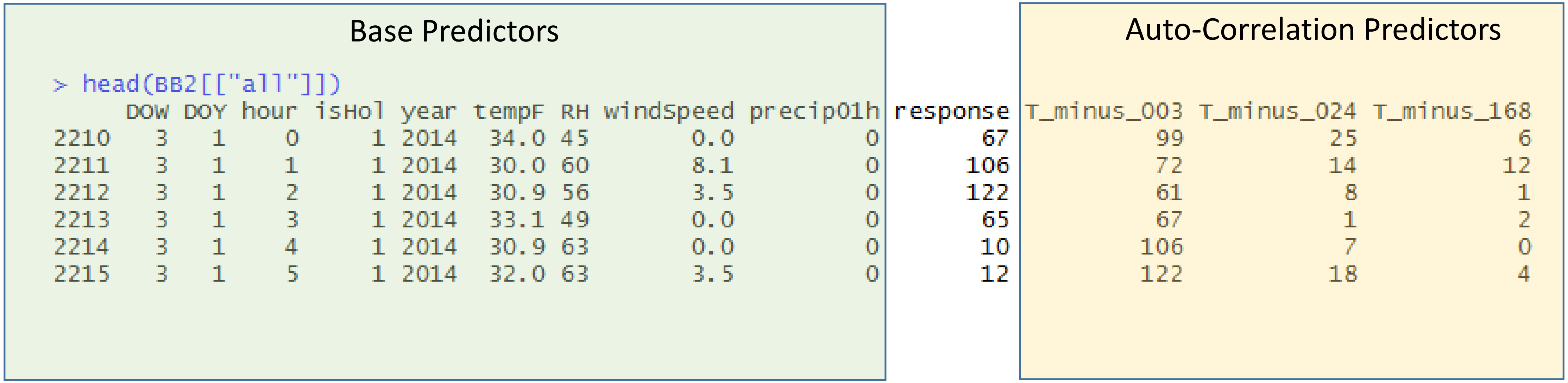
- Trip History (CaBi)
  - List of 12M rides 2010-2015
  - Begin/End Location/Time, Member Type
- Weather (DCA)
- Geographic info (multiple sources)
  - Station Lat/Long/Elevation
  - Jurisdiction / Neighborhood / etc.
- Dock Status History (webscraped, not used)

How to select predictors?

- Remove variables that don’t seem relevant (e.g. Bike #)
- Remove highly correlated predictors (e.g. 6hPrecip, dewpoint)
- Add temporal predictors that seem relevant (e.g. Day of Week)
- Try adding observations of response in previous time bins

How to represent response (demand)?

- Poisson process: rapidly drifting arrival rate
- Choose 2014 and 2015 only
- Bin by hour: 2 years = 17520 obs for each:
  - Station (360)
  - Member type (2)
  - Demand type (2)



Base Predictors

Auto-Correlation Predictors

```
> head(BB2[["all"]])
     DOW DOY hour isHol year tempF RH windSpeed precip01h response T_minus_003 T_minus_024 T_minus_168
2210   3   1    0     1 2014  34.0 45       0.0         0       67          99          25           6
2211   3   1    1     1 2014  30.0 60       8.1         0      106          72          14          12
2212   3   1    2     1 2014  30.9 56       3.5         0      122          61           8           1
2213   3   1    3     1 2014  33.1 49       0.0         0       65          67           1           2
2214   3   1    4     1 2014  30.9 63       0.0         0       10         106           7           0
2215   3   1    5     1 2014  32.0 63       3.5         0       12         122          18           4
```

Model Select: Random Forest w/o Auto-Corr

System-wide bicycle demand,
No auto-correlation predictors:

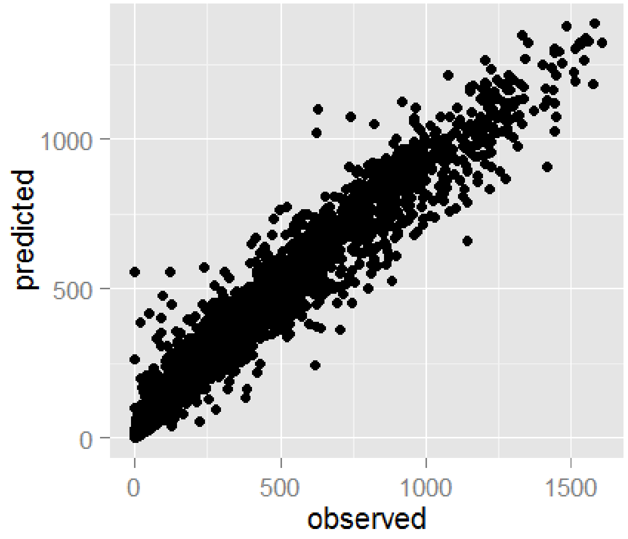
		XVAL	XVAL	TRAIN	TEST
	Tuned Params	**RMSE**	**SD**	**RMSE**	**RMSE**
KNN	k=5	183	4	145	178
MARS	nprune=18,degree=2	174	5	173	171
Neural Net	size=8,decay=0	225	13	230	229
Random Forest	not tuned: mtry=3	91	2	44	82
SVM	sigma=1,C=2	162	6	121	158

		XVAL	XVAL	TRAIN	TEST
	Tuned Params	**R2**	**SD**	**R2**	**R2**
KNN	k=5	0.71	0.01	0.82	0.72
MARS	nprune=18,degree=2	0.74	0.01	0.74	0.75
Neural Net	size=8,decay=0	0.56	0.05	0.54	0.55
Random Forest	not tuned: mtry=3	0.93	0.00	0.99	0.95
SVM	sigma=1,C=2	0.78	0.01	0.88	0.79

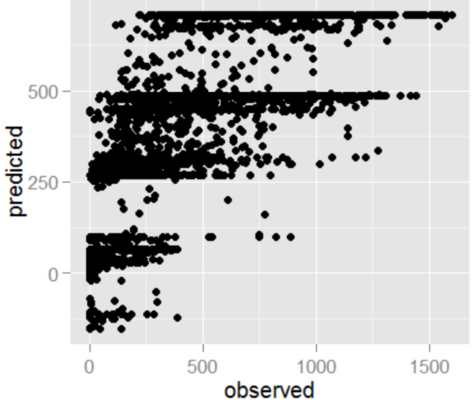
System-wide bicycle demand,
With auto-correlation predictors:

		XVAL	XVAL	TRAIN	TEST
	Tuned Params	**RMSE**	**SD**	**RMSE**	**RMSE**
Random Forest	not tuned: mtry=4	89	2	37	81
SVM	sigma=0.1,C=2	92	4	82	88
		R2	**SD**	**R2**	**R2**
Random Forest	not tuned: mtry=4	0.93	0.00	0.99	0.94
SVM	sigma=0.1,C=2	0.93	0.01	0.94	0.93

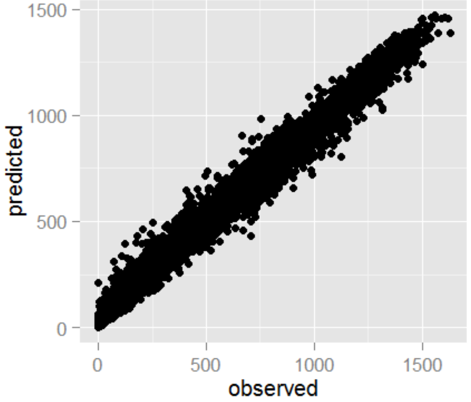
Test Set Performance, Random Forest w/o auto-corr:



Test Set Performance, NN w/o auto-corr:



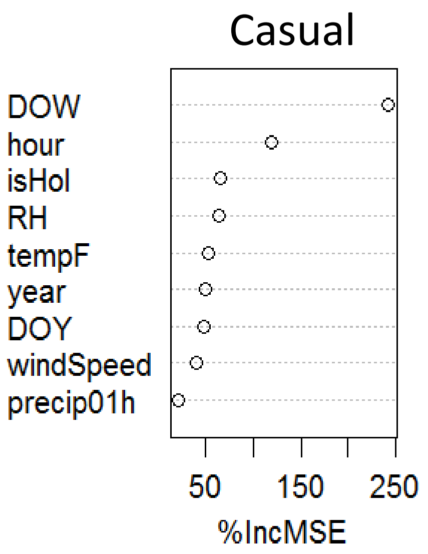
Train Set Overfit, Random Forest w/o auto-corr:

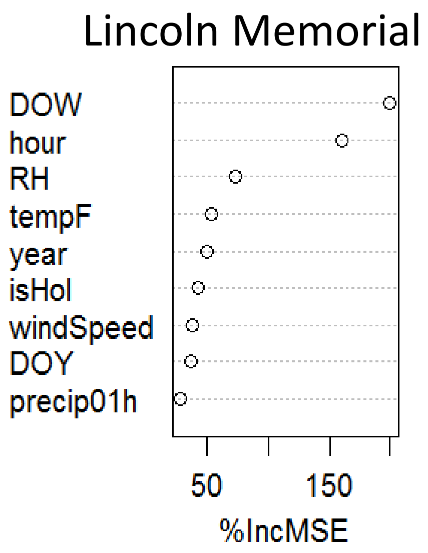


Apply Random Forest to subset Predictions

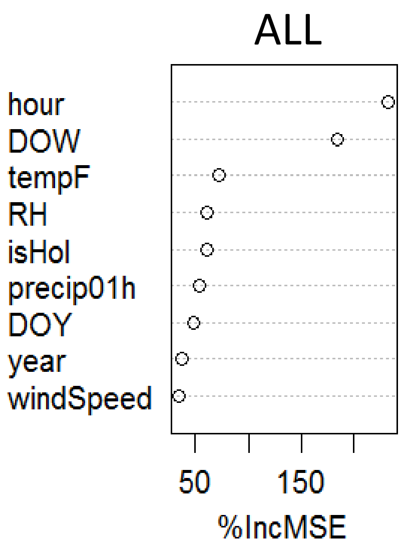
System-wide bicycle demand,
No auto-correlation predictors:

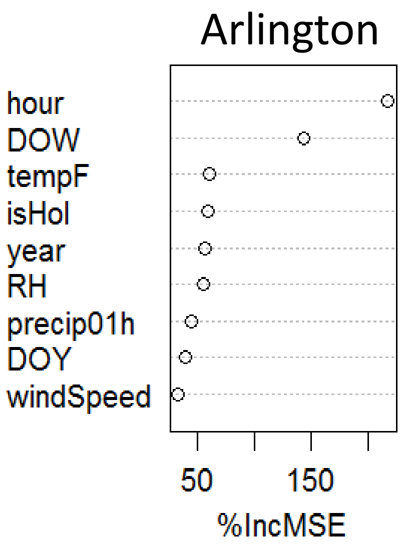
	XVAL R2	XVAL SD	TRAIN R2	TEST R2
all	0.93	0.003	0.99	0.95
casual	0.91	0.006	0.98	0.93
member	0.93	0.004	0.99	0.95
Arlington	0.89	0.004	0.98	0.91
Lincoln Memorial	0.76	0.007	0.94	0.78
Dupont Circle	0.75	0.006	0.94	0.77

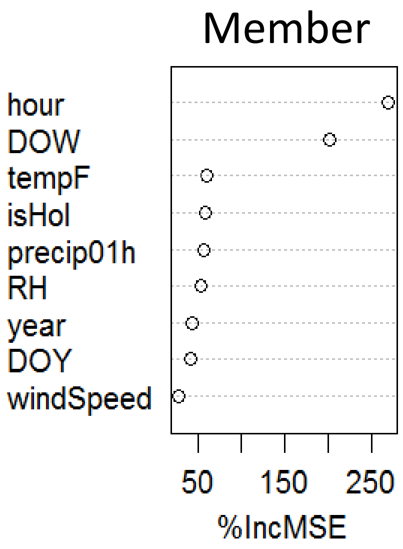

Casual

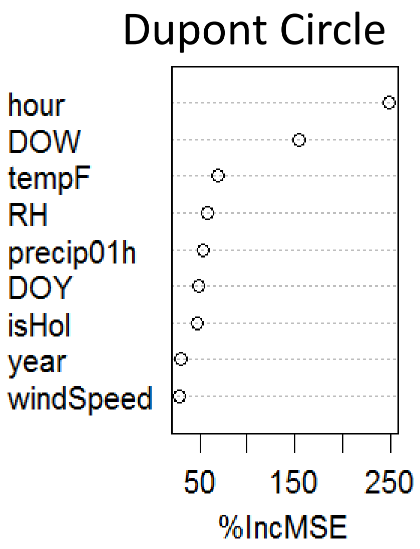

Lincoln Memorial

Hour is most informative predictor for member ridership; day of week is most informative for casual ridership.

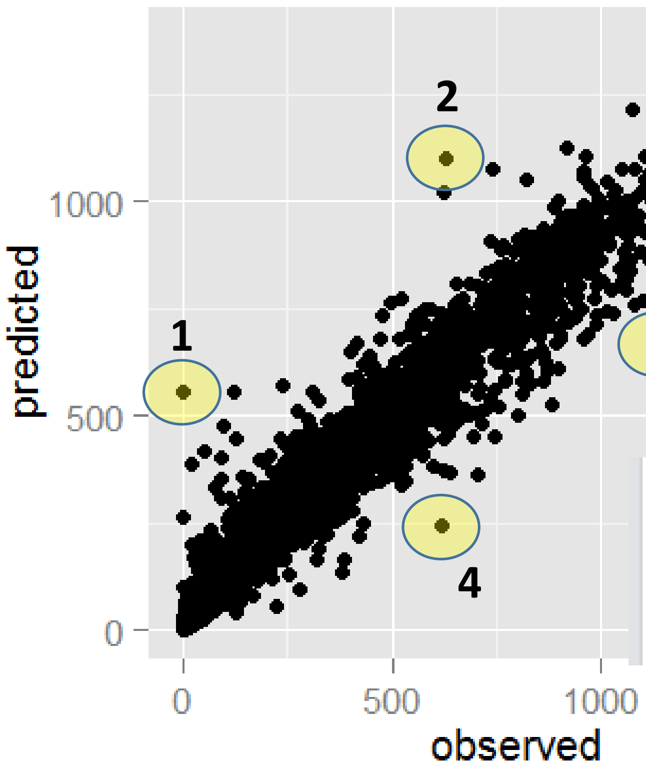

ALL


Arlington


Member

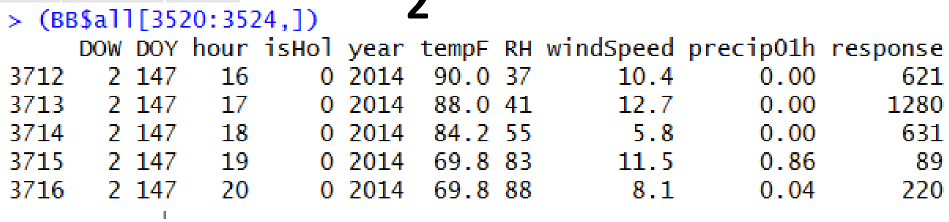

Dupont Circle

Investigate Worst Misses in Test Set



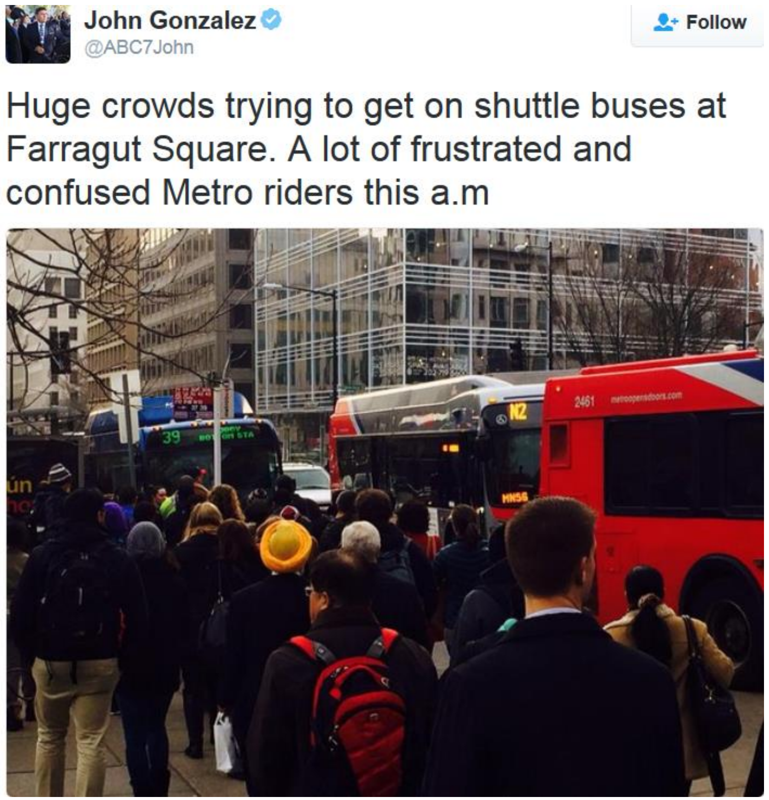
1

2



```
> (BB$all[3520:3524,])
     DOW DOY hour isHol year tempF RH windSpeed precip01h response
3712   2 147   16     0 2014  90.0 37      10.4      0.00      621
3713   2 147   17     0 2014  88.0 41      12.7      0.00     1280
3714   2 147   18     0 2014  84.2 55       5.8      0.00      631
3715   2 147   19     0 2014  69.8 83      11.5      0.86       89
3716   2 147   20     0 2014  69.8 88       8.1      0.04      220
```

3



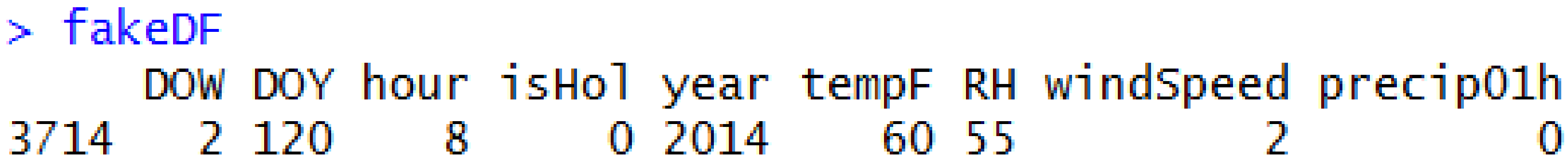
John Gonzalez
@ABC7John

Huge crowds trying to get on shuttle buses at Farragut Square. A lot of frustrated and confused Metro riders this a.m

	Date	Time	Predicted	Observed	Comment
1	FRI Feb 14 2014	8 am	555	0	Day after 6" snow + ice, OPM 2-hour delay (snowDepth predictor removed!)
2	TUE May 27 2014	6 pm	1099	631	right before major thunderstorm (time bins may be too large to capture this effect)
3	TUE Dec 16 2014	8 am	660	1141	water main break at 12th and F, WMATA suspended Silver/Orange/Blue service
4	THU Oct 01 2015	5 pm	247	620	Hurricane Joaquin approaching, rain earlier much of day

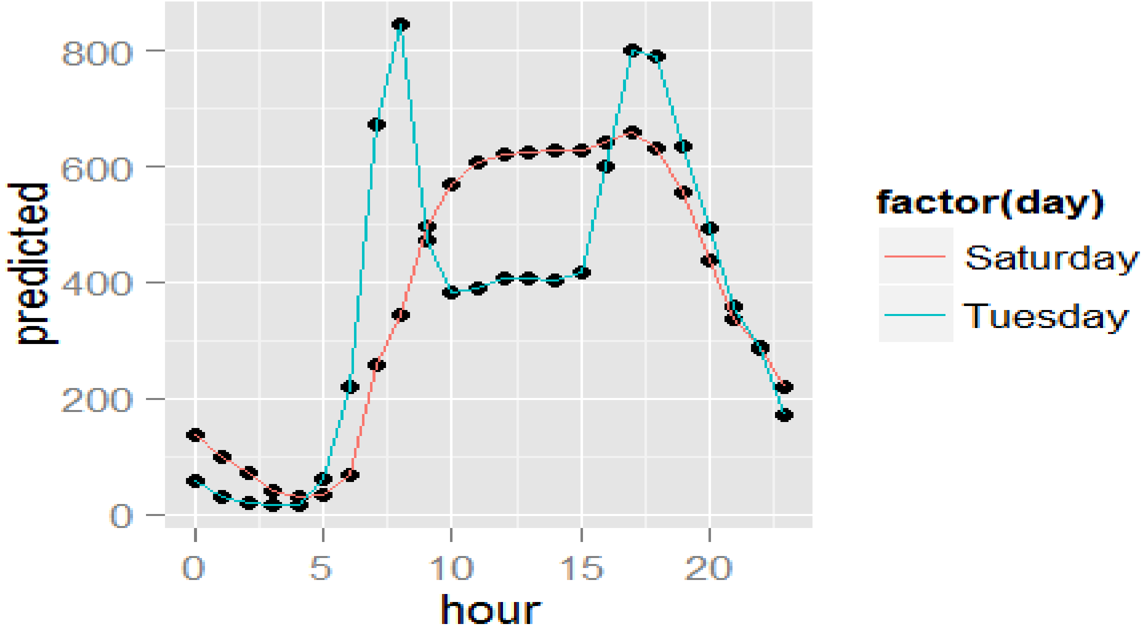
Do regression tendencies make sense?

- I simulated data to isolate certain parameters one at a time
- Base:

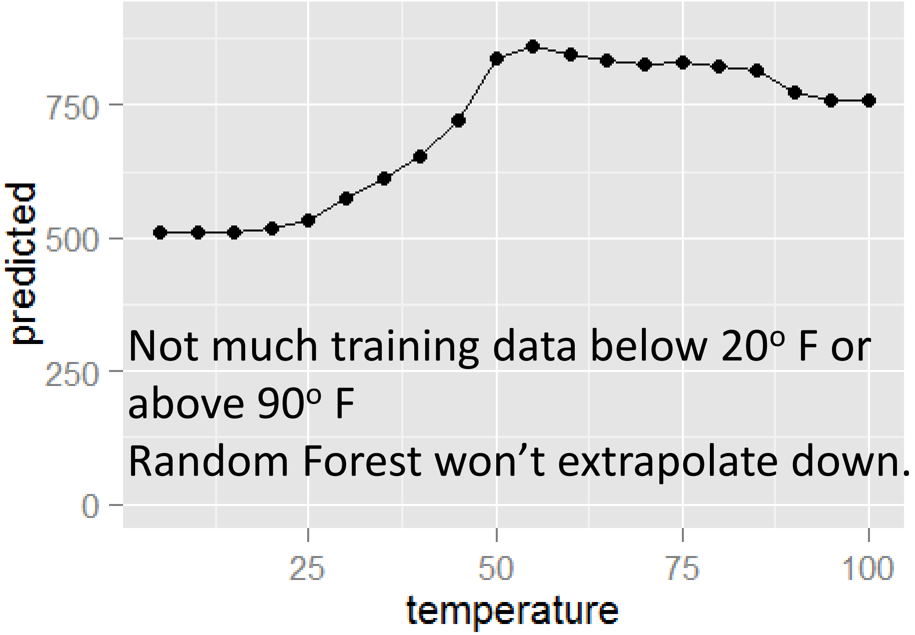


```
> fakeDF
     DOW DOY hour isHol year tempF RH windSpeed precip01h
3714   2 120    8     0 2014    60 55         2         0
```

Now vary hour and weekday:



Vary temperature:



Not much training data below 20º F or above 90º F

Random Forest won’t extrapolate down.

Summary

- Random Forest model predicts system-level bikeshare demand quite well
- Tougher to predict at micro-level: SNR is much lower
- Most important predictors came out as expected (hour, day of week)
- Poor predictions satisfactorily explained by information beyond model
- Potential improvements
    - Better demand metric: account for dockblocked times
    - More predictors (e.g. snow depth)
    - Geographic considerations
        - Treat station locations as numeric lat/long rather than categorical?
        - Elevation
    - Time Series techniques?

Backup Slides: Background info

Backup: Data Sources Used

- Capital Bikeshare Trip History
  - https://www.capitalbikeshare.com/trip-history-data
  - Click on “Download Links”
  - Unzip to extract CSVs
  - The file names and formats are inconsistent from file to file.
    - The Python code will standardize the data formats, but before running it, you must change all file names to the following naming convention: “2015-Q2-cabi-trip-history-data.csv”. The Python code is expecting that name format so that it knows which files to open.
- Weather
  - http://mesowest.utah.edu/cgi-bin/droman/download_ndb.cgi?stn=KDCA
  - I downloaded annual files, and manually converted them to CSV
  - Result: KDCA___csv_mesowest_2010-2015.csv
- Geographic Info
  - Station TerminalName/Name/Lat/Long obtained from https://www.capitalbikeshare.com/data/stations/bikeStations.xml
  - Station Elevations obtained from Google API at http://www.gpsvisualizer.com/geocoder/elevation.html
  - Jurisdiction and neighborhood added manually
  - First use and last use obtained from trip history data
  - Result: stationInfo_v8.csv
- Holiday dates
  - https://gist.github.com/shivaas/4758439
  - Added 2010 and 2011 manually
  - Result: holidays2010on.csv

Backup: Data Munging

- Python scripts convert data from original format into a SQL database
- R codes begin from SQL database
- Notes
  - Python 2.7
  - Must put all files into one directory
  - Must change CaBi trip history file names to the format indicated on slide 11
  - Line 17 of cabi_Main.py: selects which trip history files to include
    - Current selection (134,154) selects files between 2013 Q4 and 2015 Q4
    - First two digits of argument = last two digits of year
    - Last digit of argument = quarter number
- Predictive Modeling
  - Selected data set only from years 2014-2015
  - All models used preprocessing (BoxCox/center/scale) except for Random Forest
  - Discarded many weather fields immediately
  - Eventually also discarded snowDepth due to near-zero variance
  - Also Discarded precip06h and dewpointF due to high correlations with other predictors