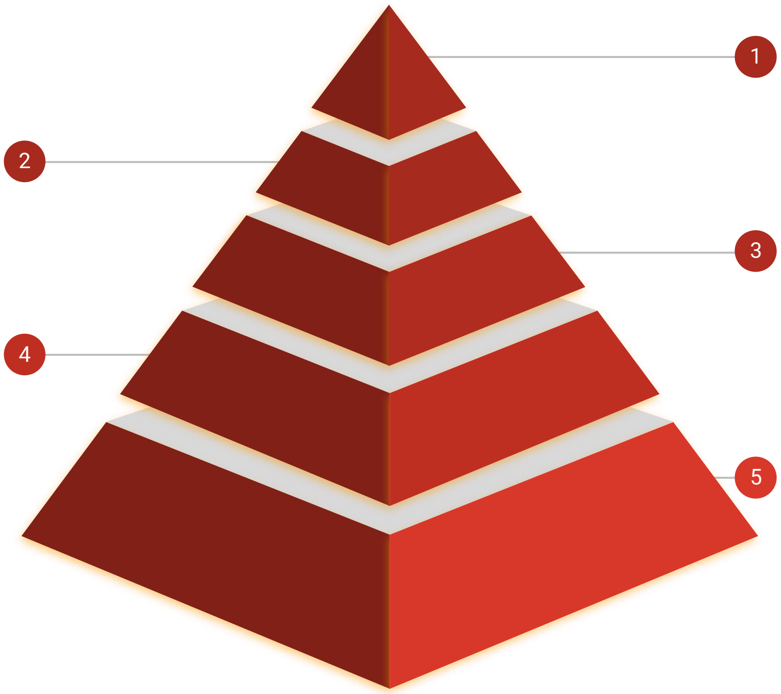
TEXAS
The University of Texas at Austin

Why you Bikein'?

TEAM

Mihir Deshpande
Eric Wang
Sreekar Lanka
Akhila Guttikonda
Parthiv Borgohain

Agenda



1

Problem Statement

Predict the demand of bike users based on the number of rides w.r.t casual and registered users.

2

Data Explanation

A normalized set of 17580 rows having number of casual and registered rides based on day of the week, month, years season and multiple weather variables.

3

Exploratory Data Analysis

Analyzing the number of rides based on categorical variables (day of the week, season and weather variables)

4

Data Preprocessing

One Hot encoding - data transmitted into 0 and 1 to turn categorical variables to numerical variables

5

Model Selection

Non-parametric and parametric model selection : Multiple regression, Trees and Knn

Problem Statement

Bike sharing is one of the most accessible way to commute for employees and students. Lime and Bird were the commonly used bikes in Austin especially for student commute and it is interesting to see how the membership, payment and trips have become more automatic.

Inspired by these observations, we took the trips data from Capital Bikeshare[1] to predict the demand of bike users based on the number of rides pertaining to registered and casual users.

[1] Capital Bikeshare – State owned bike sharing system in counties around Washington D.C and Virginia

Data Explanation

Data set : 17380 rows

Processing : One hot encoding to convert data from categorical to numerical as we are predicting a quantitative factor. Also removed highly correlated variables to reduce the RMSE.

Training 50% | Testing 25% | Validation 25%

- instant: record index
- dteday : date
- season : season (1:winter, 2:spring, 3:summer, 4:fall)
- yr : year (0: 2011, 1:2012)
- mnth : month (1 to 12)
- hr : hour (0 to 23)
- holiday : weather day is holiday or not (extracted from [Web Link])
- weekday : day of the week
- workingday : if day is neither weekend nor holiday is 1, otherwise is 0.

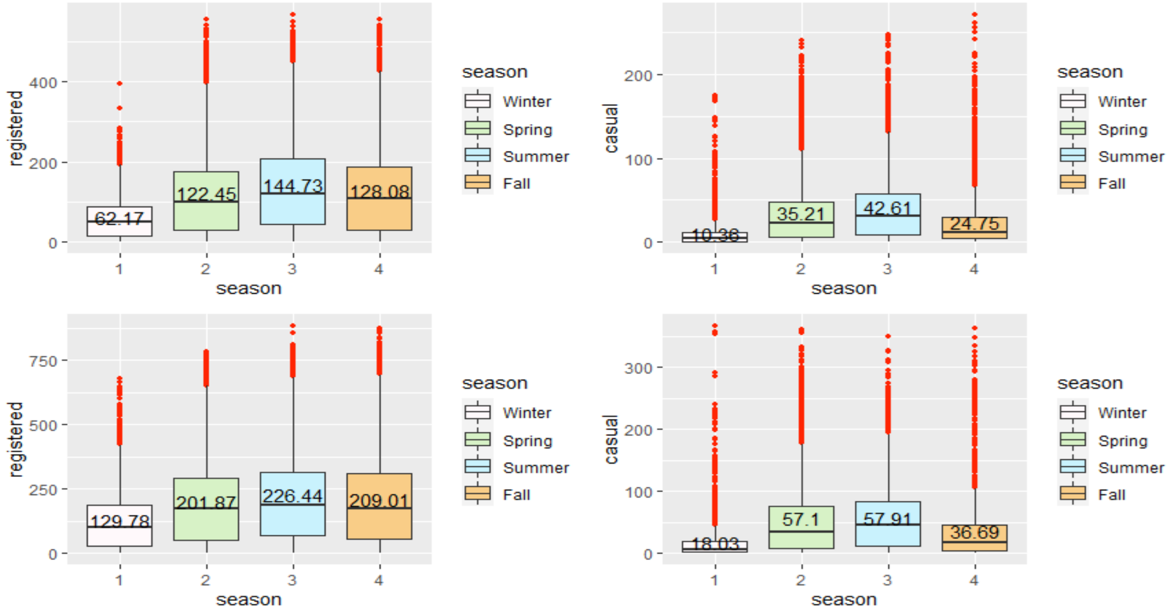
+ weathersit :
- 1: Clear, Few clouds, Partly cloudy, Partly cloudy
- 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
- 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
- 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- temp : Normalized temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min=-8, t_max=+39 (only in hourly scale)
- atemp: Normalized feeling temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min=-16, t_max=+50 (only in hourly scale)
- hum: Normalized humidity. The values are divided to 100 (max)
- windspeed: Normalized wind speed. The values are divided to 67 (max)
- casual: count of casual users
- registered: count of registered users
- cnt: count of total rental bikes including both casual and registered

Dataset obtained from University of California, Irvine archive (2nd hand) - https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset
Original data (1st hand) sourced from Capital Bikeshare system data archive - https://ride.capitalbikeshare.com/system-data

Exploratory Data Analysis

2011

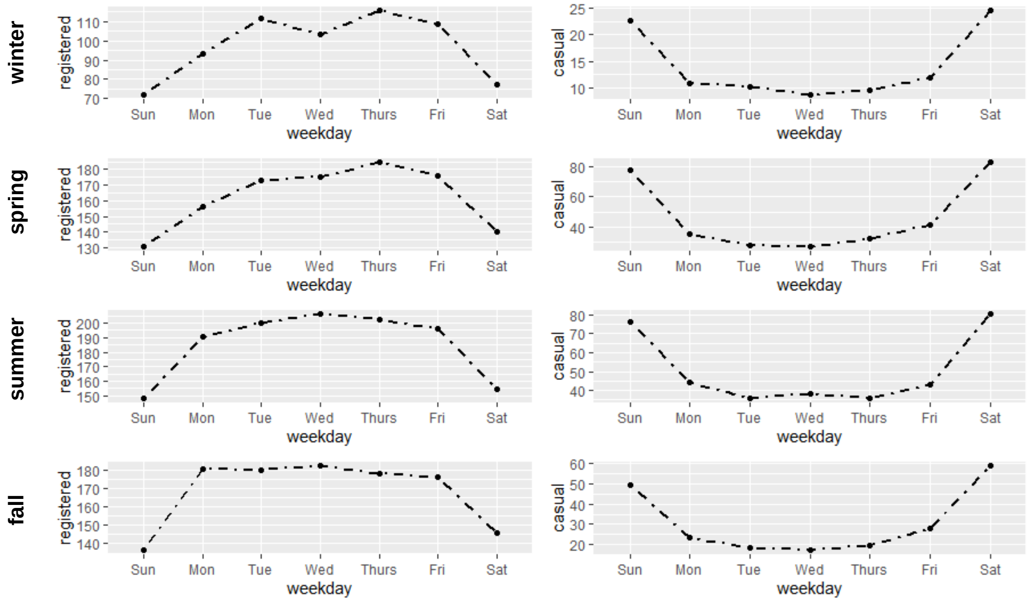
2012



Season wise registered and casual rides per hr

- Both registered and casual users had increased number of rides in Summer and Spring
- Large range of outliers for casual users while the registered users have a small range with higher magnitude

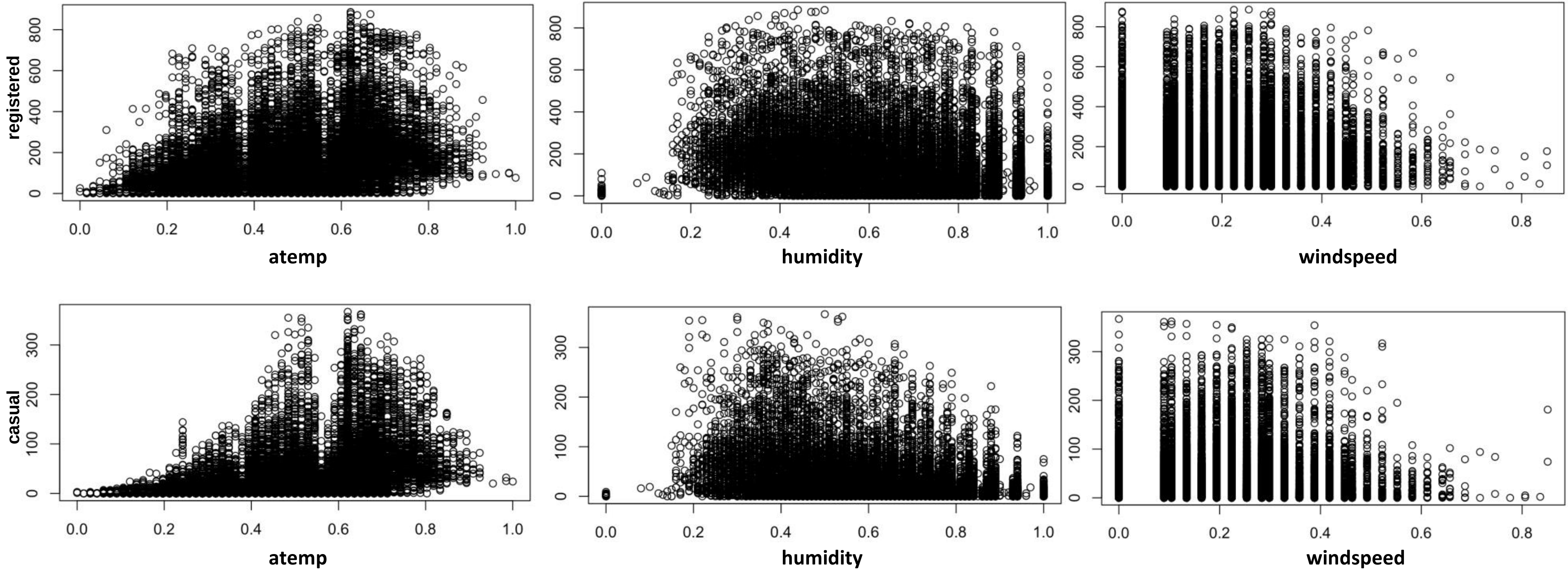
Exploratory Data Analysis



No. of rides for registered and casual across the week w.r.t seasons

- A contrasting pattern in the number of rides observed for both registered and casual riders
- Increased number of rides over the week and decreased number of rides over the weekends for registered users
- Decreased number of rides in weekdays and increased number of rides over the weekends for casual users

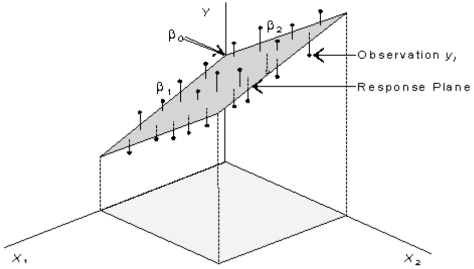
Exploratory Data Analysis



Approach

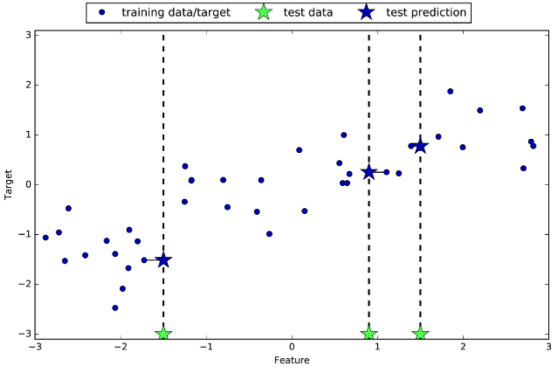
- Build separate models for *registered* users and *casual* users

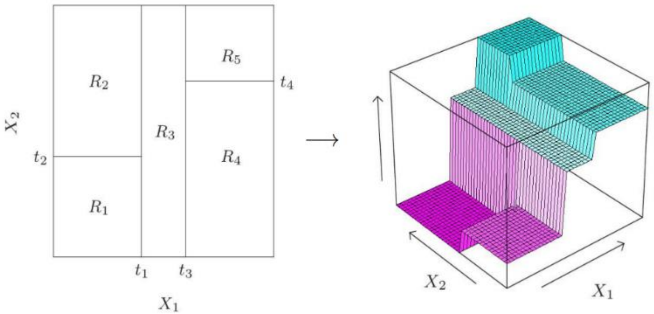
3 approaches to consider – Regression, K Nearest Neighbours, Trees



Multilinear Regression
+
(Ridge and Lasso)

K Nearest Neighbours





Tree Regression

Multilinear Regression

Data Pre-Processing

1) Encoding Categorical Variables
(converting factors to integer inputs)
Two methods –
Ordinal Encoding and OneHot Encoding

season = {1, 2, 3, 4} ->

season_1 = {0, 1}
season_2 = {0, 1}
season_3 = {0, 1}
season_4 = {0, 1}

e.g.

season	s_1	s_2	s_3	s_4
1	0	0	1	0

2) Removing unnecessary variables

- instant (serial no.), dte (date), yr (year), cnt (total rides per hour)

A) For 'registered' users

- casual (no. of casual rides per hour)

B) For 'casual' users

- registered (no. of registered rides per hour)

Standard 80-20 split for training and test/validation sets

Multilinear Regression

Registered Users

- Simple model with all predictors - `reg_mlr <- lm(registered~., data=newdata_tr)`

```
Coefficients: (6 not defined because of singularities)
```

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    78.7416    66.4693   1.185 0.236182
season_1      -64.8434     4.7986 -13.513  < 2e-16 ***
season_2      -35.1019     5.6315  -6.233 4.69e-10 ***
season_3      -36.5785     5.0584  -7.231 5.02e-13 ***
season_4            NA         NA      NA       NA
mnth_1          8.9759     4.8776   1.840 0.065754 .
mnth_2         11.5552     4.8883   2.364 0.018099 *
mnth_3          6.6308     4.8516   1.367 0.171736
mnth_4         -2.1317     6.2980  -0.338 0.735017
mnth_5          8.7006     6.6484   1.309 0.190667
mnth_6         -6.2026     6.5767  -0.943 0.345634
mnth_7        -26.5935     7.0080  -3.795 0.000148 ***
mnth_8         -4.7479     6.8233  -0.696 0.486548
mnth_9         16.9256     5.6348   3.004 0.002671 **
mnth_10         2.7937     4.2928   0.651 0.515195
mnth_11       -12.2396     4.0997  -2.985 0.002836 **
mnth_12             NA         NA      NA       NA
hr_0          -27.8180     5.2579  -5.291 1.24e-07 ***
hr_1          -41.9033     5.2408  -7.996 1.38e-15 ***
hr_2          -48.5245     5.3152  -9.129  < 2e-16 ***
hr_3          -57.7344     5.3240 -10.844  < 2e-16 ***
hr_4          -57.6386     5.3528 -10.768  < 2e-16 ***
hr_5          -42.8105     5.2923  -8.089 6.46e-16 ***
hr_6           12.0549     5.2765   2.285 0.022349 *
hr_7          138.2743     5.2635  26.270  < 2e-16 ***
hr_8          272.1059     5.2433  51.895  < 2e-16 ***
hr_9          115.1939     5.2509  21.938  < 2e-16 ***
hr_10          42.3958     5.2585   8.062 8.05e-16 ***
hr_11          56.8320     5.3169  10.689  < 2e-16 ***
hr_12          88.7857     5.3331  16.648  < 2e-16 ***
hr_13          81.1071     5.3573  15.139  < 2e-16 ***
hr_14          60.2285     5.3883  11.178  < 2e-16 ***
hr_15          73.7507     5.4346  13.570  < 2e-16 ***
hr_16         139.3247     5.4032  25.786  < 2e-16 ***
hr_17         289.5501     5.3563  54.058  < 2e-16 ***
```

```
hr_16          139.3247     5.4032  25.786  < 2e-16 ***
hr_17          289.5501     5.3563  54.058  < 2e-16 ***
hr_18          274.4671     5.3457  51.344  < 2e-16 ***
hr_19          173.6713     5.3159  32.670  < 2e-16 ***
hr_20          105.7933     5.2519  20.144  < 2e-16 ***
hr_21           64.2633     5.2572  12.224  < 2e-16 ***
hr_22           32.4652     5.2611   6.171 6.97e-10 ***
hr_23                NA         NA      NA       NA
holiday_0        9.4176     4.7327   1.990 0.046621 *
holiday_1            NA         NA      NA       NA
workingday_0   -39.1123     1.7000 -23.008  < 2e-16 ***
workingday_1         NA         NA      NA       NA
weathersit_1     2.8262    65.8346   0.043 0.965759
weathersit_2     0.4817    65.8311   0.007 0.994161
weathersit_3   -48.0169    65.8609  -0.729 0.465974
weathersit_4         NA         NA      NA       NA
temp           126.1760    28.3600   4.449 8.69e-06 ***
atemp           71.8406    29.3688   2.446 0.014450 *
hum            -81.2101     5.4268 -14.965  < 2e-16 ***
windspeed      -26.0292     6.9204  -3.761 0.000170 ***
---
```

Standard 80-20 split for training and test/validation sets

Multilinear Regression

Registered Users

What are singularities ?

- Singularity is the extreme form of multicollinearity - when a perfect linear relationship exists between variables or, in other terms, when the correlation coefficient is equal to 1.0 or -1.0

$$X_1 = a*X_2 + b*X_3 + c$$

- Usually happens when encoding dummy variables during data preprocessing

- Leads to incorrect calculations of coefficients with the OLS method

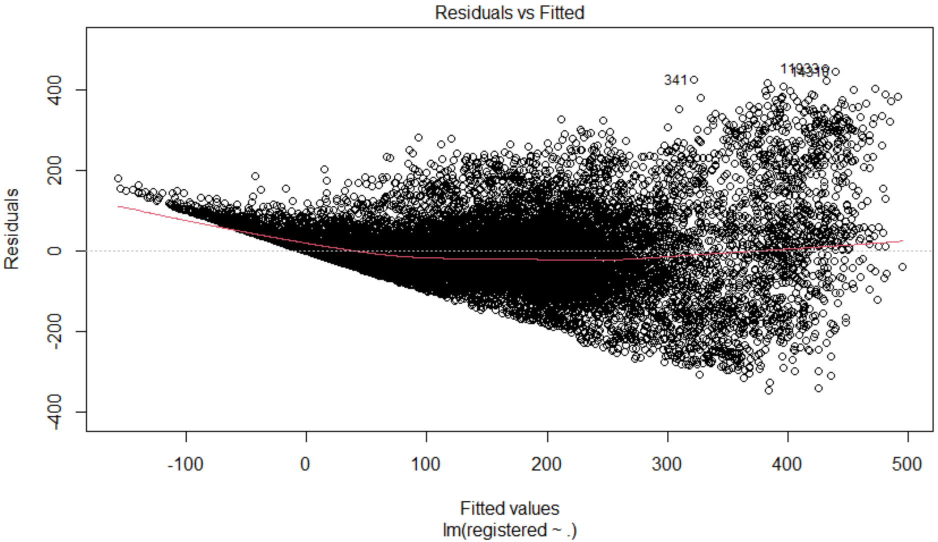
Multilinear Regression

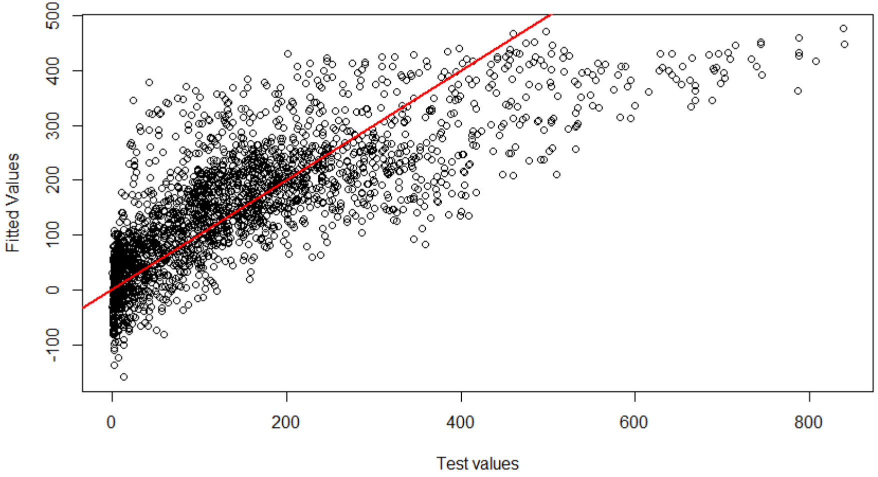
Registered Users

```
Residual standard error: 92.92 on 14953 degrees of freedom
Multiple R-squared:  0.6278,	Adjusted R-squared:  0.6267
F-statistic: 548.3 on 46 and 14953 DF,  p-value: < 2.2e-16
```

```
[1] "Root mean square error -"
> round(rmse.mlr,2)
[1] 92.92

[1] "R-squared value for test set -"
> round(r2.mlr,4)
[1] 0.599
```





Multilinear Regression

Registered Users

Other model building techniques to consider -

- Nth root model with all predictors – `reg_mlr_root <- lm(registered^(1/n)~., data=newdata_tr)`

- Log model with all predictors – `reg_mlr_log <- lm(log(registered+1)~., data=newdata_tr)`

  (Note we added a value of 1 to registered no. of riders per hour because we log(0) is not defined)

Standard 80-20 split for training and test/validation sets

Multilinear Regression

Registered Users

- Nth root model with all predictors - `reg_mlr_root <- lm(registered^(1/n)~., data=newdata_tr)`

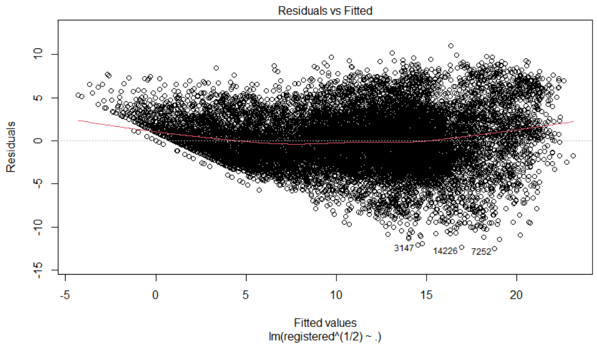
  N = 2

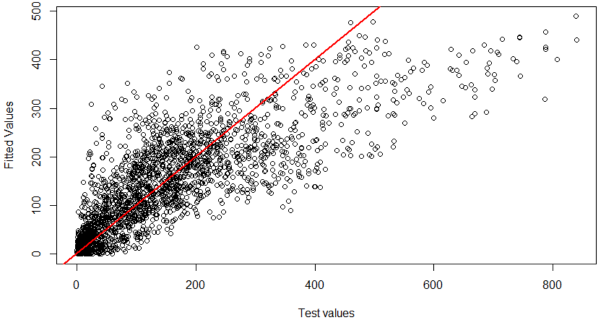
Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	9.2507104	2.2594427	4.094	4.26e-05	***
season_1	-2.8241070	0.1631162	-17.313	< 2e-16	***
season_2	-1.5211135	0.1914268	-7.946	2.06e-15	***
season_3	-1.3776772	0.1719479	-8.012	1.21e-15	***
mnth_1	0.3961497	0.1658008	2.389	0.01689	*
mnth_2	0.6695769	0.1661642	4.030	5.61e-05	***
mnth_3	0.3717264	0.1649185	2.254	0.02421	*
mnth_4	-0.0062021	0.2140834	-0.029	0.97689	
mnth_5	0.4967974	0.2259958	2.198	0.02795	*
mnth_6	-0.1336239	0.2235561	-0.598	0.55004	
mnth_7	-0.9911199	0.2382167	-4.161	3.19e-05	***
mnth_8	-0.2868708	0.2319411	-1.237	0.21617	
mnth_9	0.4191153	0.1915409	2.188	0.02868	*
mnth_10	-0.0006239	0.1459219	-0.004	0.99659	
mnth_11	-0.4122169	0.1393585	-2.958	0.00310	**
hr_0	-1.9880028	0.1787270	-11.123	< 2e-16	***
hr_1	-3.4349847	0.1781474	-19.282	< 2e-16	***
hr_2	-4.3414905	0.1806751	-24.029	< 2e-16	***
hr_3	-5.3698346	0.1809738	-29.672	< 2e-16	***
hr_4	-5.7047536	0.1819527	-31.353	< 2e-16	***
hr_5	-3.7953653	0.1798960	-21.098	< 2e-16	***
hr_6	-0.0080504	0.1793618	-0.045	0.96420	
hr_7	4.9269826	0.1789183	27.538	< 2e-16	***
hr_8	9.0629513	0.1782335	50.849	< 2e-16	***
hr_9	5.1011442	0.1784900	28.579	< 2e-16	***
hr_10	2.2436935	0.1787501	12.552	< 2e-16	***
hr_11	2.8478759	0.1807344	15.757	< 2e-16	***
hr_12	4.1110491	0.1812826	22.678	< 2e-16	***
hr_13	3.8317203	0.1821086	21.041	< 2e-16	***
hr_14	3.0323100	0.1831592	16.556	< 2e-16	***
hr_15	3.5645933	0.1847358	19.296	< 2e-16	***
hr_16	5.8082196	0.1836669	31.624	< 2e-16	***
hr_17	9.6926891	0.1820740	53.235	< 2e-16	***
hr_18	9.3604225	0.1817125	51.512	< 2e-16	***
hr_19	6.7713659	0.1806999	37.473	< 2e-16	***
hr_20	4.6199200	0.1785247	25.878	< 2e-16	***
hr_21	3.0905888	0.1787046	17.294	< 2e-16	***
hr_22	1.7510740	0.1788365	9.791	< 2e-16	***
holiday_0	0.6481452	0.1608760	4.029	5.63e-05	***
workingday_0	-1.1266033	0.0577858	-19.496	< 2e-16	***
weathersit_1	-1.2757949	2.2378693	-0.570	0.56862	
weathersit_2	-1.3349891	2.2377501	-0.597	0.55080	
weathersit_3	-3.5412173	2.2387635	-1.582	0.11372	
temp	4.7789876	0.9640217	4.957	7.22e-07	***
atemp	3.0449611	0.9983134	3.050	0.00229	**
hum	-2.8543112	0.1844680	-15.473	< 2e-16	***
windspeed	-1.0694743	0.2352411	-4.546	5.50e-06	***

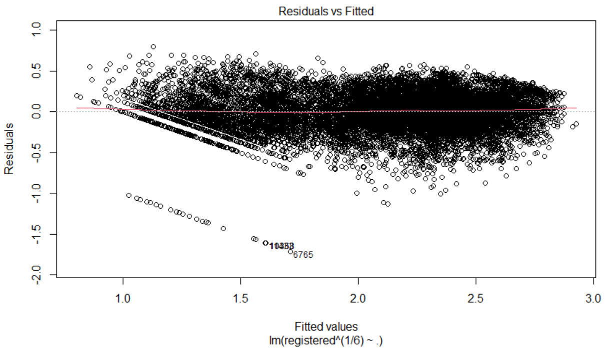
Standard 80-20 split for training and test/validation sets

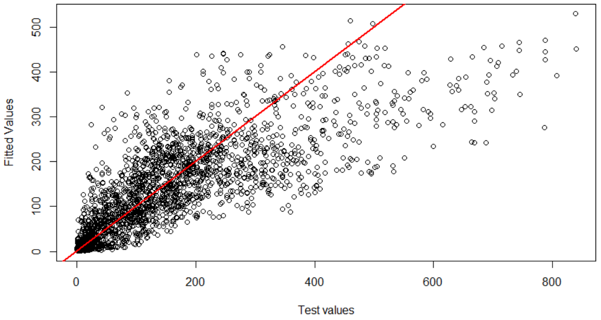
N = 2



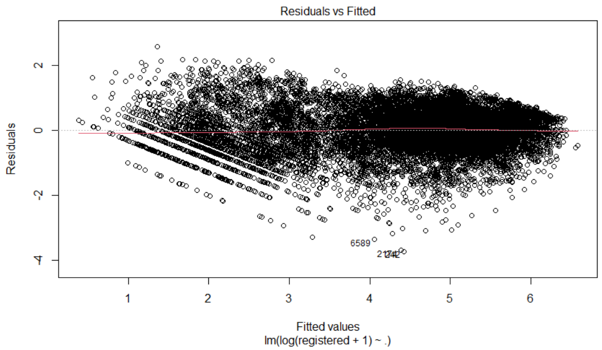


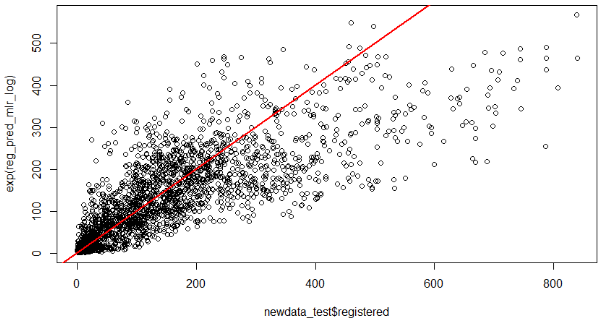
N = 6





Log Model





Multilinear Regression

Registered Users

Model (Regression)	RMSE (training)	RMSE (test)	R-square (training)	R-square (test)
Simple	92.921	92.92	0.6267	0.599
2nd Root	NA	88.99	0.729	0.632
6th Root	NA	90.62	0.7731	0.617
Log Model	NA	92.72	0.786	0.6008

Multilinear Regression

Casual Users

- Root (N=6) model with all predictors - `cas_mlr_root <- lm(casual^(1/6)~., data=newdata_tr)`

```
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.133118   0.224205   5.054 4.38e-07 ***
season_1    -0.119222   0.016186  -7.366 1.85e-13 ***
season_2     0.004937   0.018995   0.260 0.794935    
season_3    -0.021124   0.017062  -1.238 0.215712    
mnth_1      -0.065257   0.016452  -3.966 7.33e-05 ***
mnth_2      -0.001874   0.016488  -0.114 0.909525    
mnth_3       0.145727   0.016365   8.905  < 2e-16 ***
mnth_4       0.121930   0.021244   5.740 9.67e-09 ***
mnth_5       0.149956   0.022426   6.687 2.36e-11 ***
mnth_6       0.076181   0.022183   3.434 0.000596 ***
mnth_7       0.036126   0.023638   1.528 0.126463    
mnth_8       0.080506   0.023016   3.498 0.000470 ***
mnth_9       0.126893   0.019007   6.676 2.54e-11 ***
mnth_10      0.154256   0.014480  10.653  < 2e-16 ***
mnth_11      0.078886   0.013829   5.705 1.19e-08 ***
hr_0        -0.134661   0.017735  -7.593 3.31e-14 ***
hr_1        -0.314302   0.017678 -17.780  < 2e-16 ***
hr_2        -0.465062   0.017928 -25.940  < 2e-16 ***
hr_3        -0.681903   0.017958 -37.972  < 2e-16 ***
hr_4        -0.808220   0.018055 -44.764  < 2e-16 ***
hr_5        -0.691850   0.017851 -38.757  < 2e-16 ***
hr_6        -0.316069   0.017798 -17.759  < 2e-16 ***
hr_7        -0.013345   0.017754  -0.752 0.452273    
hr_8         0.192759   0.017686  10.899  < 2e-16 ***
hr_9         0.248392   0.017712  14.024  < 2e-16 ***
hr_10        0.301961   0.017737  17.024  < 2e-16 ***
hr_11        0.360564   0.017934  20.105  < 2e-16 ***
hr_12        0.389066   0.017989  21.628  < 2e-16 ***
hr_13        0.379683   0.018071  21.011  < 2e-16 ***
```

```
hr_12          0.389066   0.017989  21.628  < 2e-16 ***
hr_13          0.379683   0.018071  21.011  < 2e-16 ***
hr_14          0.381031   0.018175  20.965  < 2e-16 ***
hr_15          0.391224   0.018331  21.342  < 2e-16 ***
hr_16          0.391985   0.018225  21.508  < 2e-16 ***
hr_17          0.418805   0.018067  23.180  < 2e-16 ***
hr_18          0.357717   0.018031  19.839  < 2e-16 ***
hr_19          0.292608   0.017931  16.319  < 2e-16 ***
hr_20          0.207832   0.017715  11.732  < 2e-16 ***
hr_21          0.151586   0.017733   8.548  < 2e-16 ***
hr_22          0.098353   0.017746   5.542 3.04e-08 ***
holiday_0      0.070149   0.015964   4.394 1.12e-05 ***
workingday_0   0.258768   0.005734  45.128  < 2e-16 ***
weathersit_1 -0.198020   0.222064  -0.892 0.372555    
weathersit_2 -0.214215   0.222052  -0.965 0.334708    
weathersit_3 -0.442147   0.222153  -1.990 0.046578 *  
temp           0.488980   0.095660   5.112 3.23e-07 ***
atemp          0.512748   0.099063   5.176 2.30e-07 ***
hum           -0.174239   0.018305  -9.519  < 2e-16 ***
windspeed     -0.157970   0.023343  -6.767 1.36e-11 ***
---
```

Standard 80-20 split for training and test/validation sets

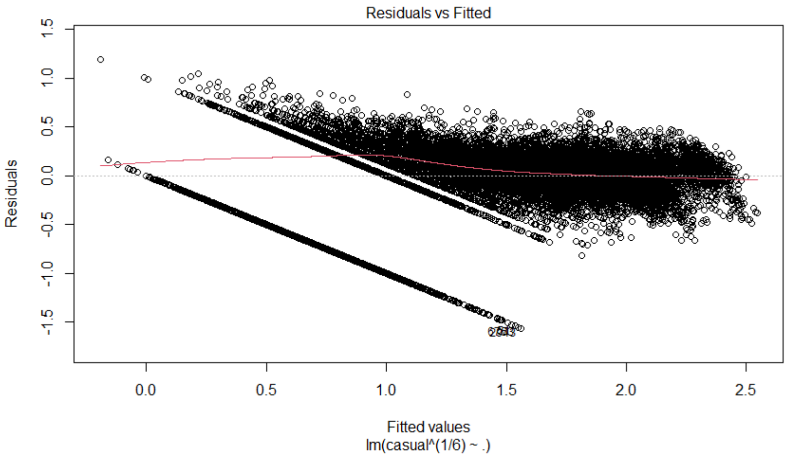
Multilinear Regression

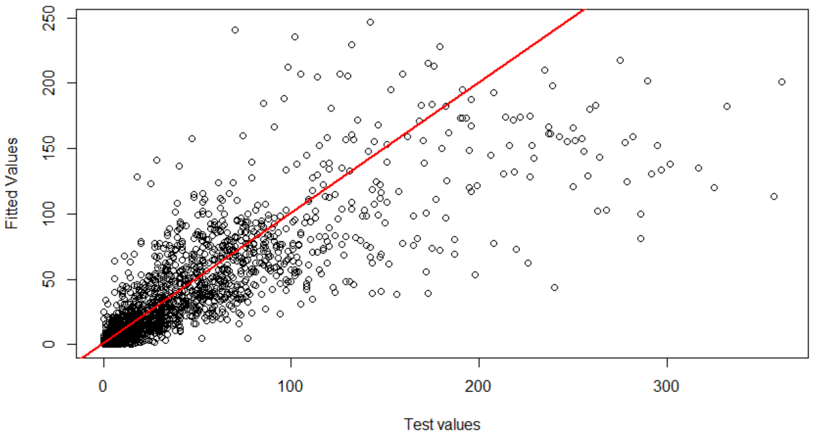
Casual Users

```
Residual standard error: 0.3134 on 14953 degrees of freedom
Multiple R-squared:  0.7246,	Adjusted R-squared:  0.7238
F-statistic: 855.3 on 46 and 14953 DF,  p-value: < 2.2e-16
```

```
[1] "Root mean square error -"
> round(rmse_lm_root_cas,2)
[1] 28.14

[1] "R-square value for test set -"
> round(rsq_lm_root_cas,4)
[1] 0.6847
```





Lasso and Ridge Regression

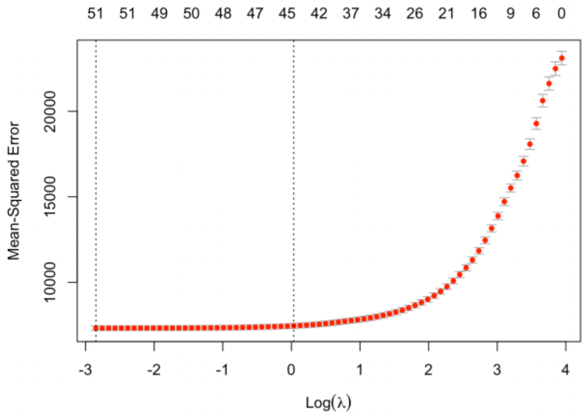
We tried to predict the registered users using regularization models - Lasso and Ridge regression

Lasso Regression

Training data
- **R square_test -** 0.68
- **RMSE_test** - 85.29

Test data
- **R square_test -** 0.66
- **RMSE_test** - 85.55

Lasso regression removes the predictors -
mnth_4 and weathersit_2

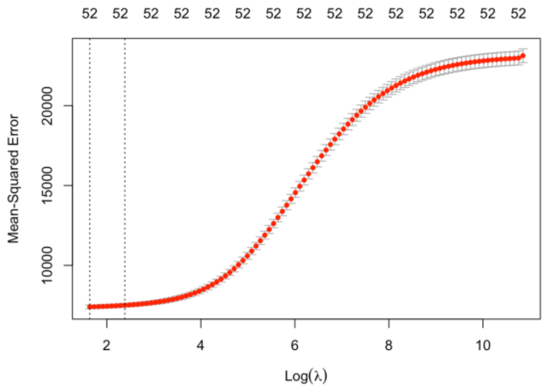


Ridge Regression

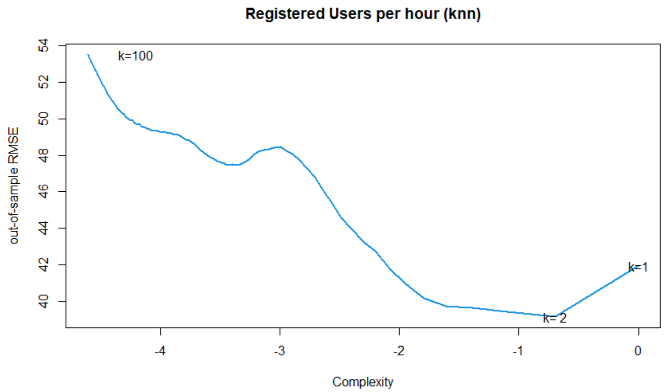
Training data
- **R square_test -** 0.68
- **RMSE_test** - 85.71

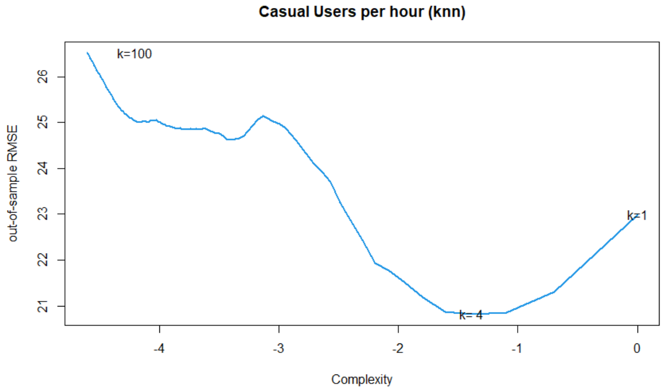
Test data
- **R square_test -** 0.65
- **RMSE_test** - 90.20

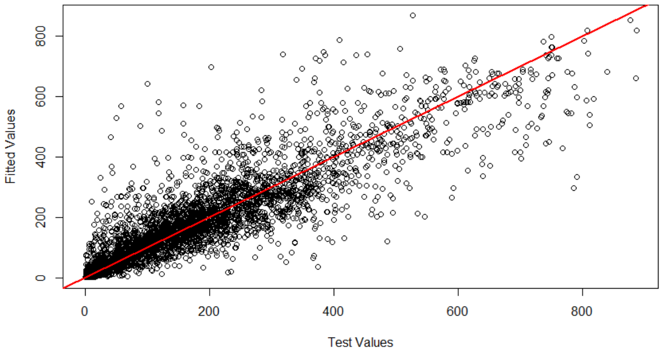
Ridge regression shrinks the following predictors to 0 -
mnth_3 and hr_22



K Nearest Neighbours







K Nearest Neighbours

- K Nearest Neighbours algorithm works on the principle of ‘distance’
- Measuring ‘distance’ is confusing when you introduce nominal categorical variables

How do you measure distance between “Sunny” and “Cloudy”?

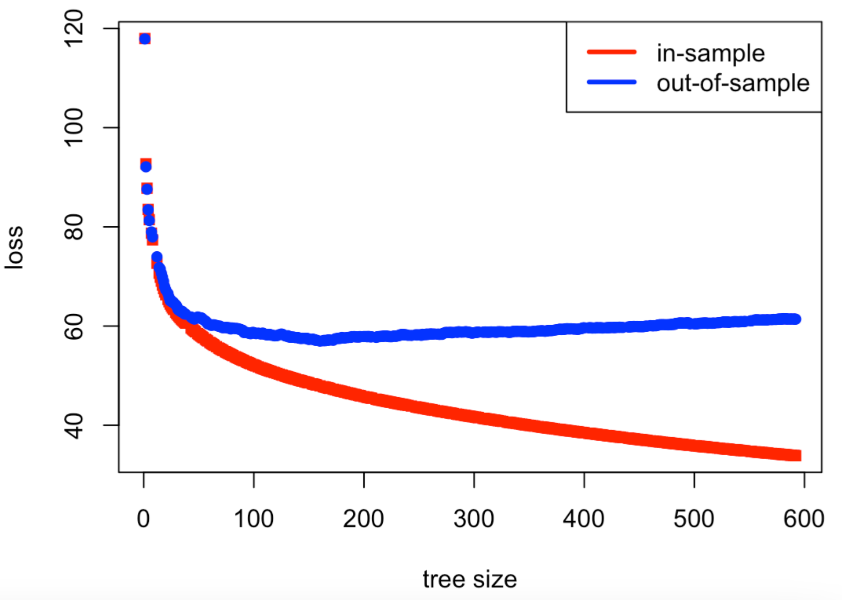
Compared to a temperature of 50 degrees, how far away is the month of May?

Trees

Fit a big tree using rpart

For Registered Users:

```
size of big tree:  592
```

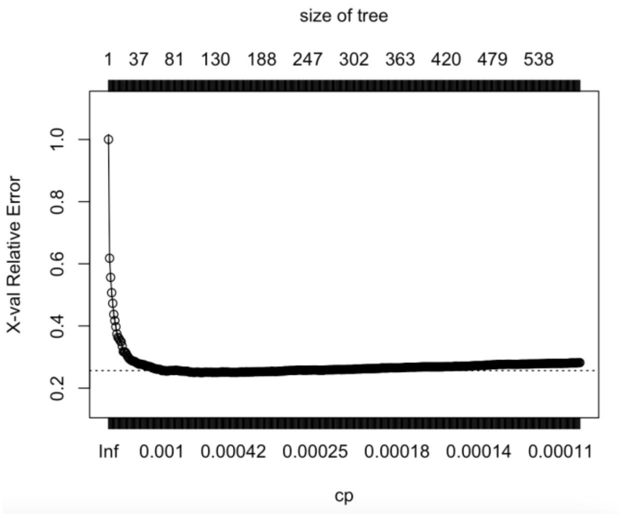



```
> mean(oltree)
[1] 60.08772
> mean(iltree)
[1] 44.58782
```

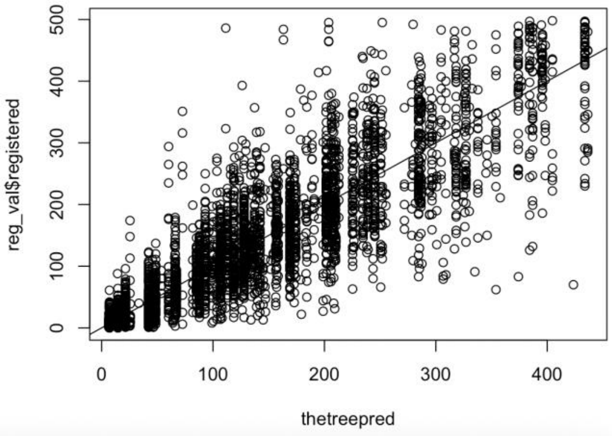
- We first fit a big tree to the training set through a very small CP value.
- Then we fit on the training set and predict on the validation set.
- Mean in sample loss came to be 44.58, but mean out of sample loss is **60.08!**

Trees

Pruning the tree



Pruned Tree RMSE for validation set : 56.9648



For Registered Users:

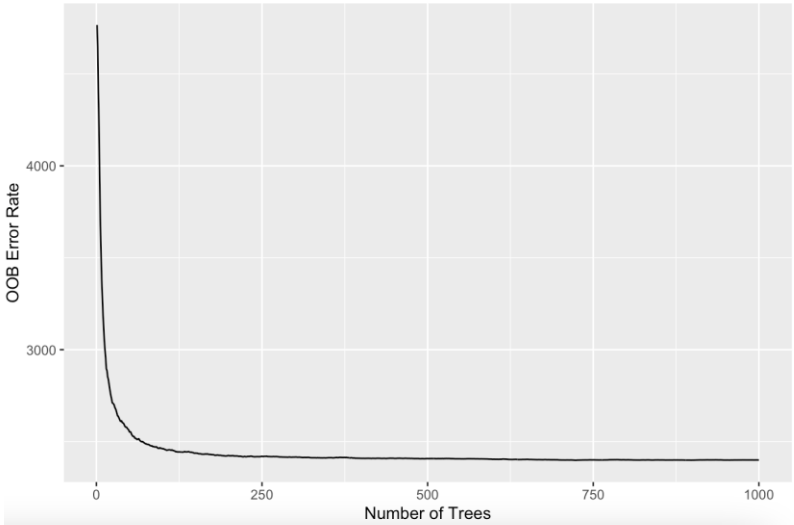
- Let’s look at the cross validation results from plotcp.
- Prune the tree using the CP value that gave us the lowest out of sample loss

76.64% Variance Explained

56.96 RMSE

Random Forests

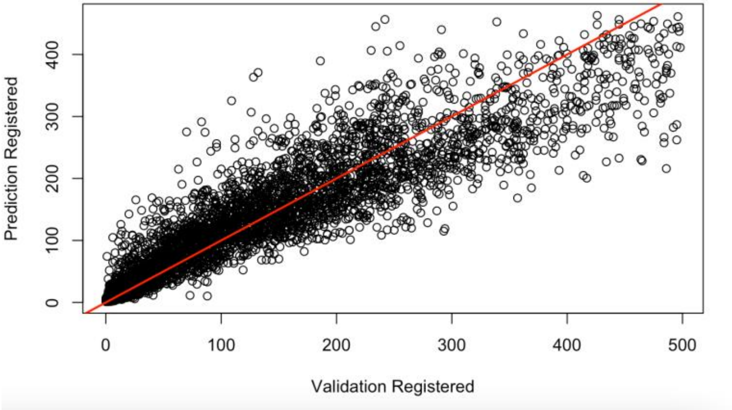
mtry	ntree	olrf	ilrf
11	200	49.488	49.697
3	200	53.757	53.736
11	500	$\boxed{49.449}$	49.583
3	500	53.661	53.683

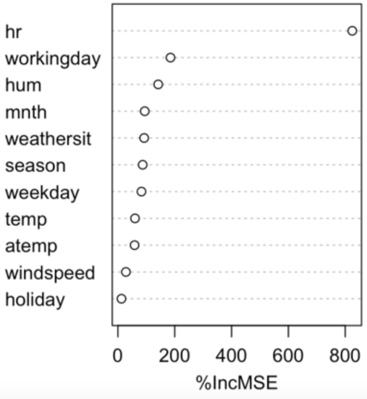


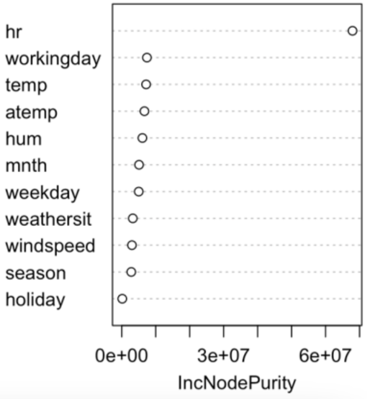
For Registered Users:

- Using random forests, we first tried 11 predictors (bagging) and 3 predictors with 200 and 500 trees.
- Minimum out of sample loss of **49.50** is much better than pruned tree
- Clearly, out of sample loss is lesser when we use the entire predictor set

Random Forests (Bagging)





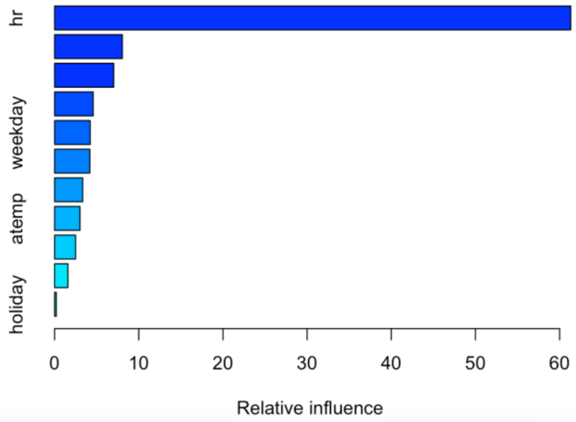


<u>**For Registered Users:**</u>

- Thus, we wanted to see if we can get better results by exploring more with bagging
- In fact we can! 1000 trees and 100 trees both gave us very similar **RMSE of ~48.8** and **~82.8% Variance Explained**
- Hr is clearly the most impactful variable

Boosting

tdepth	ntree	lam	olb	ilb
5	500	0.05	49.760	44.758
10	500	0.05	48.797	39.301
20	500	0.05	48.399	32.547
5	1000	0.05	49.386	40.935
10	1000	0.05	49.042	33.714
20	1000	0.05	48.878	25.338
5	2000	0.05	49.274	36.204
10	2000	0.05	49.113	27.062
20	2000	0.05	49.703	17.473
5	500	0.20	50.695	37.554
10	500	0.20	51.796	29.241
20	500	0.20	52.737	19.703
5	1000	0.20	51.979	33.069
10	1000	0.20	53.642	21.946
20	1000	0.20	53.162	11.817
5	2000	0.20	52.772	27.245
10	2000	0.20	54.334	14.450
20	2000	0.20	53.719	5.463



For Registered Users:

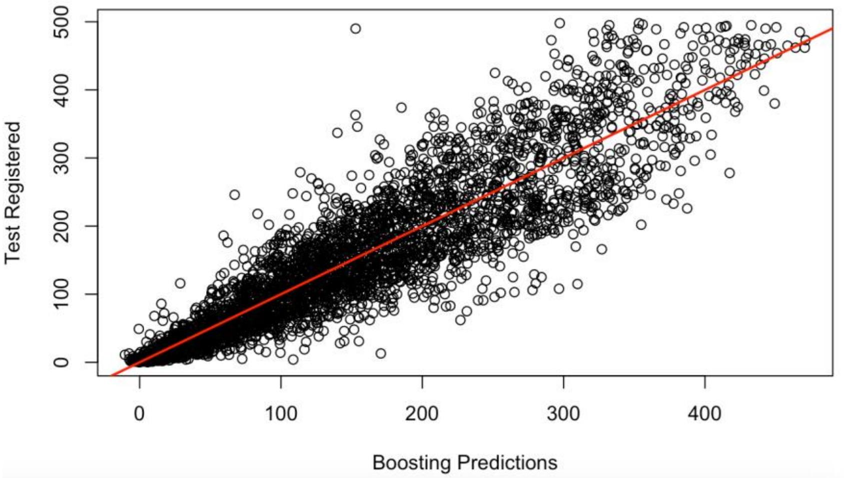
- We wanted to see if we could improve the results by tuning the parameters through boosting

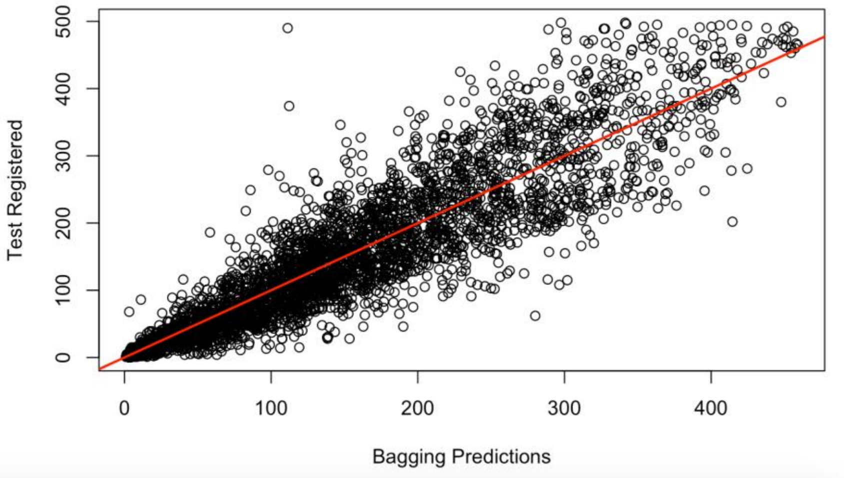
Best in sample loss: **5.463**

Best out of sample loss: **48.399**

Again, hr is the most important variable

Fit on Testing Set

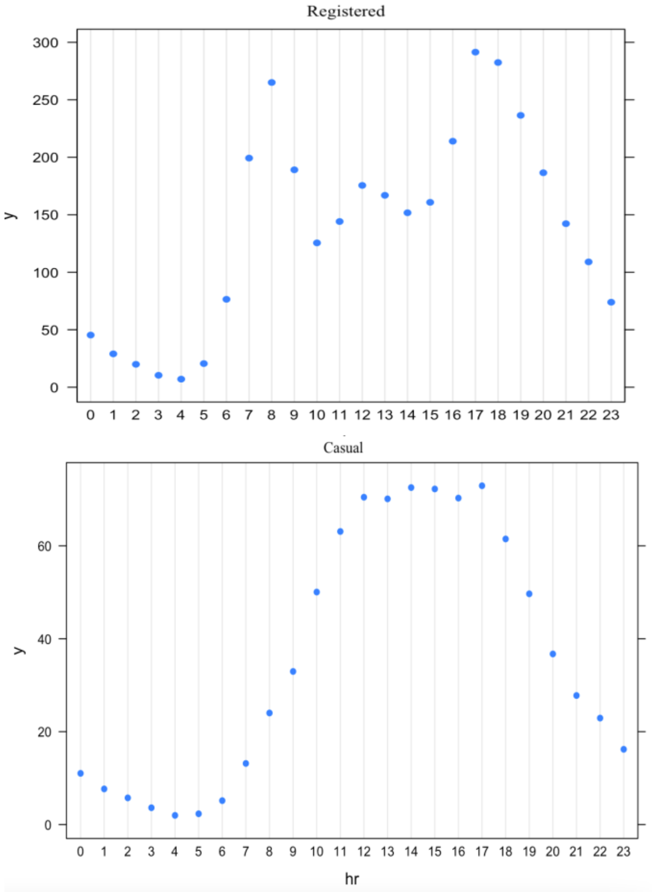




Fit on Testing Set

Registered Users	RMSE (Validation)	RMSE (Test)	R-square (Validation)	R-square (Test)
Pruned Tree	56.9	NA	76.6%	NA
Random Forest(Bagging)	48.8	46.2	82.2%	83.5%
Boosting	48.3	45.6	84.8%	84.4%

Casual User Predictions

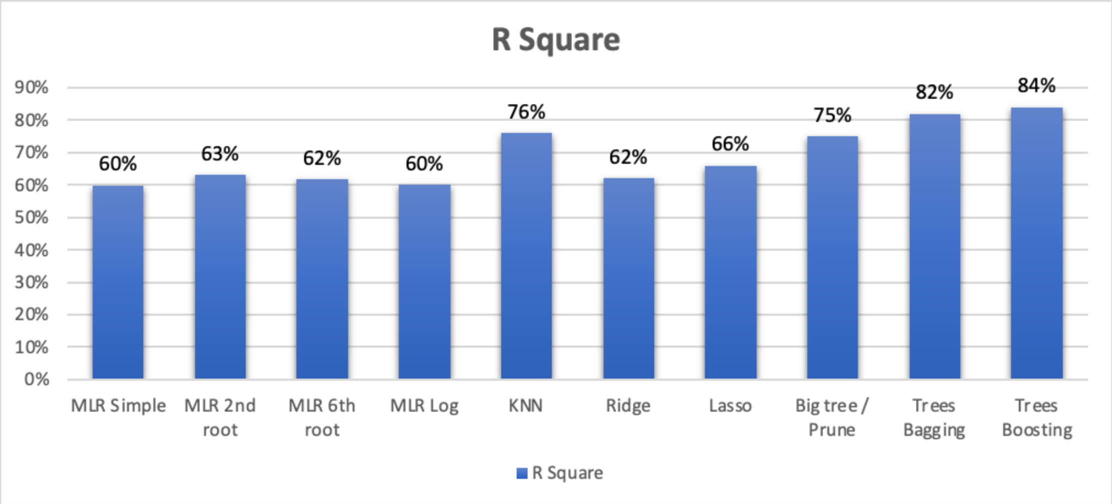


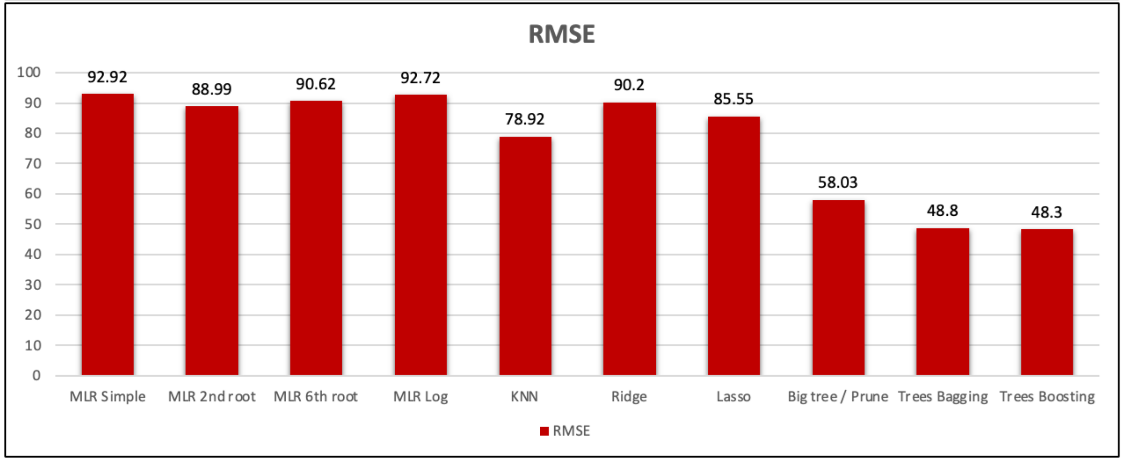
For Casual Users:

var	rel.inf
hr	36.4039009
temp	18.0786968
workingday	13.0298824
weekday	9.8553259
atemp	8.1963716
mnth	6.8000714
hum	4.4867479
windspeed	1.1222483
weathersit	0.8801604
season	0.7659165
holiday	0.3806780

- We used the same approach in predicting the registered users and found that boosting & bagging produced very similar results in the test set

Casual Users	RMSE (Test)	R Squared (Test)
Random Forest (Bagging)	17.01	87.8%
Boosting	17.13	88.7


R Square
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
60%
63%
62%
60%
76%
62%
66%
75%
82%
84%
MLR Simple
MLR 2nd root
MLR 6th root
MLR Log
KNN
Ridge
Lasso
Big tree / Prune
Trees Bagging
Trees Boosting
R Square


RMSE
100
90
80
70
60
50
40
30
20
10
0
92.92
88.99
90.62
92.72
78.92
90.2
85.55
58.03
48.8
48.3
MLR Simple
MLR 2nd root
MLR 6th root
MLR Log
KNN
Ridge
Lasso
Big tree / Prune
Trees Bagging
Trees Boosting
RMSE

Conclusion

- On running multiple linear regression, we found that the best fit was the square root model. So this implied that at least a square relationship existed.
- The coefficients in our Square Root MLR model seemed to agree with the initial Exploratory Data Analysis for both registered and casual users.
- As expected, a non parametric model like trees worked best for our dataset which had multiple nominal categorical variables (without an inherent order).
- So, the major factor influencing registered users was **Hour of the Day**, while the temporal/weather factors were not as important.
- On the other hand, while the **Hour of the Day** was still the most important variable influencing the casual riders, the **Temperature** too was of considerable importance.

Next Steps

- For casual users, we can try introducing a weekend pass model to take advantage of the positive correlation between no. of casual users and weekend days.
- Price surging/dynamic pricing can be based on weather conditions (temperature) for casual users. E.g. if it is a hot, sunny day and a weekend, our model says that there should be a higher number of casual users. It can be a good opportunity to raise prices.
- Using our model, Capital Bikeshare can perform demand and supply planning with reasonable accuracy.
- Finally, Capital Bikeshare should hire us for a model with even better accuracy!!