Taxi trip counts (2019 December excluding holiday season)

All: 6896317 8-9am: 294597 within Manhattan: 255183 After cleaning: 237727

Pick 10000 randomly from 237727 do ->	SLR	SLR + log	MLR v4	MLR v6	Binary Tree	Hybrid: MLR v2 + Tree
Model code	1	3	v4	v6	binTr2	hy2
R^2	0.5759	0.6739	0.7334	0.7338		0.7327 (MLR v2)
SSLF	2.41x10^8	147	263	261		132.34 (MLR v2)
Test for beta0=0	<2e-16	<2e-16	0.777	0.767		< 2e-16 (MLR v2)
Test for longitude				** for both		
cp					0.01	0.01 (Tree)
In-sample MSE	0.1882	0.1368	0.1118	0.1117	0.1233	0.1009 (Total)

SLR: trip_time ~ trip_distance

SLR + log: log(trip_time) ~ log(trip_distance)

Exhaustive search & stepwise regression on 6 variables suggest MLR v4 (no longitude) & v6 (use all)

MLR v4: log(trip_time) ~ log(trip_distance)+ trip_weekday (factor 0-6) + pickup_latitude + dropoff_latitude

MLR v6: log(trip_time) ~ log(trip_distance)+ trip_weekday (factor 0-6) + pickup_latitude + dropoff_latitude

+ pickup_longitude + dropoff_longitude

Because of higher SSLF, we may need spline/nonlinear functions for geo variables?

Try binary tree?

Binary Tree: use same set of predictors as MLR v6

Or hybrid?

MLR v2: log(trip_time) ~ log(trip_distance)+ trip_weekday (factor 0-6)

MLR v2 + Tree: get residual from MLR v2, fit residuals by 4 geometric variables only using binary tree

Tree splits from MLR v2 + Tree on actual map: (left – pickup, right – drop off)

Given distance and weekday, time may still variate based on pickup locations from the 4 rectangles on the left figure and drop off locations from the 6 rectangles on the right figure, within Manhattan.