[2015_KAGGLE CHALLENGE CASE I – 자전거대여예측]

Bike Sharing Demand

Completed • Knowledge • 3,252 teams

Bike Sharing Demand

Wed 28 May 2014 – Fri 29 May 2015 (60 days ago)

최종등수 : 20등

2015.4 ~ 5

JASON, MIN

https://www.kaggle.com/c/bike-sharing-demand

2015_kaggle_Bike Sharing 문제

- **12개의 Data Fields**

datetime - hourly date + timestamp
season - 1 = spring, 2 = summer, 3 = fall, 4 = winter
holiday - whether the day is considered a holiday
workingday - whether the day is neither a weekend nor holiday
weather -
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 - temperature in Celsius
atemp - "feels like" temperature in Celsius
humidity - relative humidity
windspeed - wind speed
casual - number of non-registered user rentals initiated
registered - number of registered user rentals initiated
count - number of total rentals

Ensemble Method in Machine Learning & Optimizing

예측성능 강화를 위한 고민 1

- **데이터변환, 선택, 알고리즘 비교, 알고리즘 파라메터 튜닝 등**

Targeted value transform, Algorithm Param Tuning (RMSLE Check)

Count Transform : log(count+1)

문제 이해 및 데이터 분석 → R, PYTHON 언어, 툴 학습 (DT 이용예제 확인) → 알고리즘 별 성능확인 (With Param Optimized)

KAGGEL Submit Result : 0.5~0.4

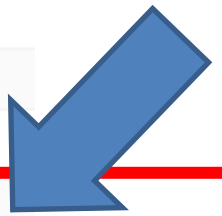
RandomForest : 0.381X
GradientBoosting : 0.368X
AdaBoost (With DT): 0.386X

→ 알고리즘 조합 성능확인(일부개선) → 특정일자(행사, 모래폭풍 등) 데이터 보정하기 (여기까지는 안함)

RandomForest (A%) & GradientBoosting(B%) : 0.365X

0.2 대로 진입하기 위해서는 필수적인 부분임 (Data Exploring)

#	Δ1w	Team Name * in the money	Score	Entries	Last Submission UTC (Best – Last Submission)
1	—	Team Oliver *	0.21545	23	Sun, 18 Jan 2015 00:03:26
2	—	Alliance	0.24976	5	Tue, 30 Sep 2014 11:20:46
3	—	A_Power	0.28820	14	Tue, 23 Sep 2014 00:53:13 (-23.7h)
4	↑733	Vecihi	0.29401	22	Sun, 24 May 2015 15:09:56
5	↓1	Pourquoipas	0.29585	21	Fri, 15 Aug 2014 16:01:06 (-2.3h)
6	↓1	3+me	0.31438	13	Mon, 18 May 2015 17:21:02
7	↓1	Starboy	0.31967	11	Sat, 23 Aug 2014 14:22:44
8	↓1	Bolaka Mukherjee	0.33757	26	Wed, 27 May 2015 11:12:31
9	↑7	Logical Guess	0.34821	25	Fri, 29 May 2015 14:35:32
10	↑2978	Louis Martin	0.34835	18	Fri, 29 May 2015 11:32:22
11	↓3	张 李	0.34928	14	Thu, 01 Jan 2015 04:59:53
12	↓3	rediculous	0.34928	5	Sat, 03 Jan 2015 12:46:15 (-0.5h)
13	↓3	just_did	0.34930	17	Sat, 03 Jan 2015 08:51:41 (-28.5h)
14	↓3	Greg	0.35570	47	Fri, 21 Nov 2014 14:23:12
15	↓3	allmi	0.35596	9	Sat, 14 Mar 2015 16:37:26
16	↓3	Gopal Joshi	0.35705	5	Tue, 25 Nov 2014 07:05:17
17	↓3	Steven Lee	0.35784	23	Fri, 23 Jan 2015 10:11:23 (-11.9d)
18	↑29	tsing	0.35868	4	Fri, 29 May 2015 17:36:06
19	↑3	ngon	0.35902	5	Fri, 29 May 2015 03:04:57
20	↓5	jasonM	0.36064	26	Fri, 22 May 2015 07:04:23
21	↓4	ZTRON	0.36084	16	Thu, 26 Mar 2015 16:47:54 (-0.1h)
22	↓4	junglee	0.36249	22	Wed, 17 Sep 2014 04:39:28



NN with R Script START
2015.5.26

(기본시도 1) NeuralNet 알고리즘 적용가능성 확인

- count를 단순히 1/1000로 조정 후 복원, 피처 단순화 적용

 # 코드중 일부

```
ibrary(neuralnet)

...

fit <- neuralnet(formula ,data=trainmat, hidden=c(7,8,9,8,7), threshold=.04, stepmax=1e+06,
learningrate=.001, algorithm='rprop+',lifesign='full',likelihood=T)

...
...
predict <- compute(fit,testmat)
```

 # (콘솔) 출력값

```
hidden: 7, 8, 9, 8, 7 thresh: 0.04 rep: 1/1
steps: 1000 min thresh: 0.1159639204
       2000 min thresh: 0.1074123852
       3000 min thresh: 0.06619459617
       4000 min thresh: 0.05100477041
       5000 min thresh: 0.04617860885
       6000 min thresh: 0.04617860885
       6338 error: 15.1958 aic: 1140.3916 bic: 5189.24582 time: 11.41 mins
```

 # (KAGGLE) SUBMISSION RESULT

https://www.kaggle.com/c/bike-sharing-demand/forums/t/11022/a-simple-neural-network-model-for-bike-sharing/76002

(개선 1) NeuralNet 알고리즘 적용 및 개선

- count를 log로 조정 후 exp로 복원 (피처 단순화 적용)

 # 코드개선

```
#count <- train$count/1000
count <- log(train$count+1)  # COUNT SCALE : LOG => EXP
```

… … 연산 … …

```
#predict<- predict*1000
predict<- exp(predict)-1 # COUNT RESCALE : LOG => EXP
```

 # (KAGGLE) SUBMISSION RESULT – NOT GOOD

```
hidden: 7, 8, 9, 8, 7    thresh: 0.04    rep: 1/1    steps:    1000 min thresh: 0.2172270469
                                                                2000 min thresh: 0.1227302811
                                                                3000 min thresh: 0.07834961153
                                                                4000 min thresh: 0.05676199977
                                                                5000 min thresh: 0.05676199977
                                                                6000 min thresh: 0.05676199977
                                                                7000 min thresh: 0.05676199977
                                                                8000 min thresh: 0.05676199977
                                                                9000 min thresh: 0.05676199977
                                                               10000 min thresh: 0.05676199977
```

Your submission scored **3.29720**, which is not an improvement of your best score. Keep trying!

관련 자료 조사

Forecasting Utilization in City Bike-Share Program

Christina Lee, David Wang, Adeline Wong

2. Neural network

The Poisson regression model assumes that the variables are more-or-less independent. Another way to view the problem is throw at the problem a black-box machine learning model that is able to capture complexity, such as interacting variables. Neural networks are good at recognizing these hidden patterns, hence our choice to train a neural network. After trying a variety of parameters, the final neural network consisted of four hidden layers (9 nodes in the first hidden layer, 3 in the second hidden layer, 3 in the third hidden layer, and 23 in the fourth hidden layer), trained using the `neuralnet` R package and learning rate of 0.001 with resilient backpropagation with weight backtracking.

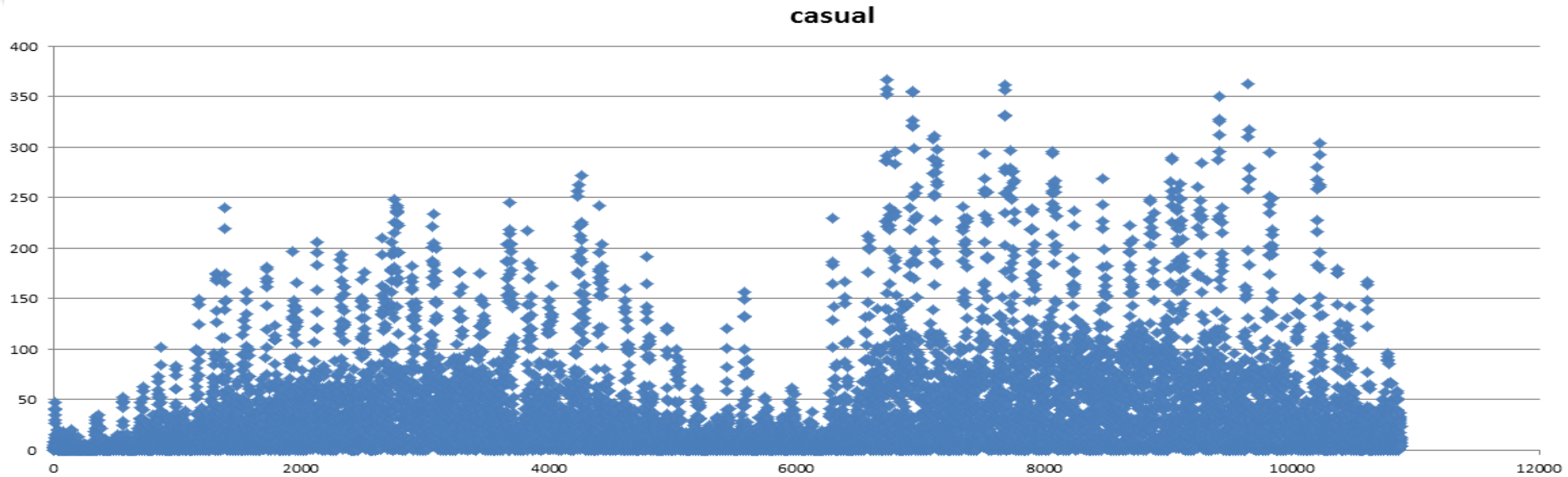
Model	Training error	Test error
Neural Network	0.451017	0.48992
Poisson Regression	0.676233	0.69899
Markov Model	0.73738	1.70463
Mean Value Benchmark*	1.569198	1.58456

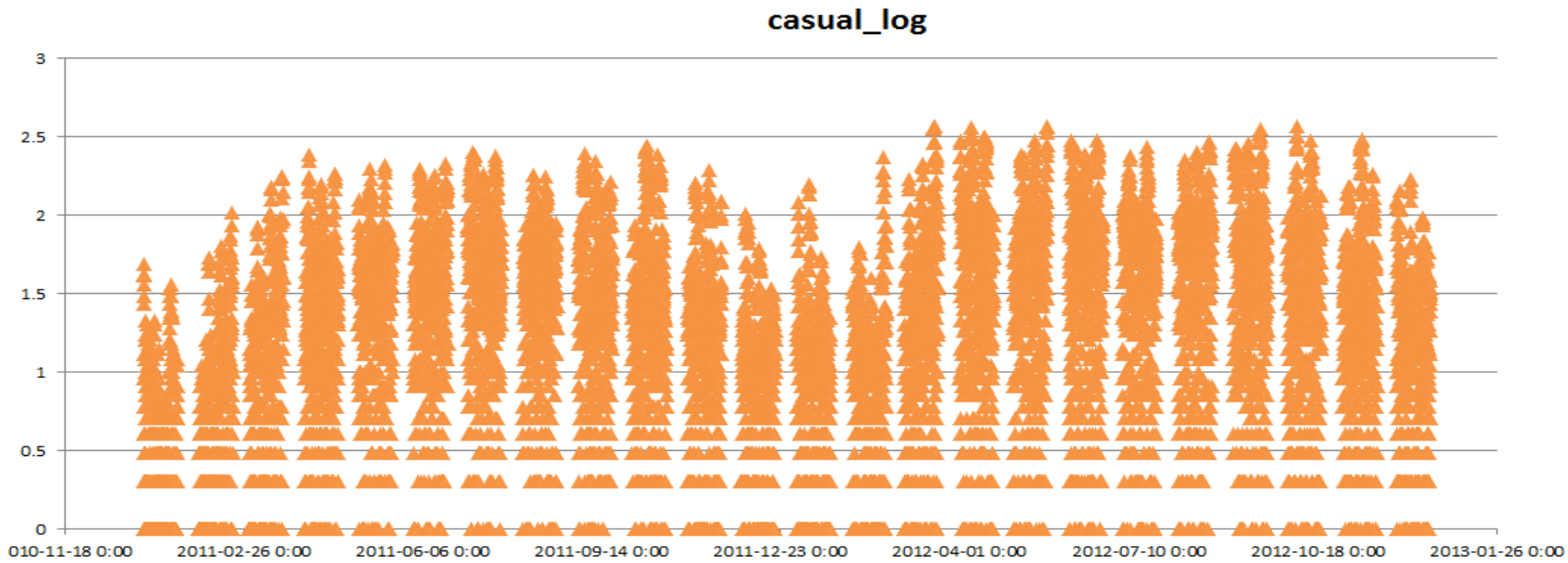
*For reference purposes, predictions take the mean value of the training data.

기타 시도들

Data Transform 1 log(count+1) : 자연로그

10 => 2.3, 100=>4.6, 300=>5.7





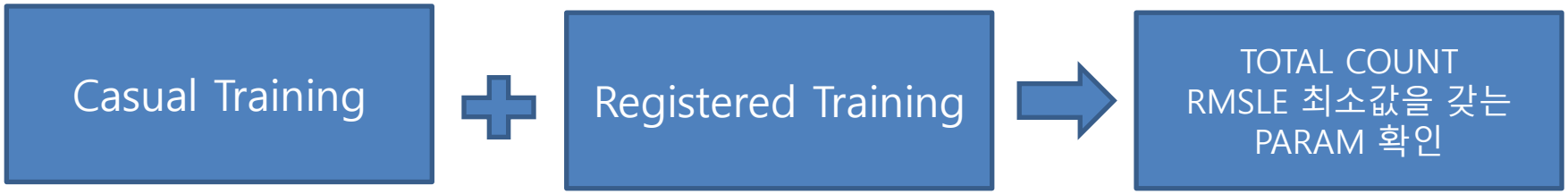
Ensemble Method in Machine Learning & Optimizing

예측성능 강화를 위한 고민 2

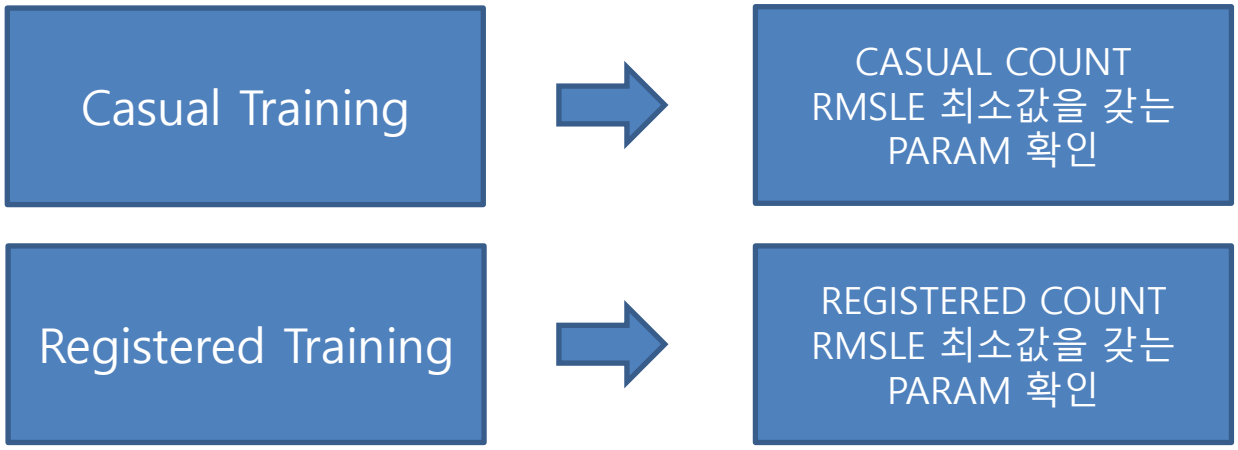
- **파람 튜닝 2**

Casual, Registered Training 을 위한 Param 최적값 각각 확인

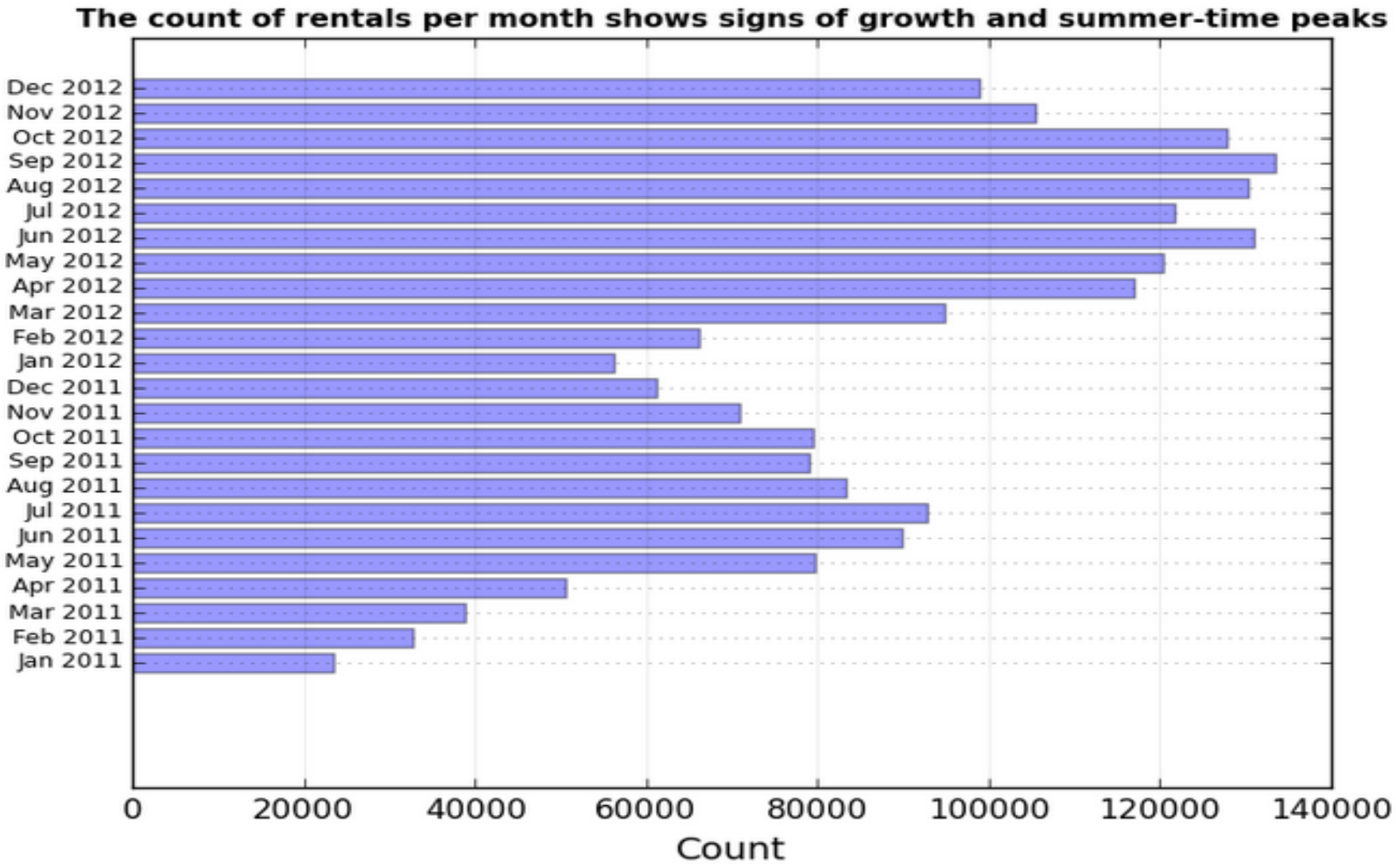
기존



개선

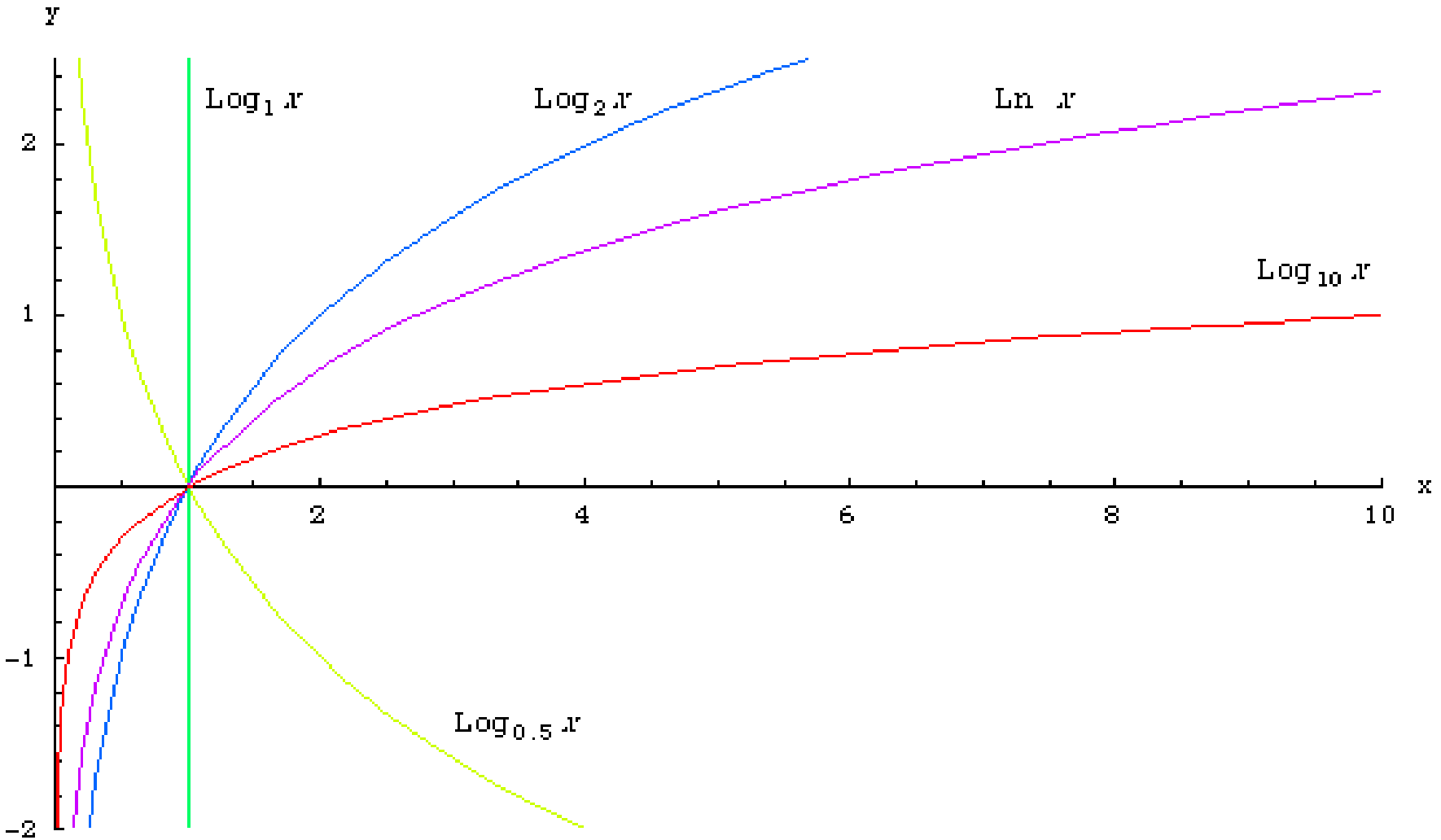


Data Exploring



https://www.kaggle.com/users/346867/vijaysathish/bike-sharing-demand/average-daily-rental-counts-by-month

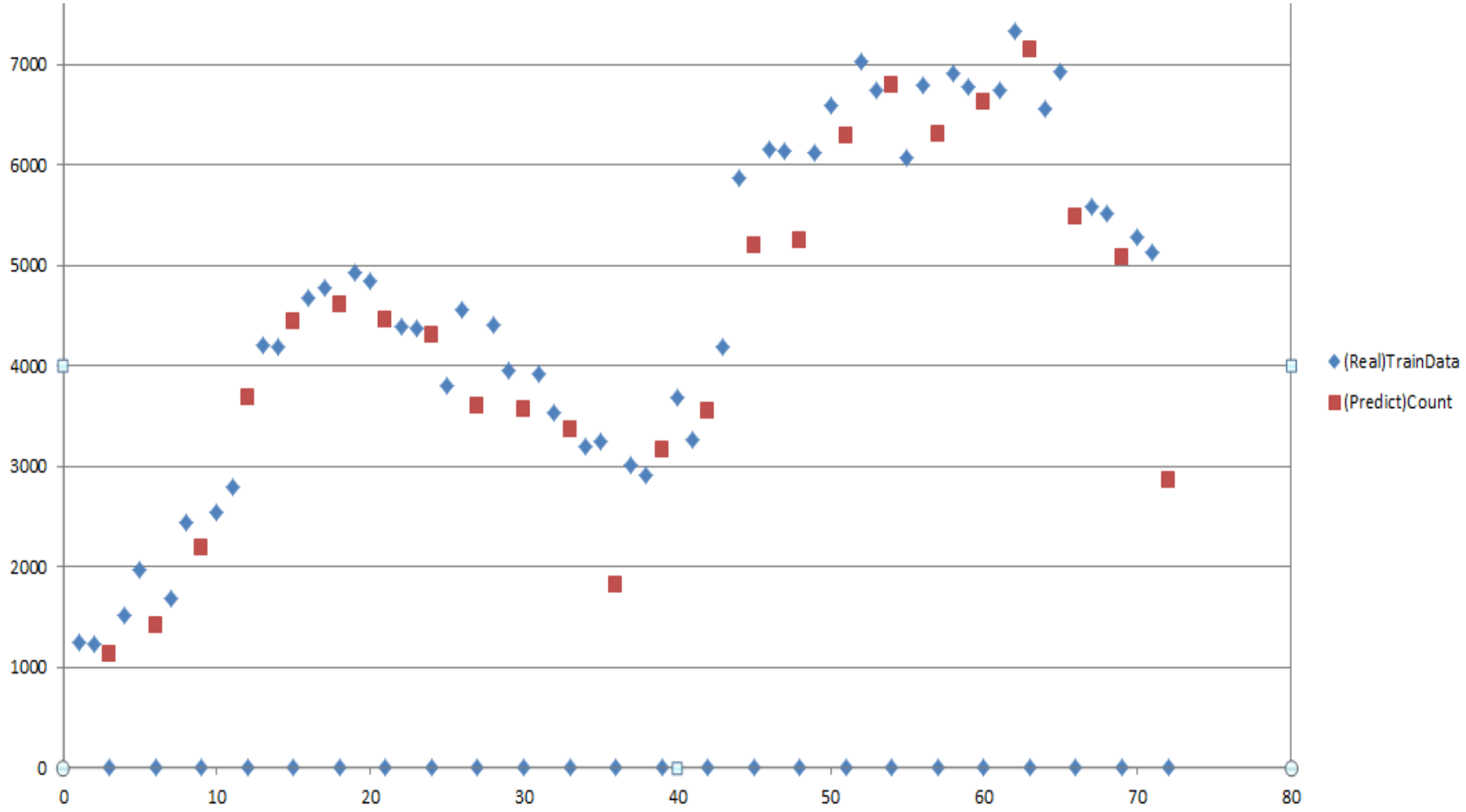
Data Exploring



http://webmath.amherst.edu/qcenter/logarithms/index.html

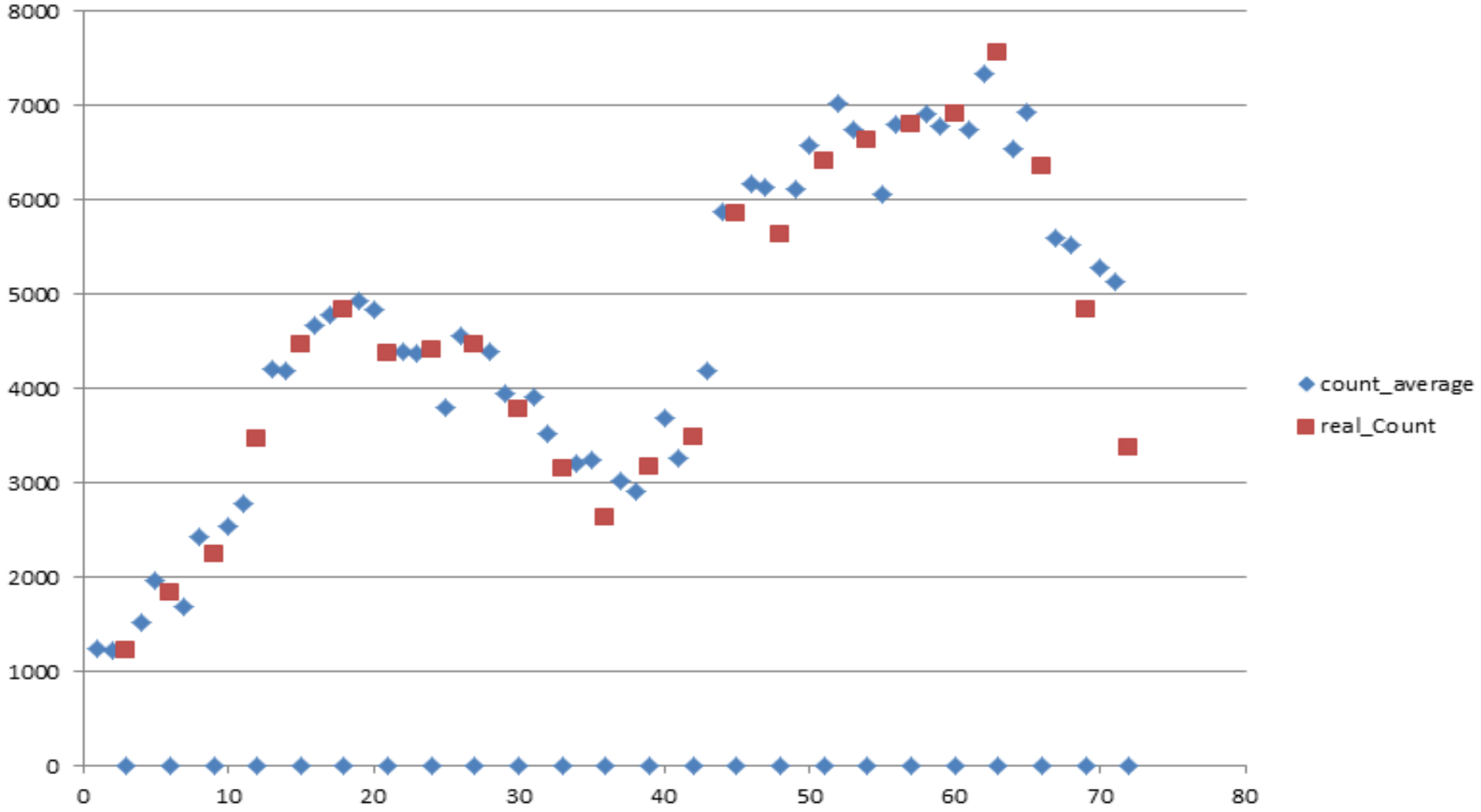
실제 데이터 (1~10, 11~19일) 와 예측데이터(20~말일)의 차이 – Mean

TODO 1 : 데이터(Feature) 추가(가상으로, 20~말일) – 의미가 있을지.. &
가정 1 : 결과(Target-Count)에 대한 LR보정 필요



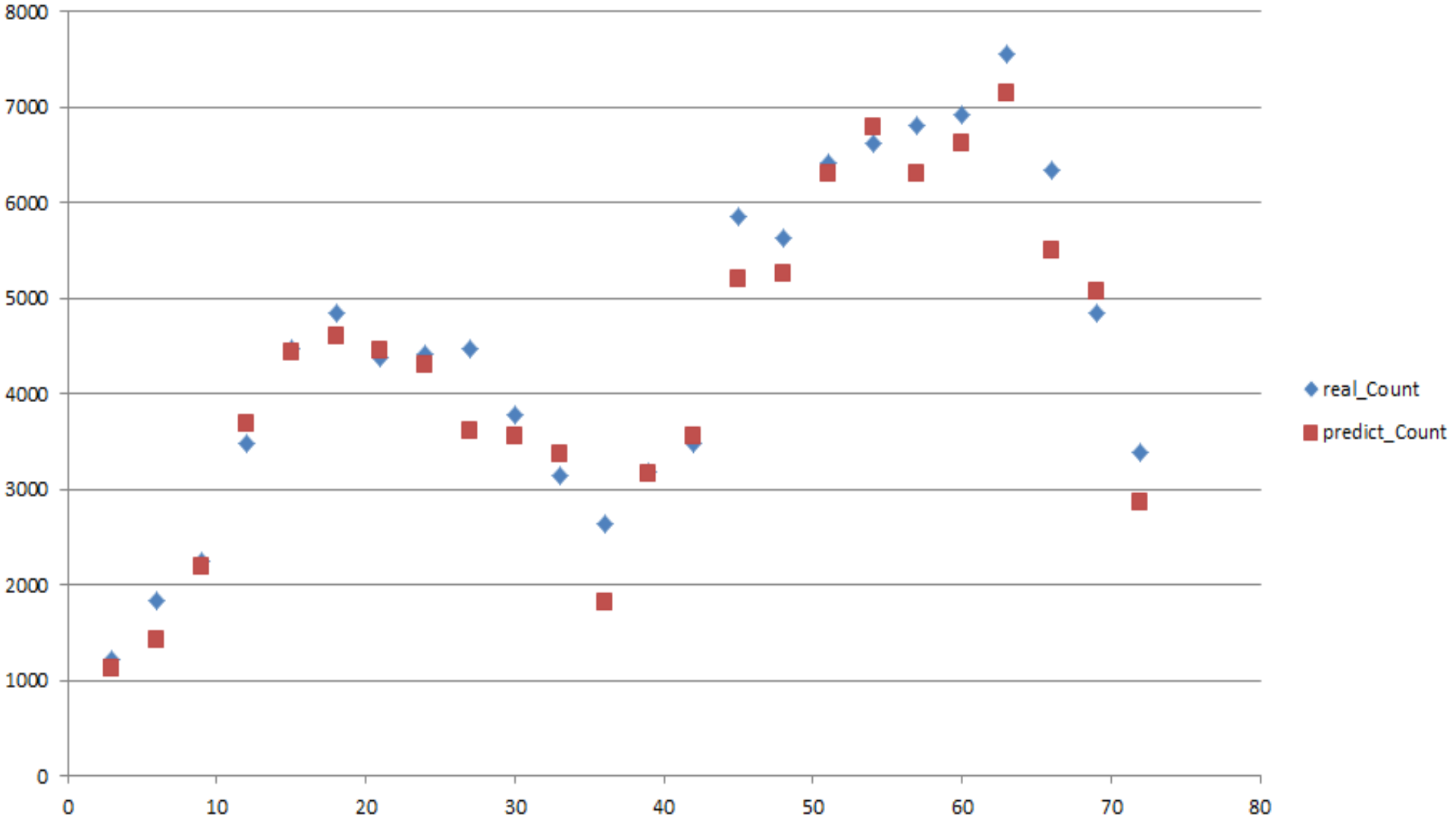
실제 이용데이터 확인 – 가정 확인 (이건 단순 검증용 자료임) – 5.20

실제 이용수요(바이크 대여 카운드)가 완만한 선형으로 나타날지 확인
(월을 상,중,하순으로 나누어 평균값 확인)



실제 이용데이터 확인 – 가정 확인 (이건 단순 검증용 자료임) – 5.20

예측값과 실제값의 차이, 평균 차이 카운트 : 약 **233** 정도임

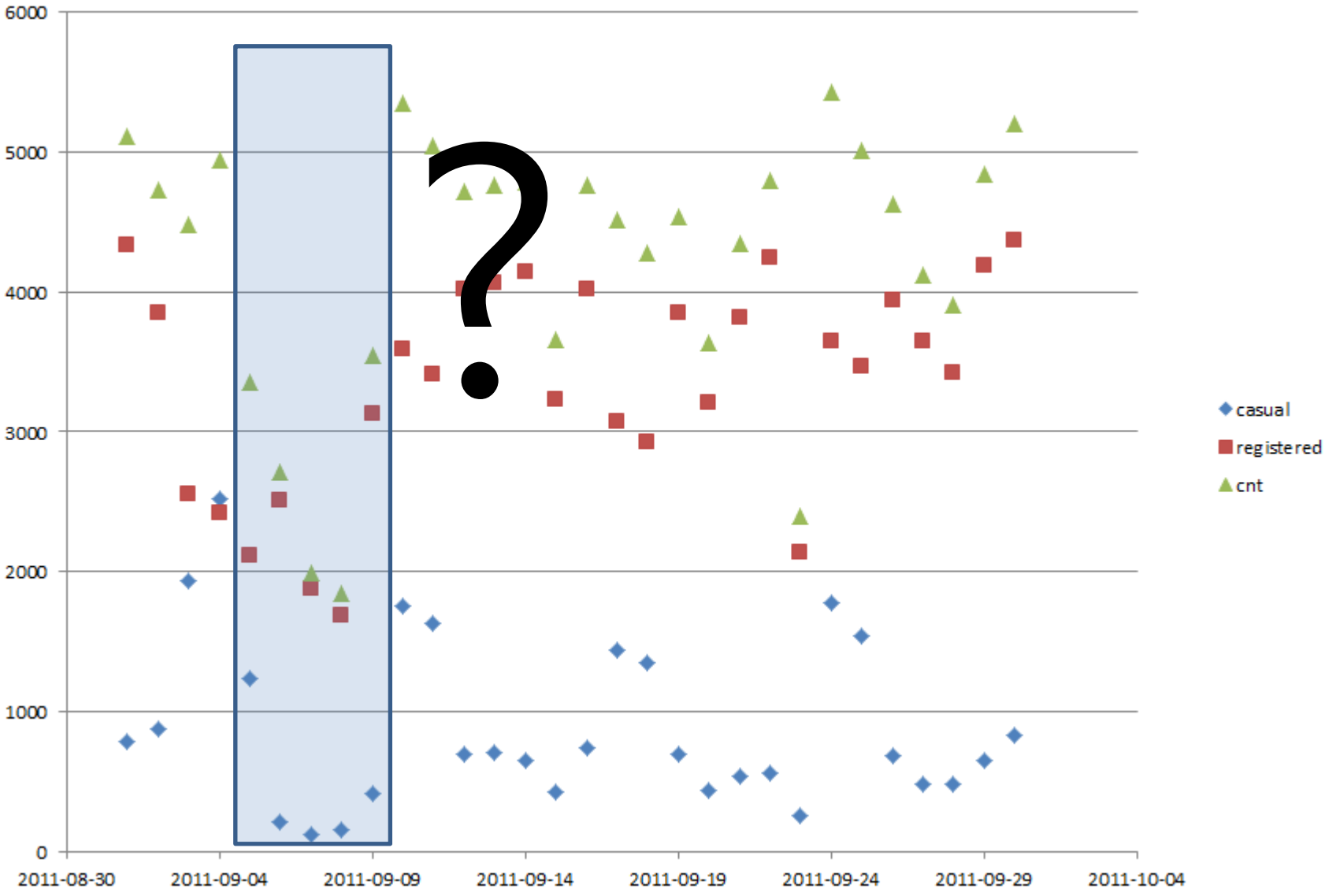


보정값, 실제값, 예측값 결과 비교

1. 보정기준 : 차이가 많이 나는경우 (보정후보값과 예측값(%)의 90% 값)
2. 12월의 경우 방향성을 보고 예측값 차이%의 60%정도 보정)

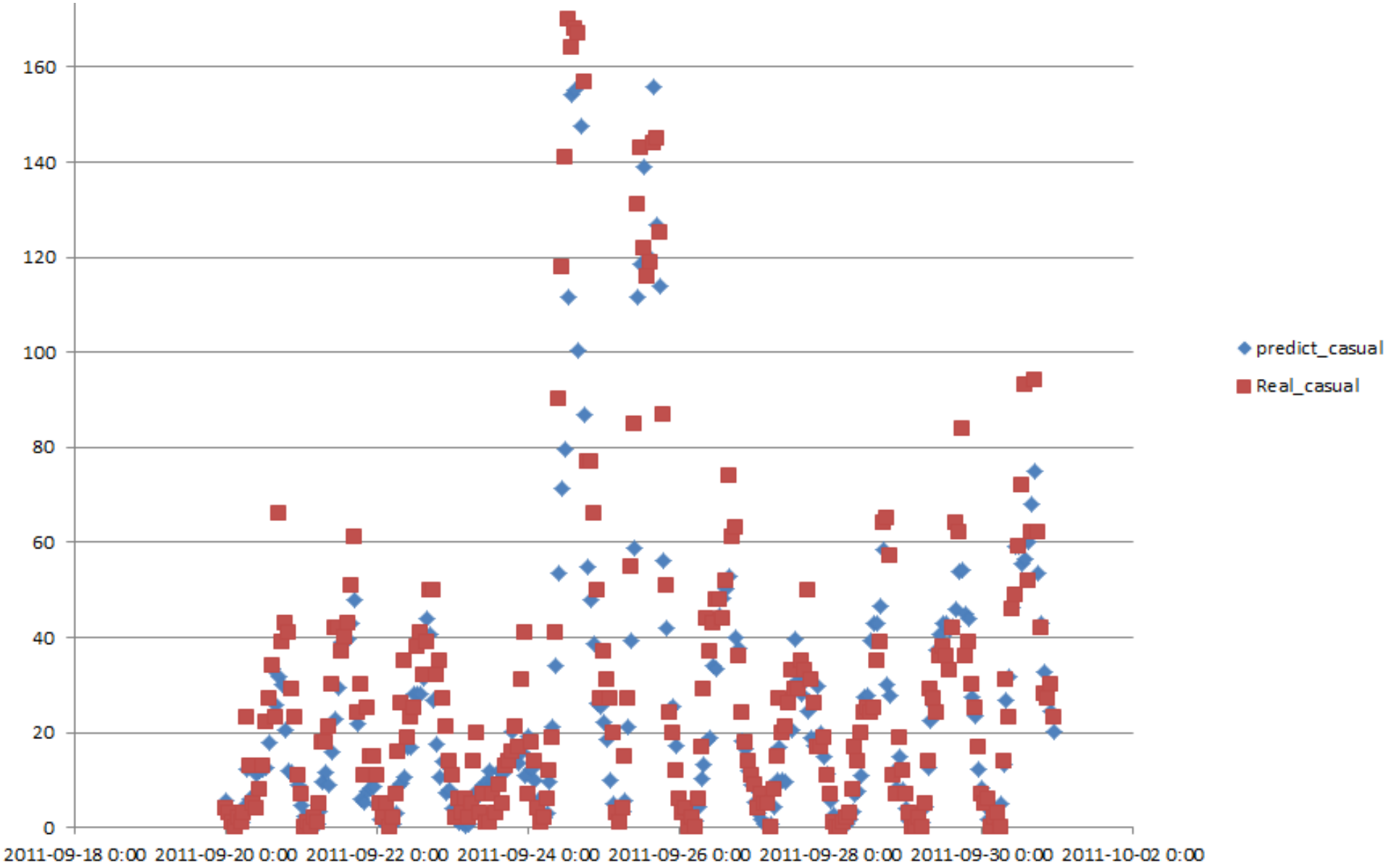
보정값, 실제값, 예측값 결과 비교

2011.9월 데이터 분석 REAL Casual, Registered



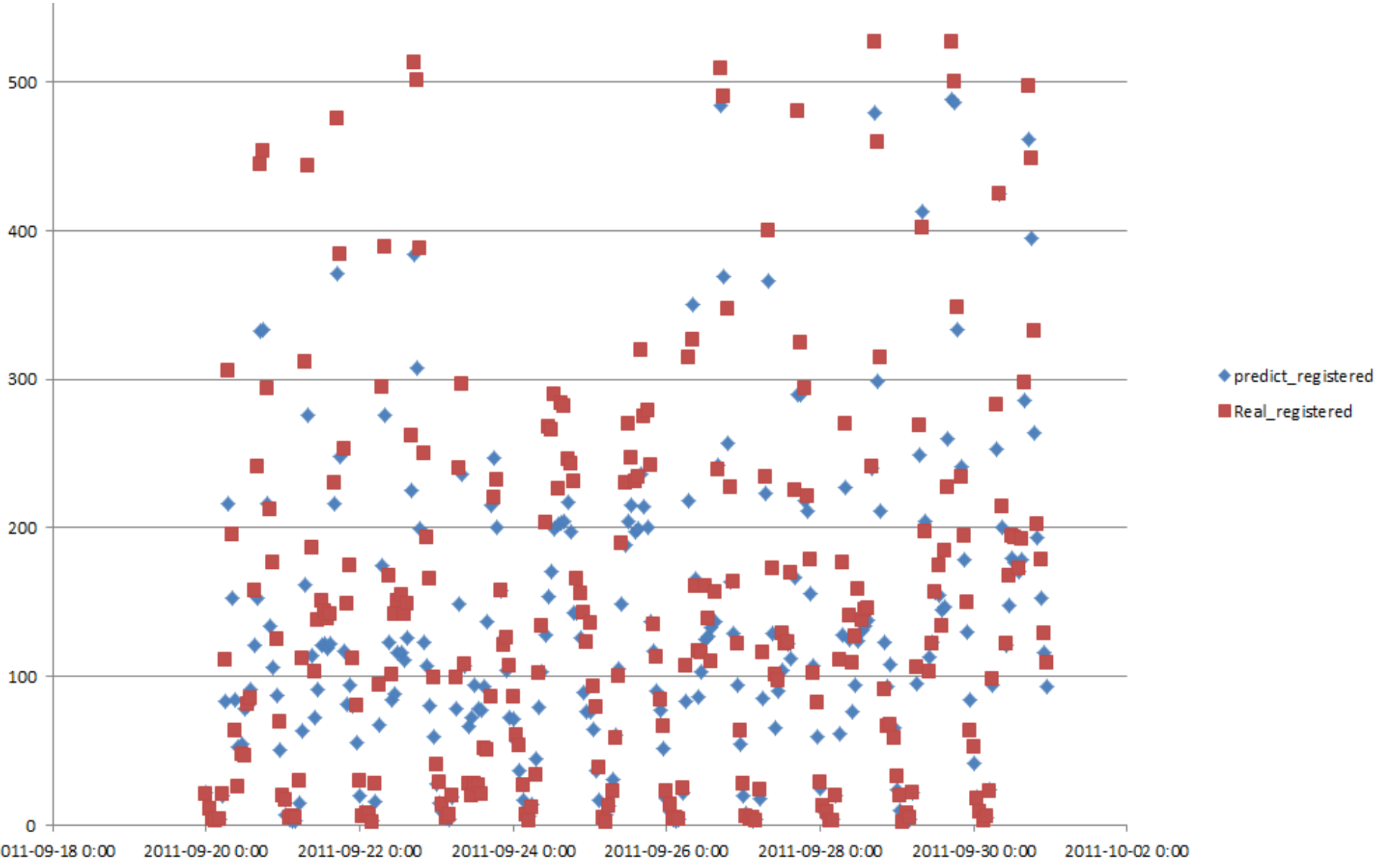
보정값, 실제값, 예측값 결과 비교

2011.9월 데이터 분석 REAL/PREDICT Casual, Registered



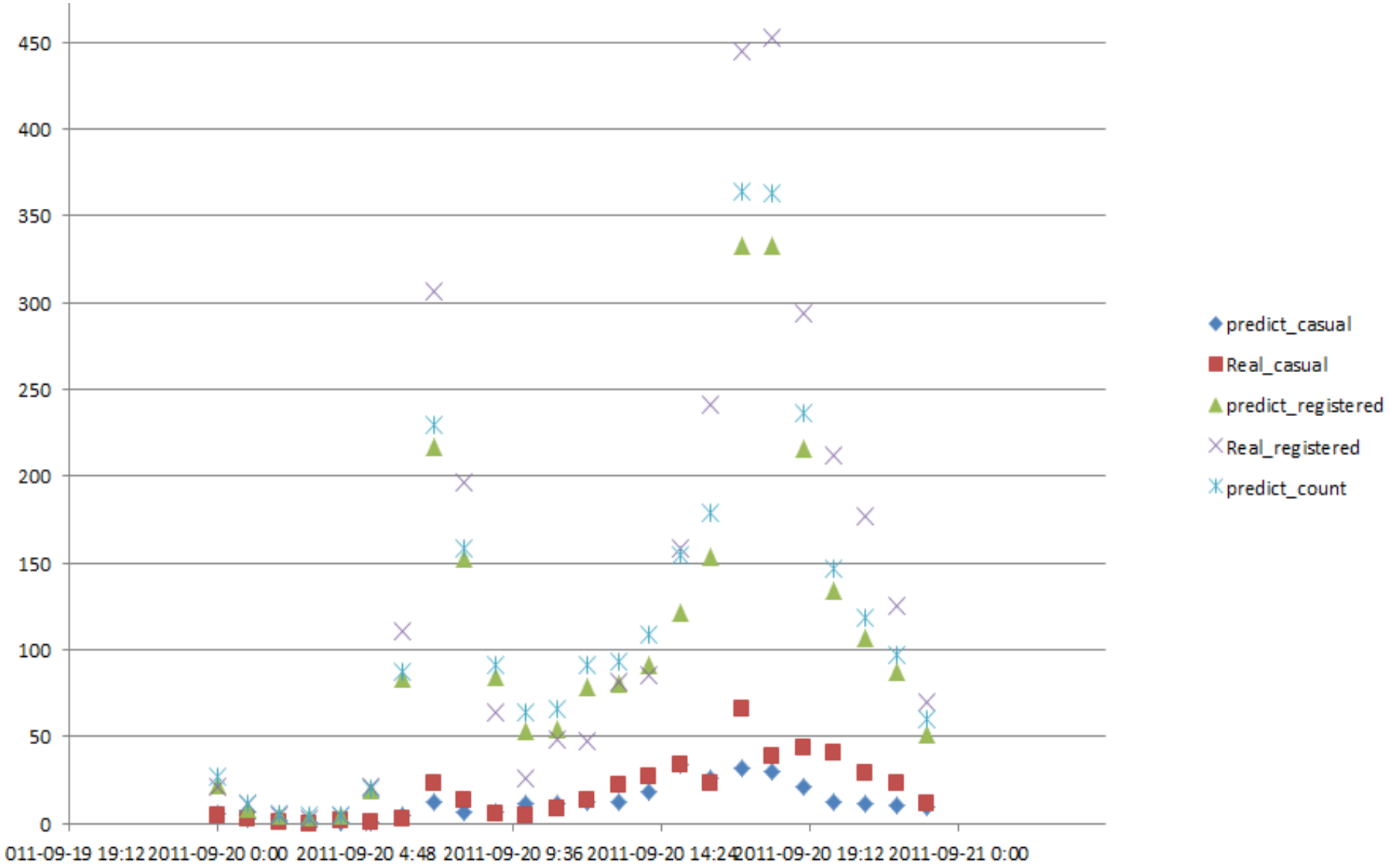
보정값, 실제값, 예측값 결과 비교

2011.9월 데이터 분석 REAL/PREDICT Registered

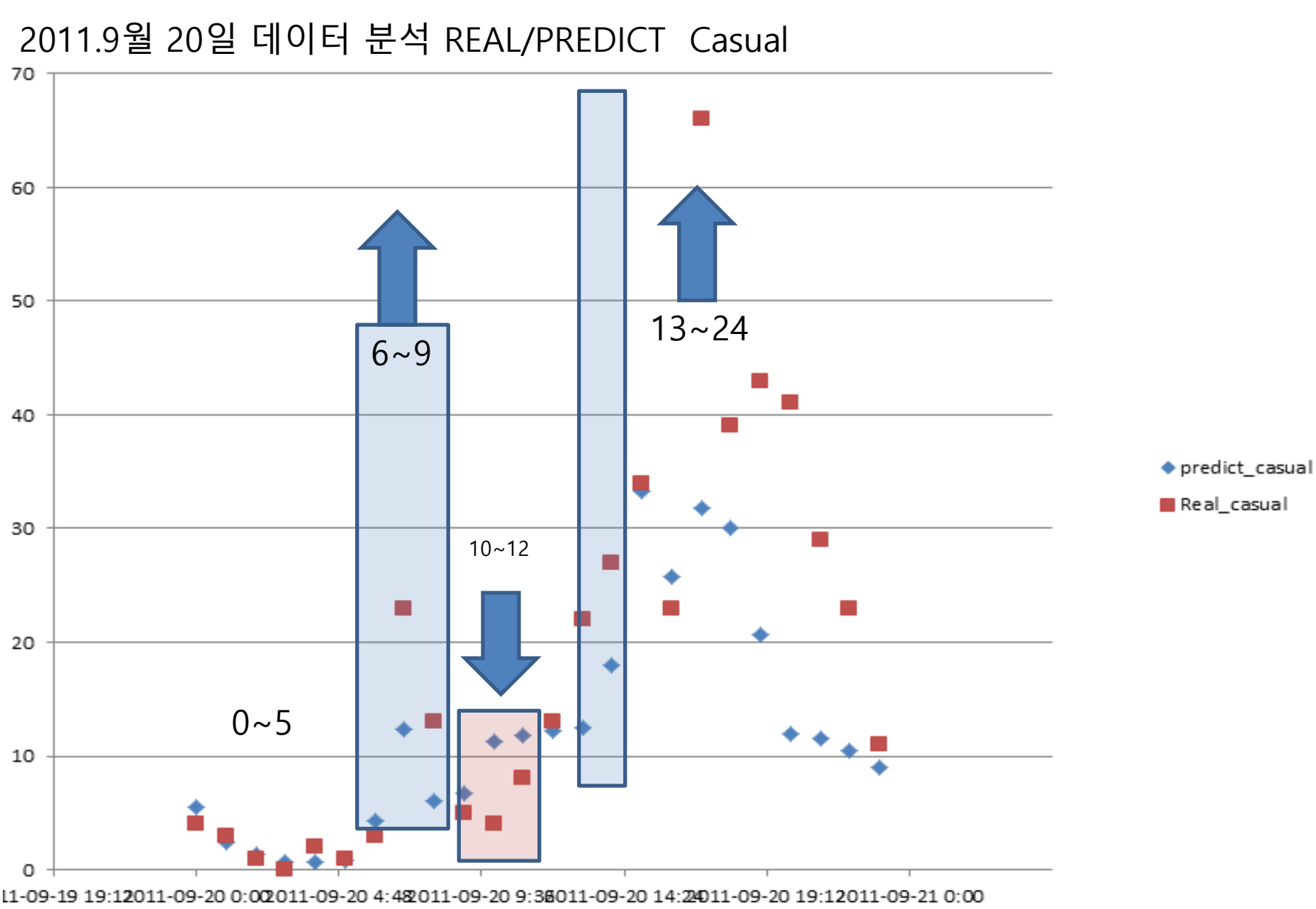


보정값, 실제값, 예측값 결과 비교

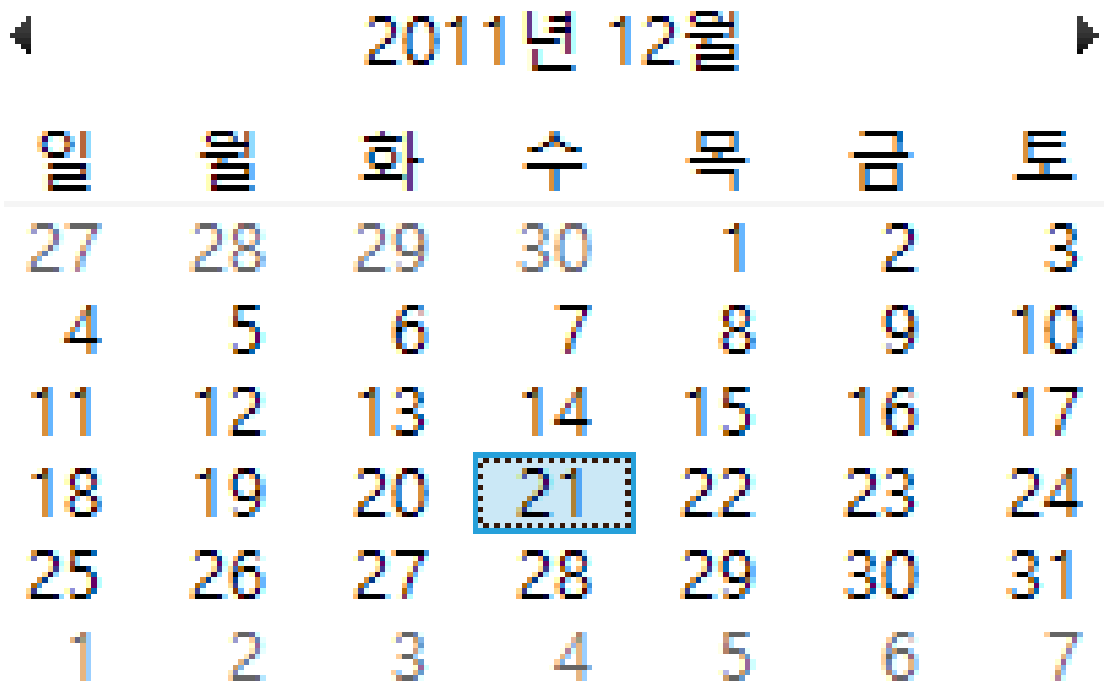
2011.9월 20일 데이터 분석 REAL/PREDICT Casual, Registered



보정값, 실제값, 예측값 결과 비교

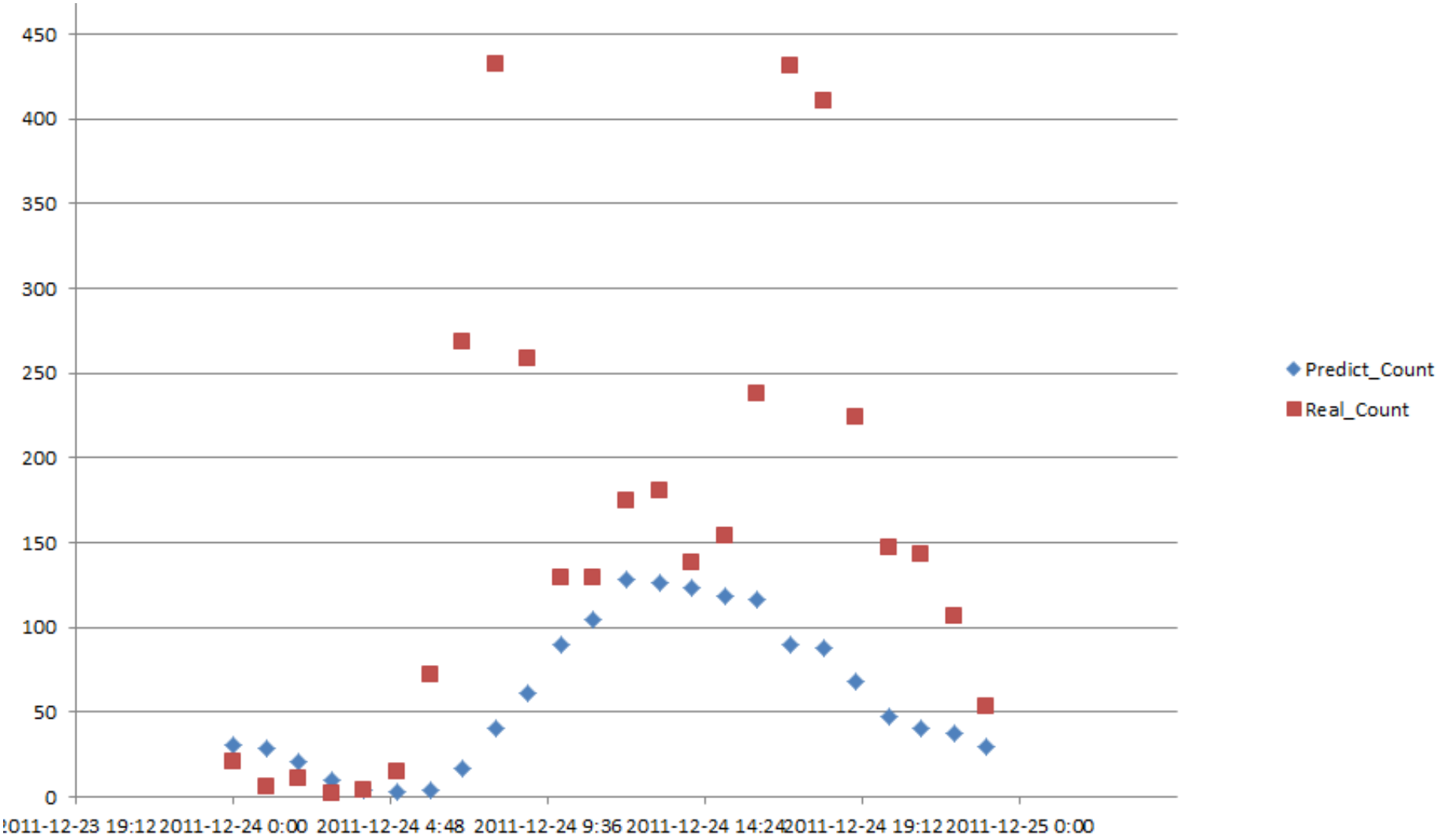


보정값, 실제값, 예측값 결과 비교



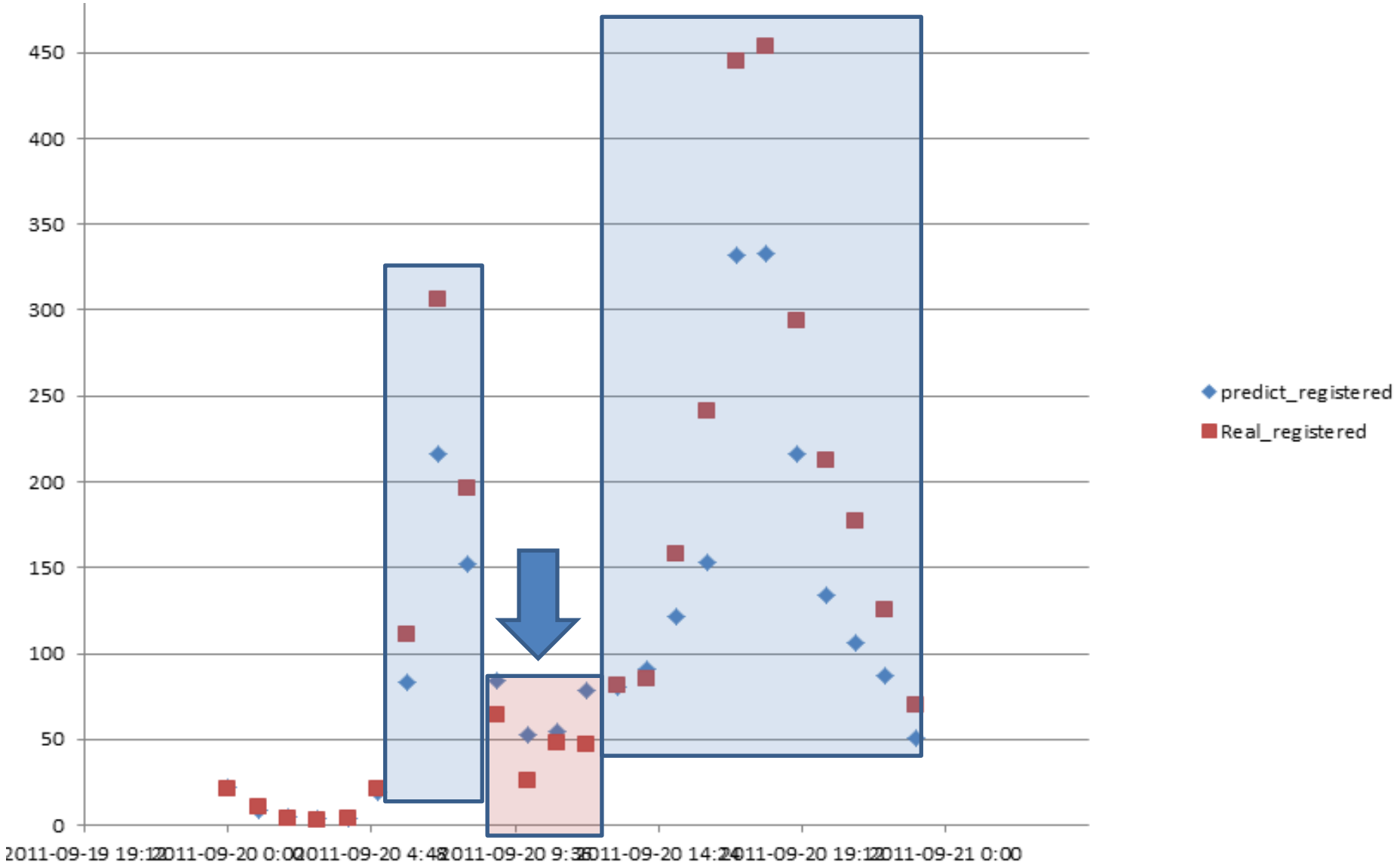
실제값, 예측값 결과 비교

2011.12월 20일(토) 데이터 분석 REAL/PREDICT Count



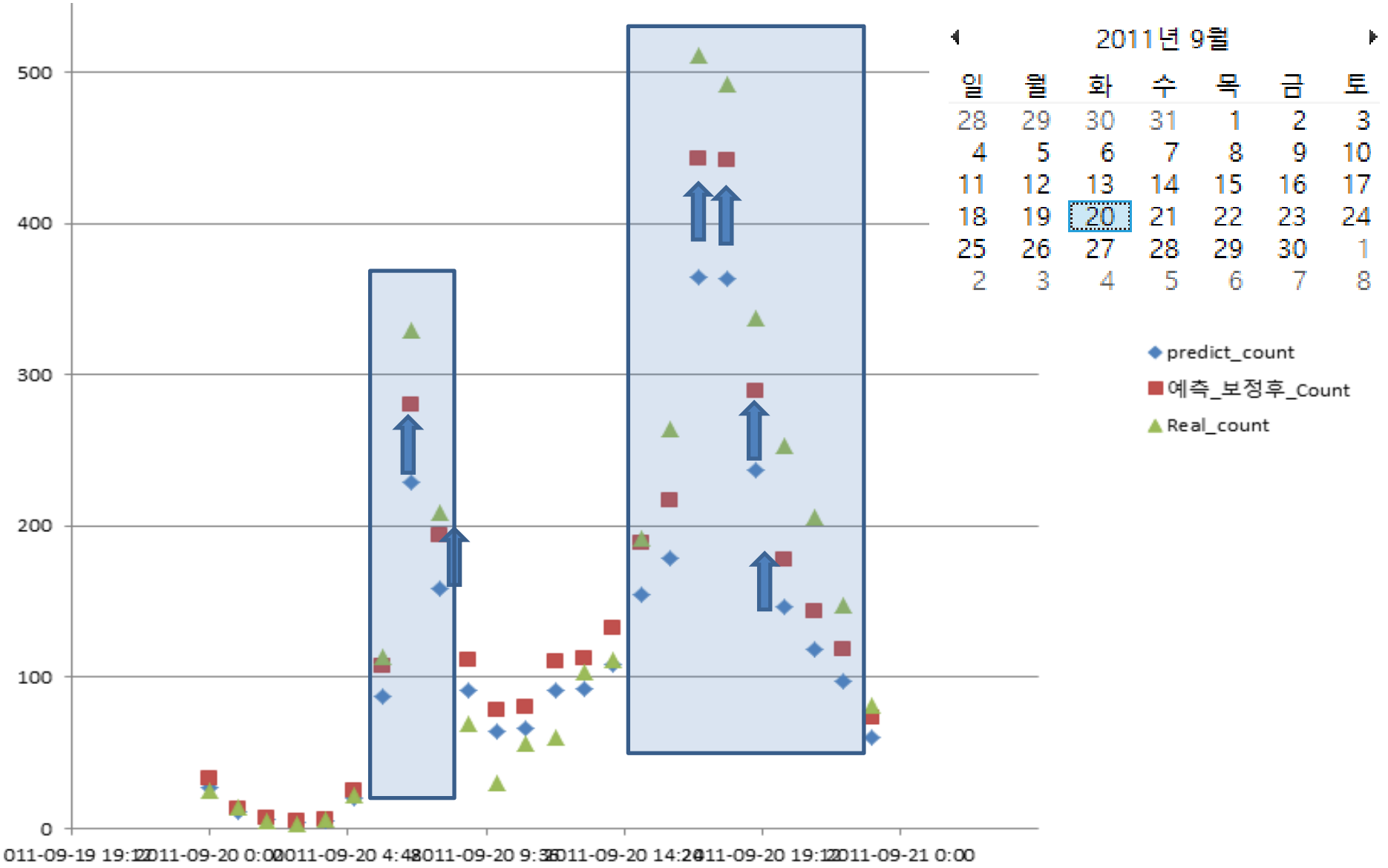
보정값, 실제값, 예측값 결과 비교

2011.9월 20일 데이터 분석 REAL/PREDICT Registered



보정값, 실제값, 예측값 결과 비교

2011.9월 20일(화) 데이터 분석 REAL/PREDICT Total Count

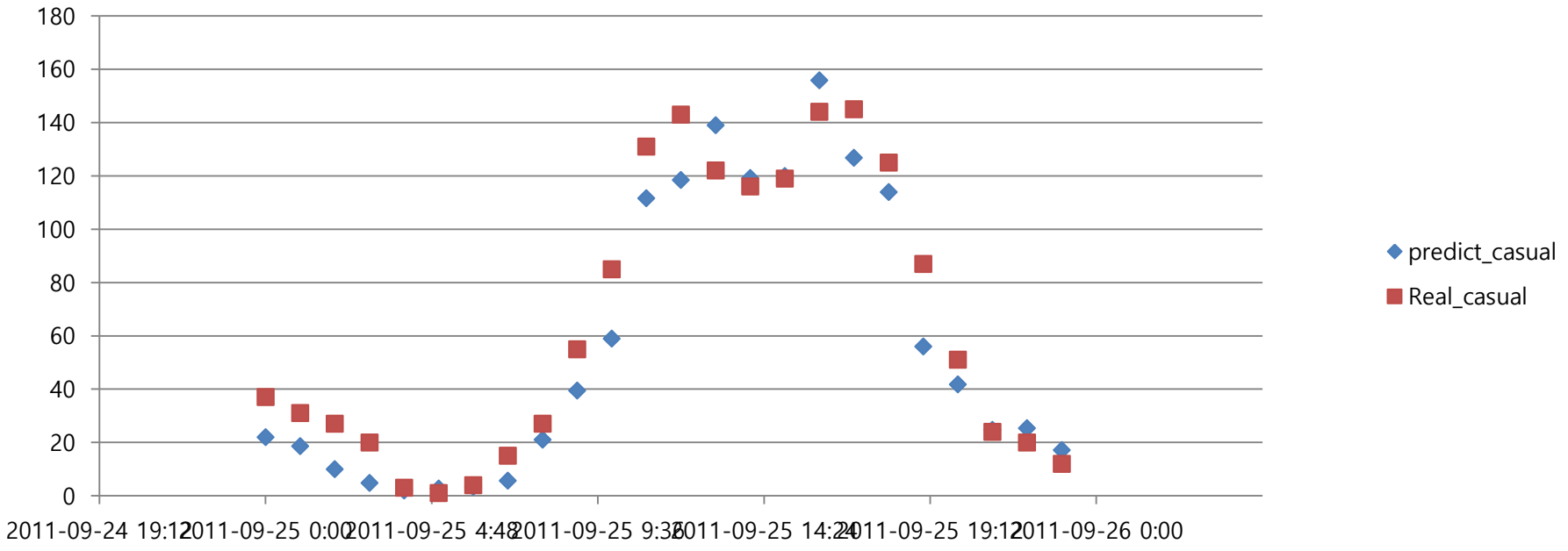


보정값, 실제값, 예측값 결과 비교

2011.9월 25일(일) 데이터 분석 REAL/PREDICT Casual

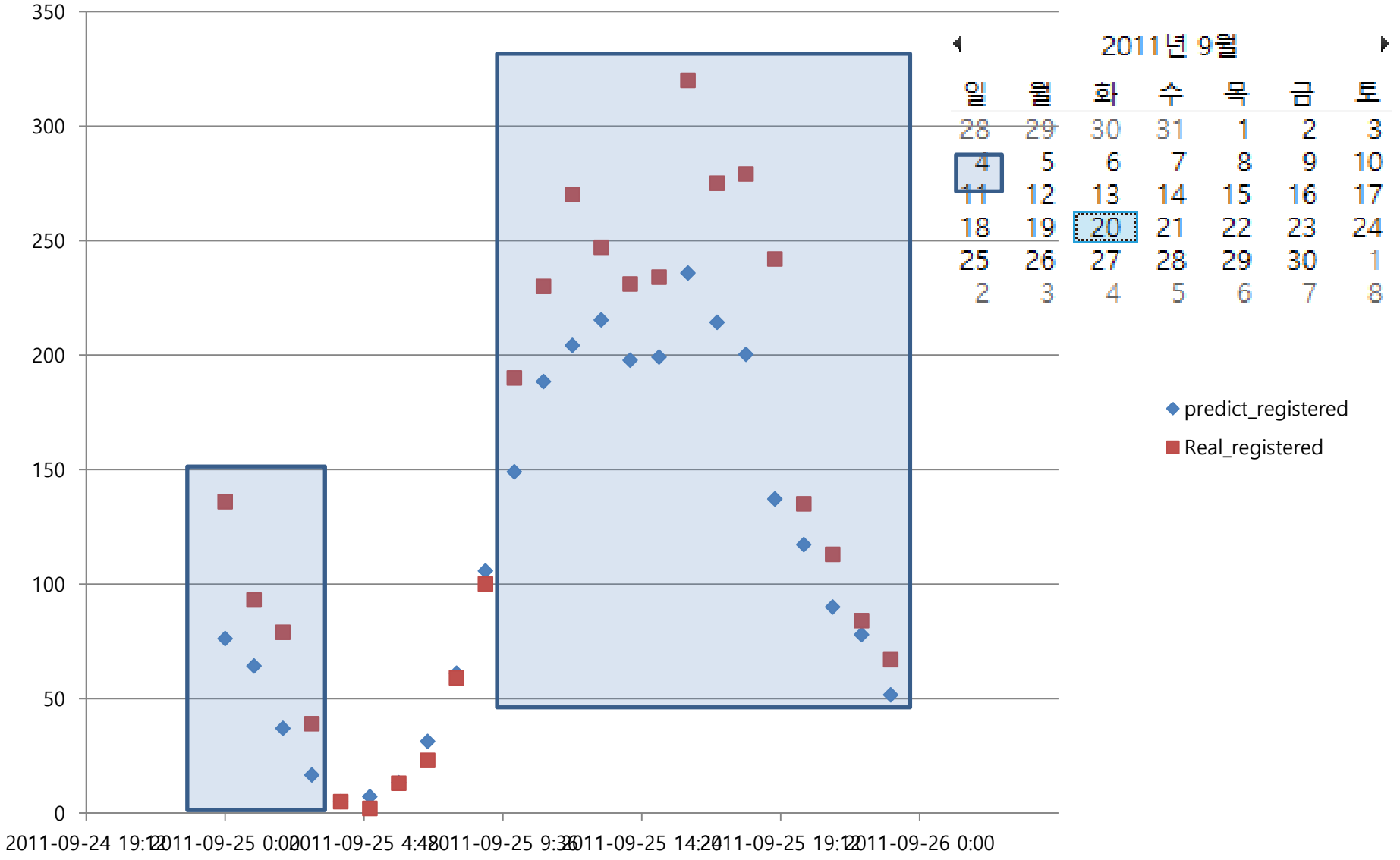
2011년 9월

일	월	화	수	목	금	토
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8



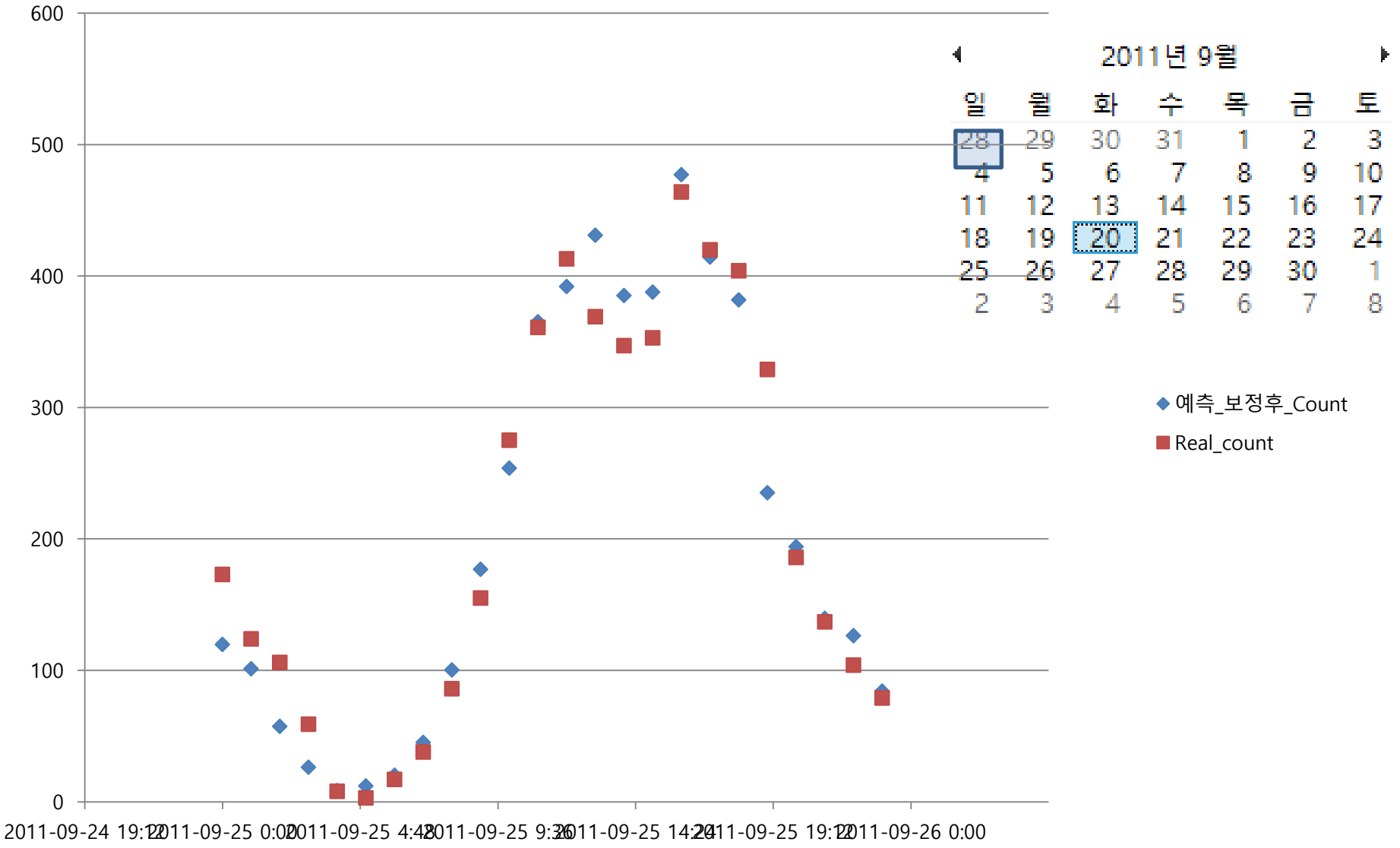
보정값, 실제값, 예측값 결과 비교

2011.9월 25일(일) 데이터 분석 REAL/PREDICT Registered



보정값, 실제값, 예측값 결과 비교

2011.9월 25일(일) 데이터 분석 REAL/PREDICT Total Count



2011년 9월

일	월	화	수	목	금	토
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

알고리즘 별 비교 연구자료 확인

[Dietteric (2000) Study] : AdaBoost often gives the best results
(no Noise in Data)

Table 1. All pairwise combinations of the four ensemble methods. Each cell contains the number of wins, losses, and ties between the algorithm in that row and the algorithm in that column.

	C4.5	AdaBoost C4.5	Bagged C4.5
Random C4.5	14 – 0 – 19	1 – 7 – 25	6 – 3 – 24
Bagged C4.5	11 – 0 – 22	1 – 8 – 24	
AdaBoost C4.5	17 – 0 – 16		

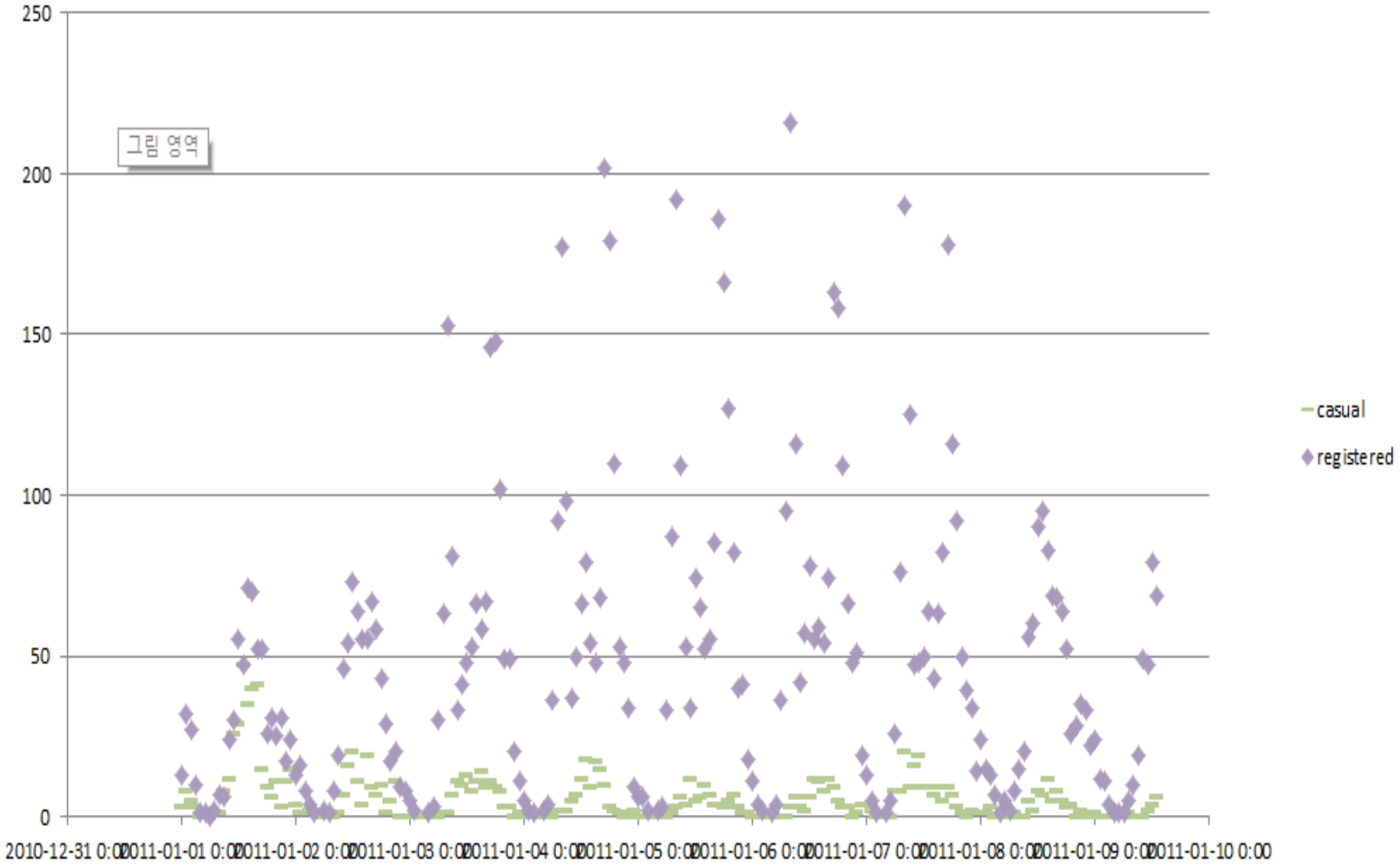
[Dietteric (2000) Study] : AdaBoost overfits
(데이터에 노이즈(20%)가 있는 경우)

Table 2. All pairwise combinations of C4.5, AdaBoosted C4.5, Bagged C4.5, and Randomized C4.5 on 9 domains with 20% synthetic class label noise. Each cell contains the number of wins, losses, and ties between the algorithm in that row and the algorithm in that column.

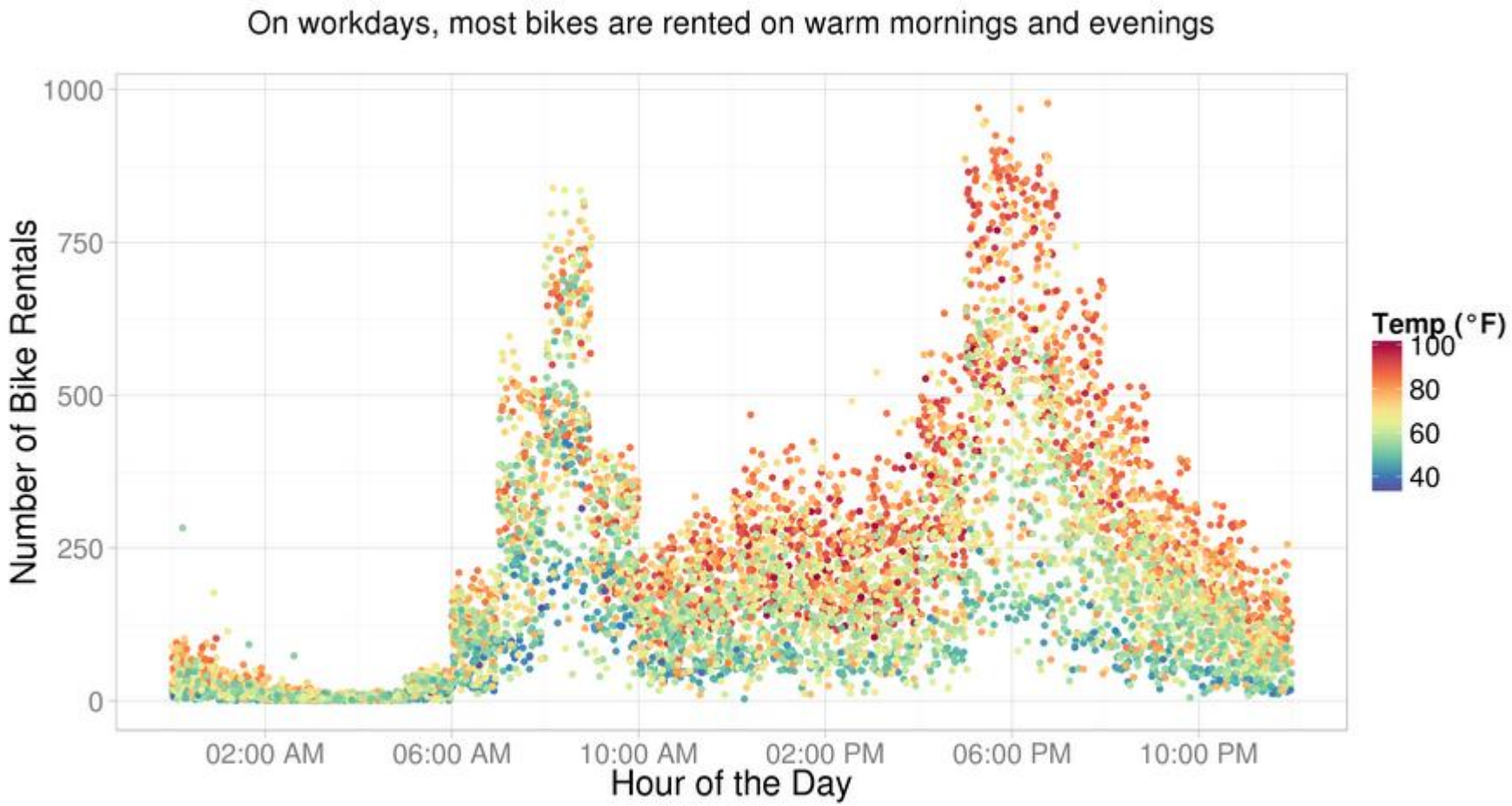
	C4.5	AdaBoost C4.5	Bagged C4.5
Random C4.5	5 – 2 – 2	5 – 0 – 4	0 – 2 – 7
Bagged C4.5	7 – 0 – 2	6 – 0 – 3	
AdaBoost C4.5	3 – 6 – 0		

출처 : Ensemble Methods in Machine Learning
Thomas G Dietteric h

Data Exploring

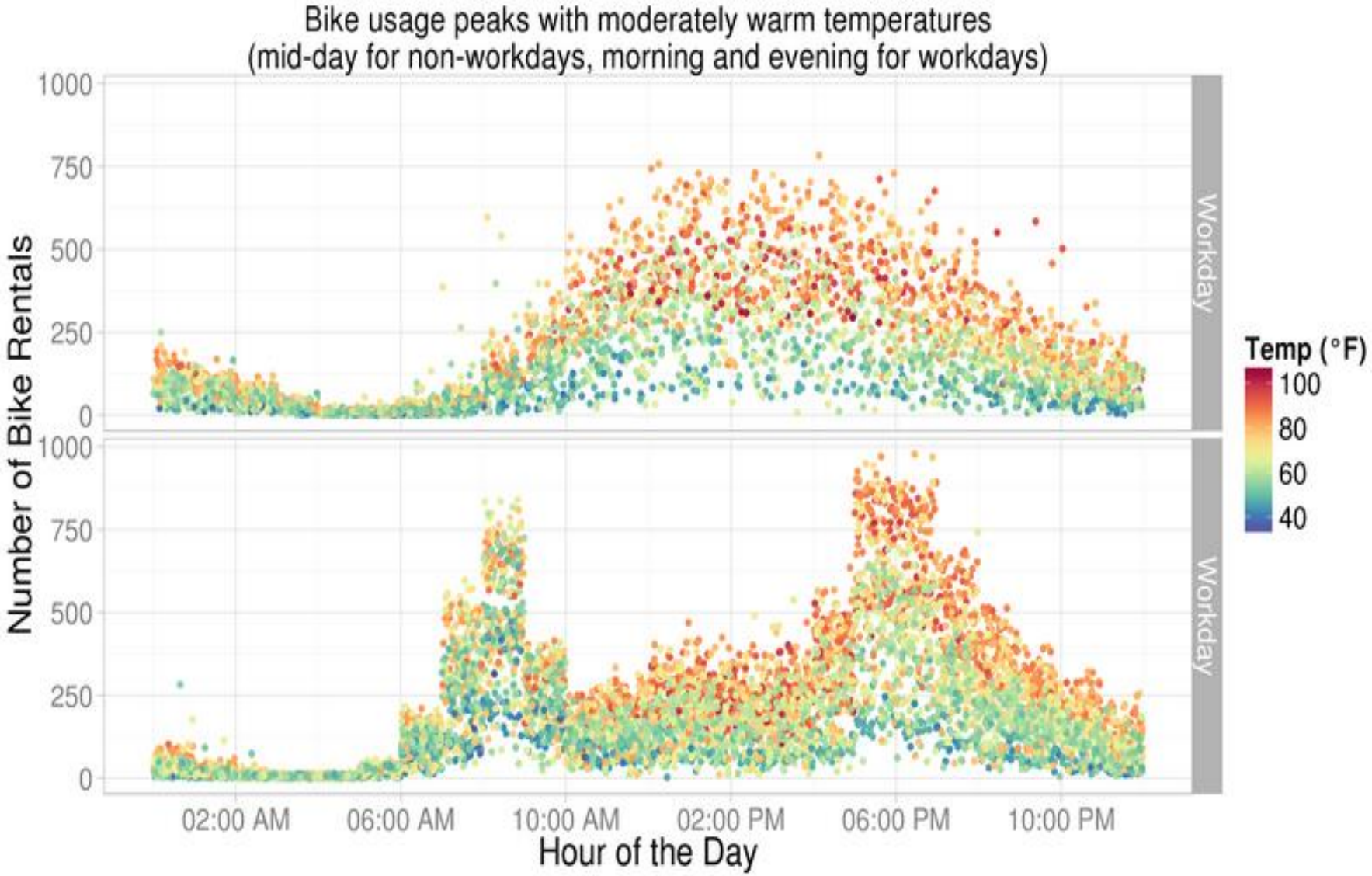


Data Exploring



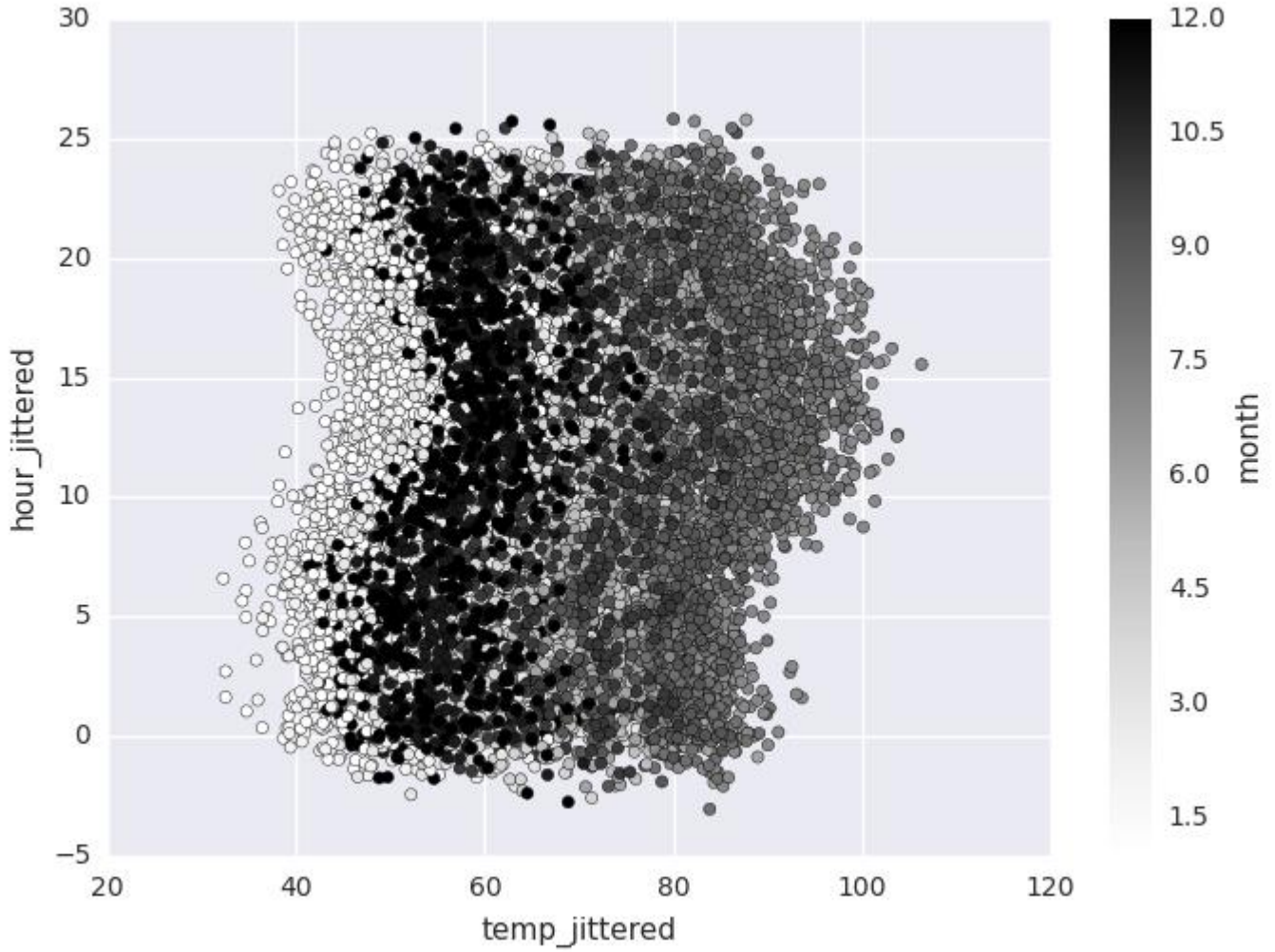
https://www.kaggle.com/users/993/ben-hamner/bike-sharing-demand/bike-rentals-by-time-and-temperature

Data Exploring



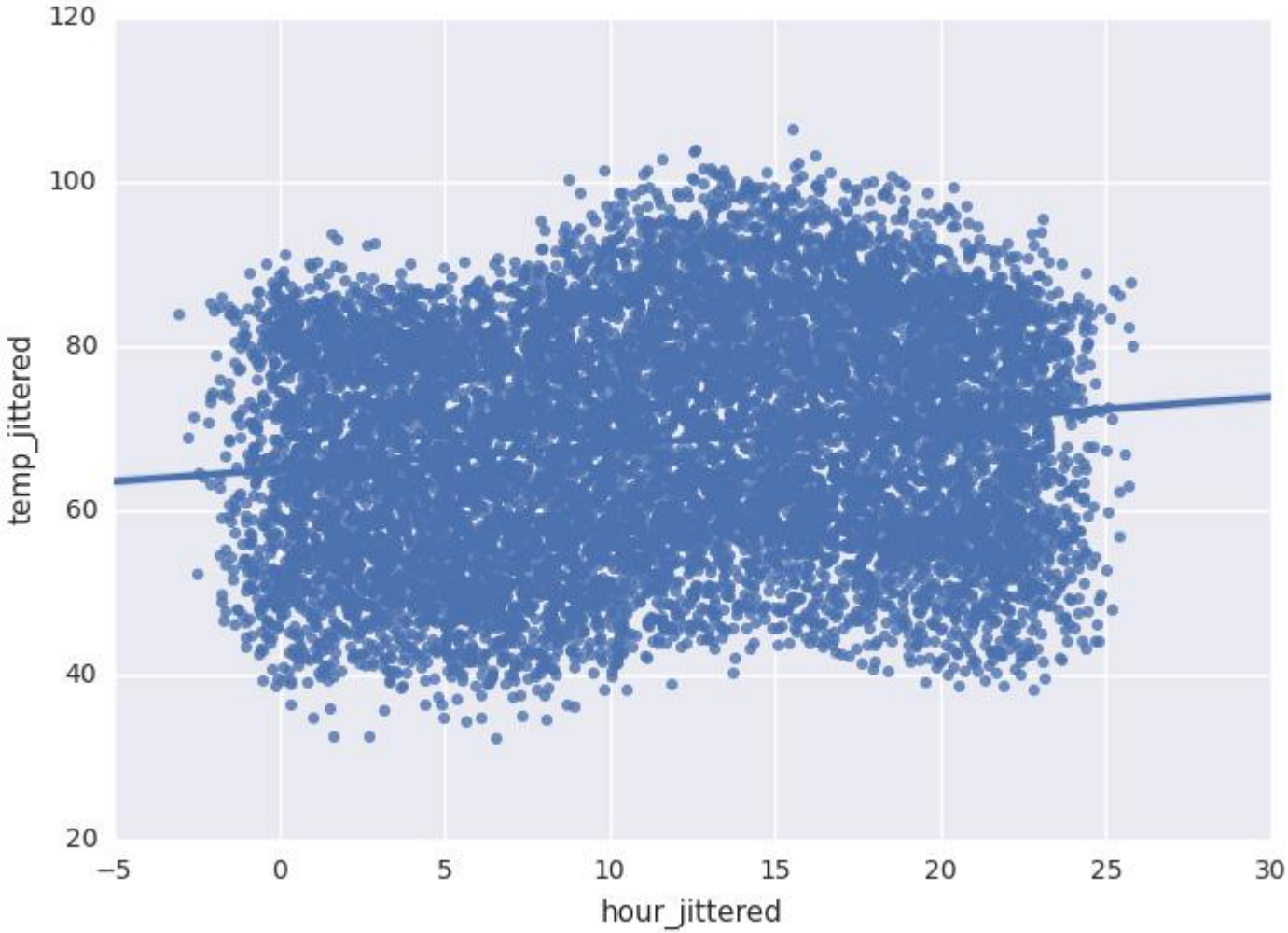
https://www.kaggle.com/users/338016/jatin-bhatia/bike-sharing-demand/rentals-by-time-temp-workingday

Data Exploring



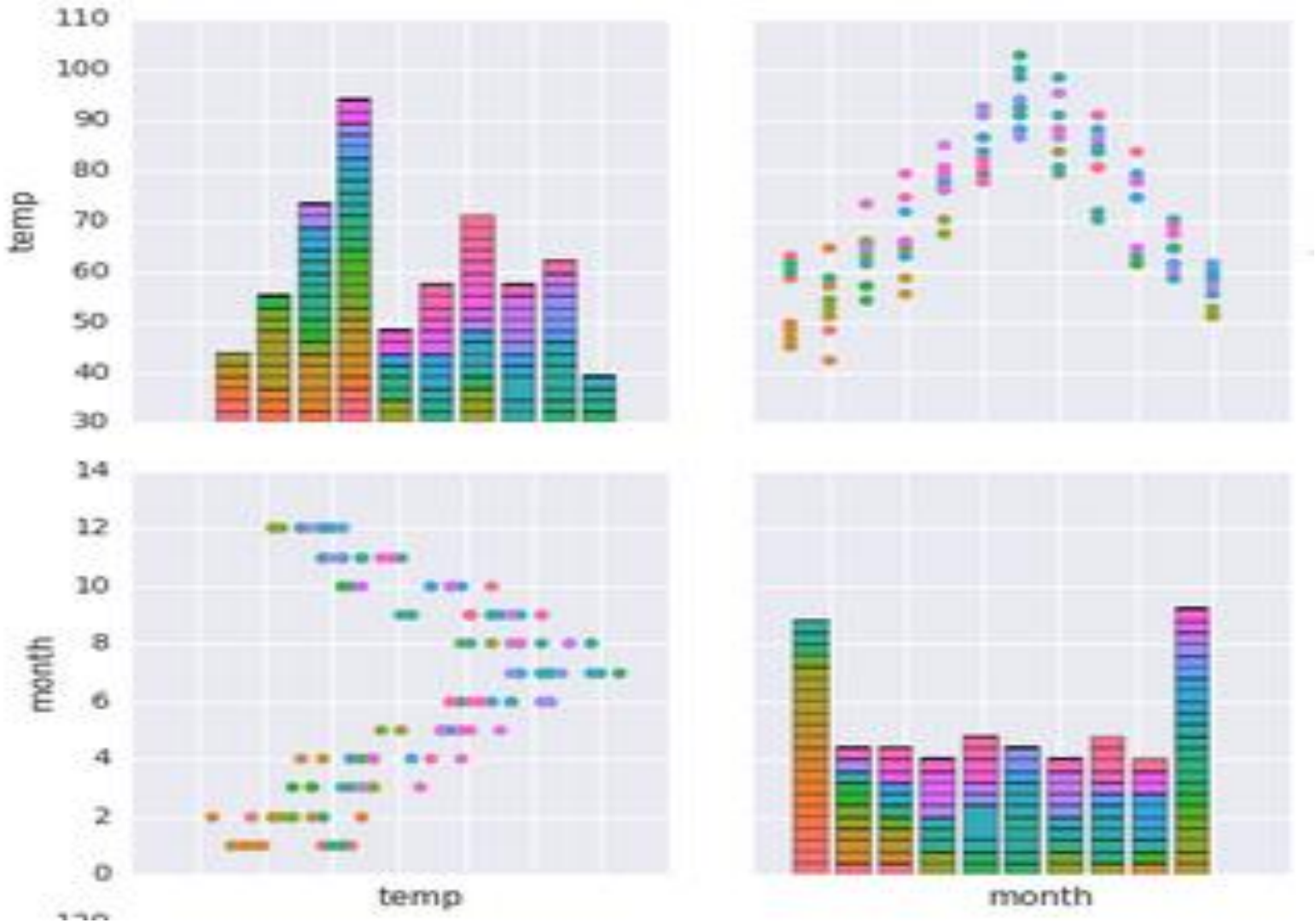
https://www.kaggle.com/users/993/ben-hamner/bike-sharing-demand/fiddling-with-python

Data Exploring



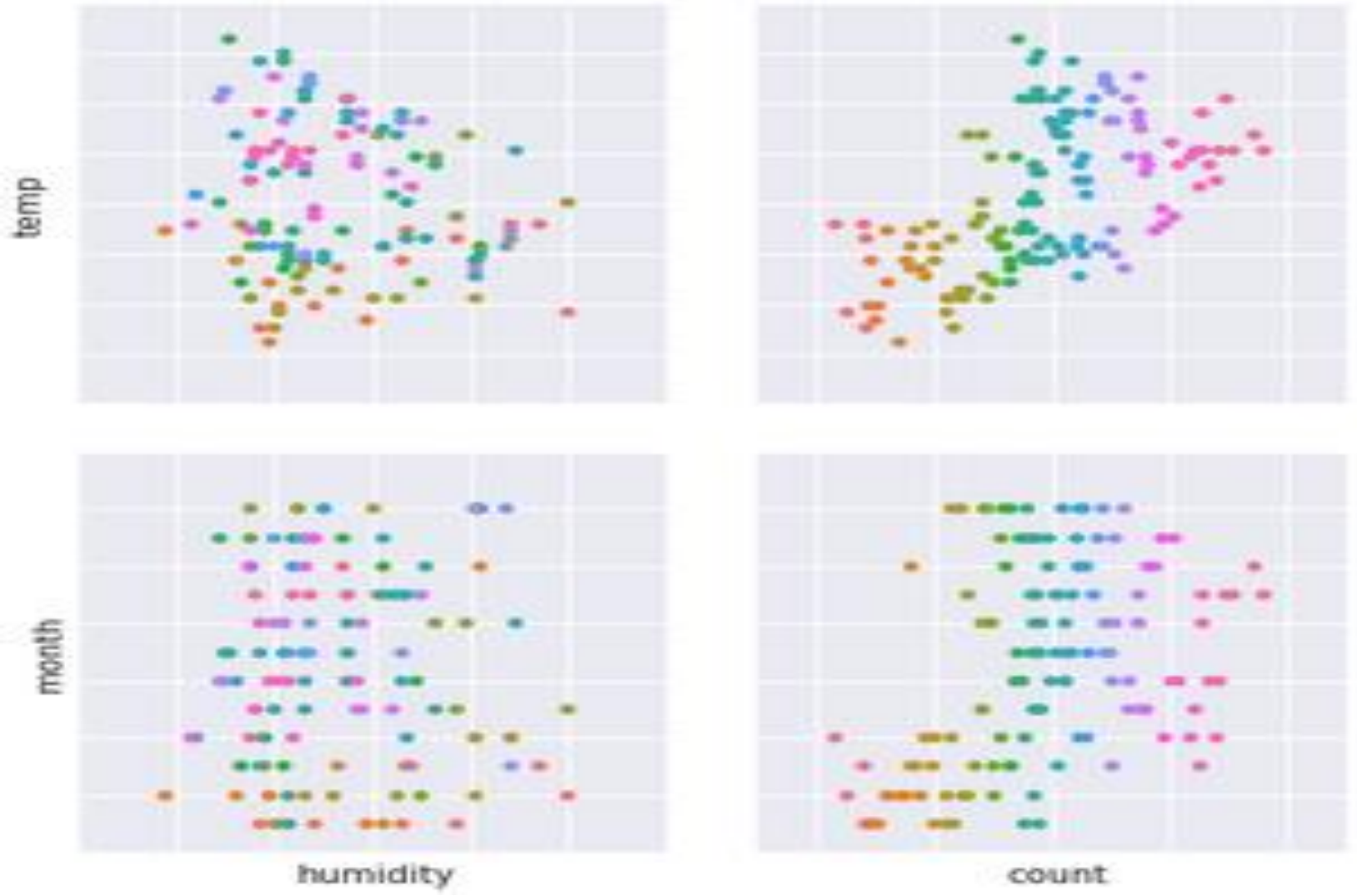
https://www.kaggle.com/users/993/ben-hamner/bike-sharing-demand/fiddling-with-python

Data Exploring



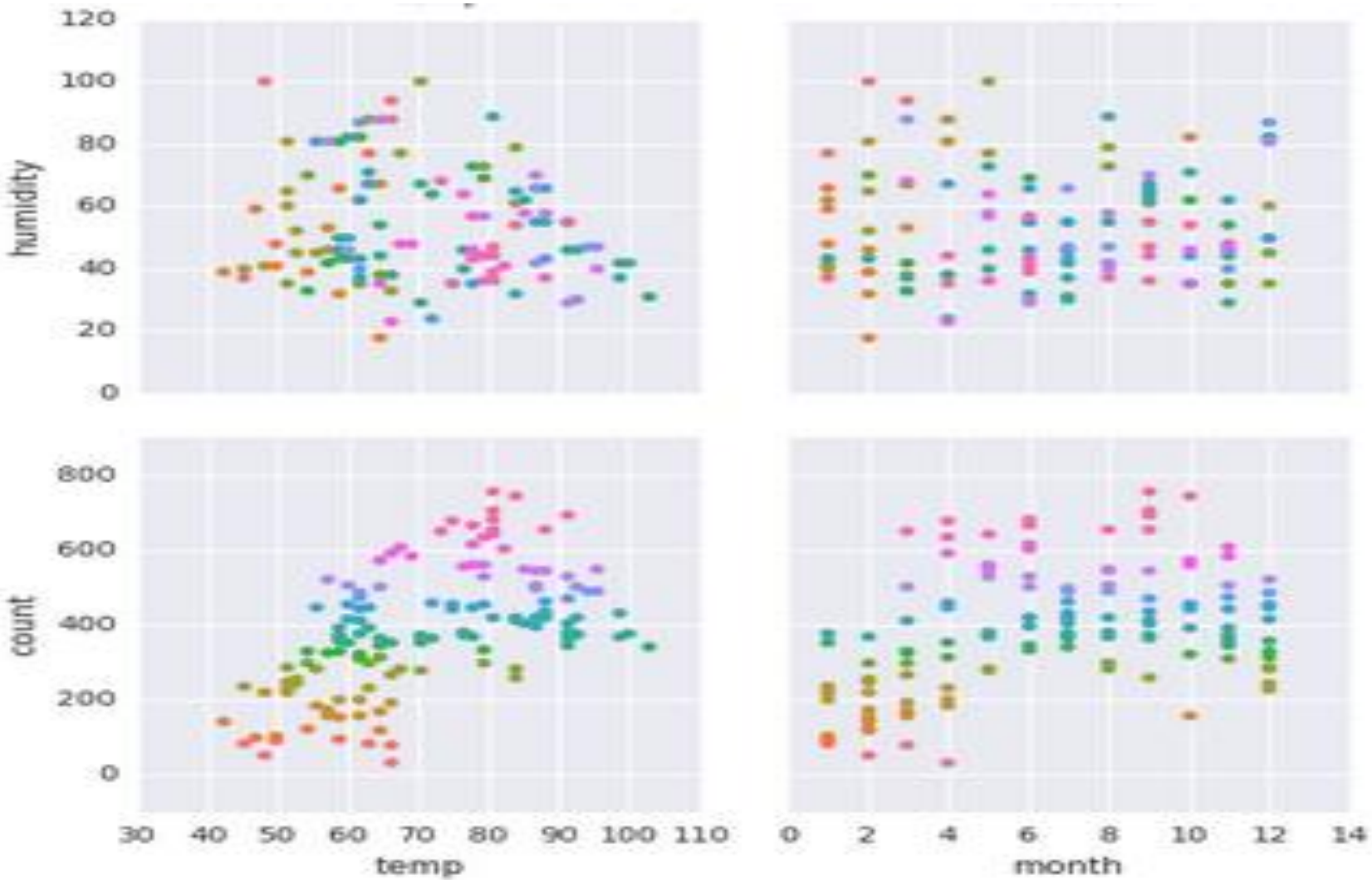
https://www.kaggle.com/users/993/ben-hamner/bike-sharing-demand/fiddling-with-python

Data Exploring



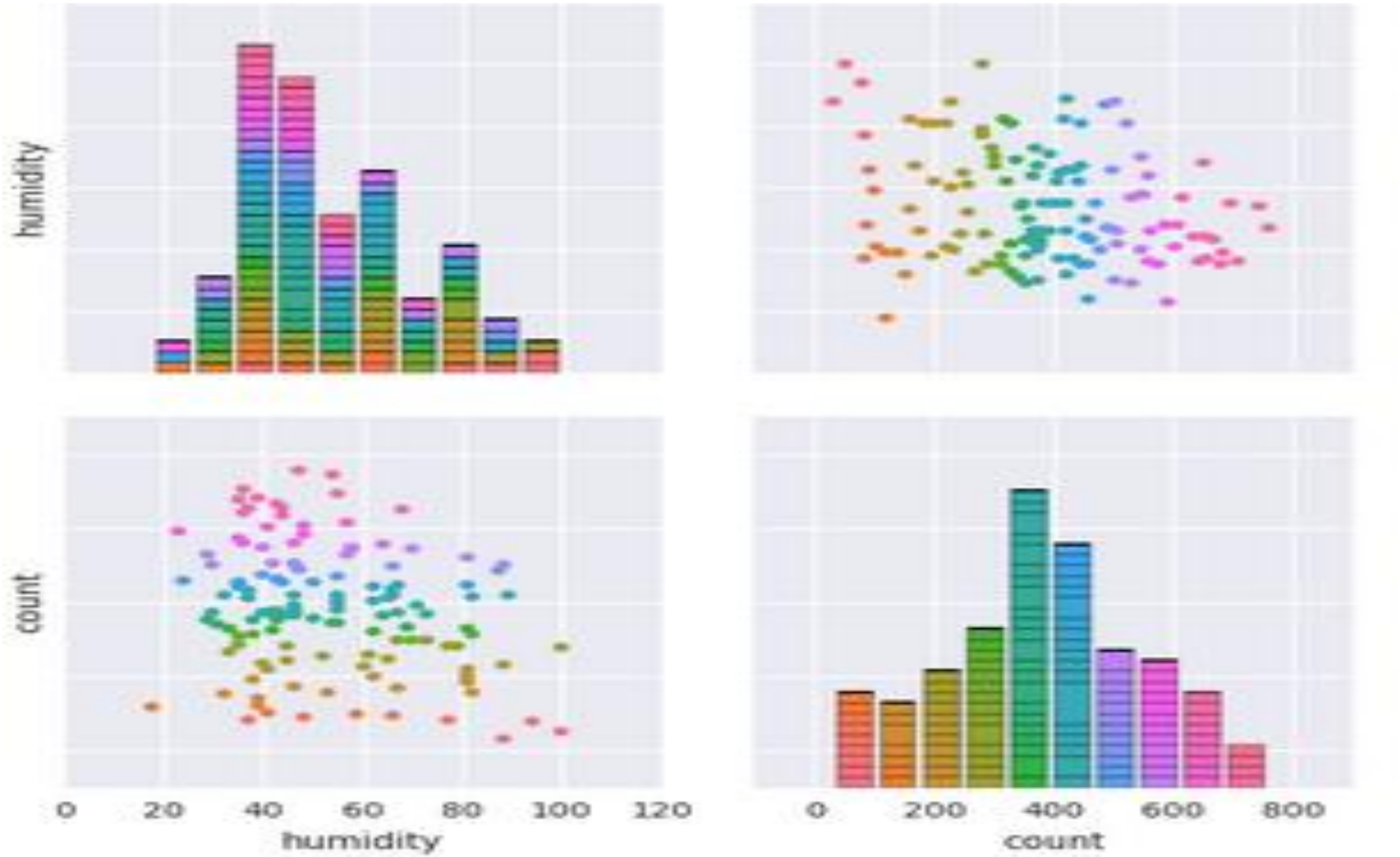
https://www.kaggle.com/users/993/ben-hamner/bike-sharing-demand/fiddling-with-python

Data Exploring



https://www.kaggle.com/users/993/ben-hamner/bike-sharing-demand/fiddling-with-python

Data Exploring



https://www.kaggle.com/users/993/ben-hamner/bike-sharing-demand/fiddling-with-python

Data Exploring

Summary of train dataset:

	season	holiday	workingday	weather	temp
count	10886.000000	10886.000000	10886.000000	10886.000000	10886.00000
mean	2.506614	0.028569	0.680875	1.418427	20.23086
std	1.116174	0.166599	0.466159	0.633839	7.79159
min	1.000000	0.000000	0.000000	1.000000	0.82000
25%	2.000000	0.000000	0.000000	1.000000	13.94000
50%	3.000000	0.000000	1.000000	1.000000	20.50000
75%	4.000000	0.000000	1.000000	2.000000	26.24000
max	4.000000	1.000000	1.000000	4.000000	41.00000

	atemp	humidity	windspeed	casual	registered
count	10886.000000	10886.000000	10886.000000	10886.000000	10886.000000
mean	23.655084	61.886460	12.799395	36.021955	155.552177
std	8.474601	19.245033	8.164537	49.960477	151.039033
min	0.760000	0.000000	0.000000	0.000000	0.000000
25%	16.665000	47.000000	7.001500	4.000000	36.000000
50%	24.240000	62.000000	12.998000	17.000000	118.000000
75%	31.060000	77.000000	16.997900	49.000000	222.000000
max	45.455000	100.000000	56.996900	367.000000	886.000000

	count
count	10886.000000
mean	191.574132
std	181.144454
min	1.000000
25%	42.000000
50%	145.000000
75%	284.000000
max	977.000000

https://www.kaggle.com/users/9028/danb/bike-sharing-demand/descriptive-statsimport-pandas-as-pd-imp

Data Exploring

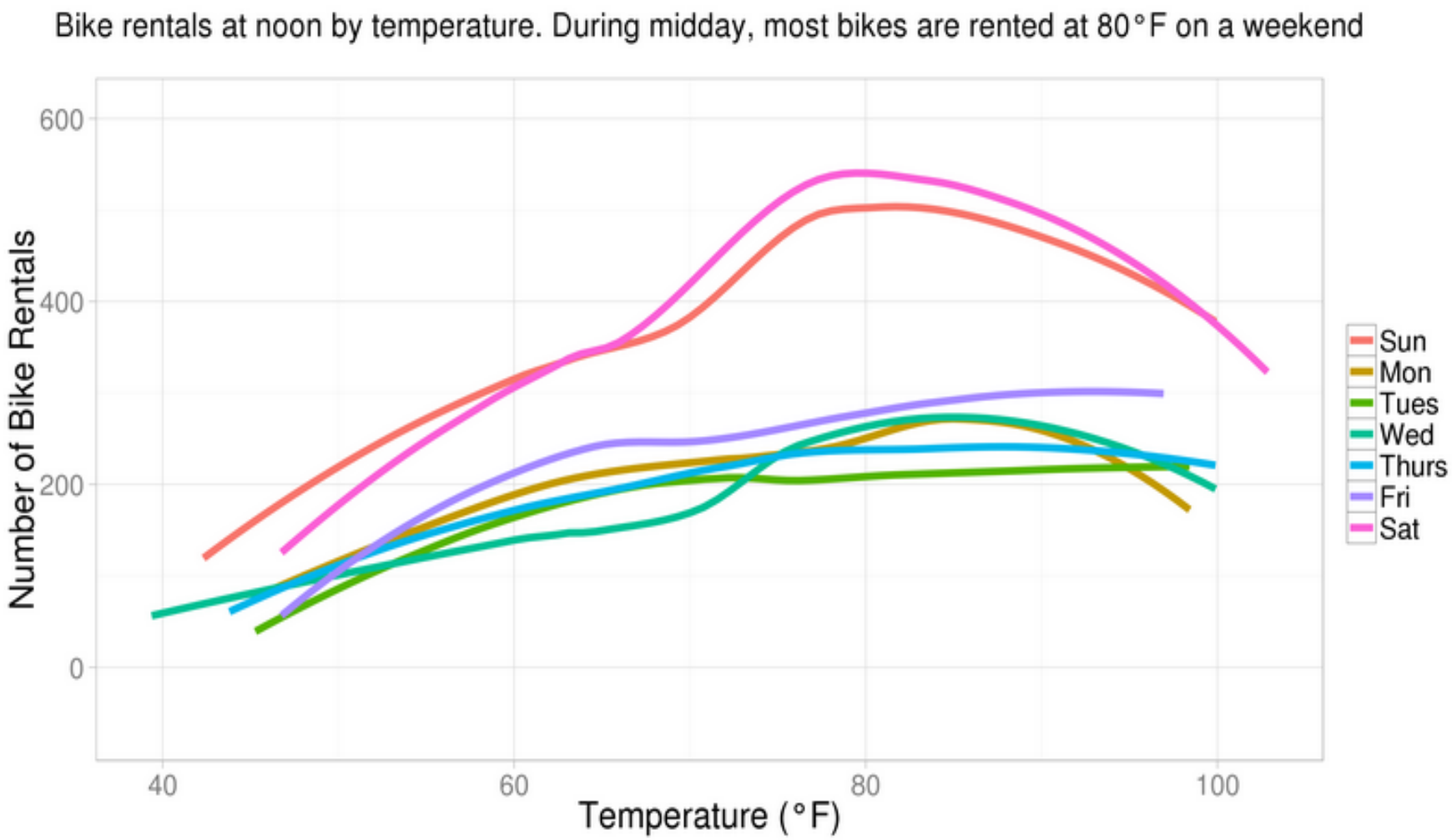
Summary of test dataset:

	season	holiday	workingday	weather	temp
count	6493.000000	6493.000000	6493.000000	6493.000000	6493.000000
mean	2.493300	0.029108	0.685815	1.436778	20.620607
std	1.091258	0.168123	0.464226	0.648390	8.059583
min	1.000000	0.000000	0.000000	1.000000	0.820000
25%	2.000000	0.000000	0.000000	1.000000	13.940000
50%	3.000000	0.000000	1.000000	1.000000	21.320000
75%	3.000000	0.000000	1.000000	2.000000	27.060000
max	4.000000	1.000000	1.000000	4.000000	40.180000

	atemp	humidity	windspeed
count	6493.000000	6493.000000	6493.000000
mean	24.012865	64.125212	12.631157
std	8.782741	19.293391	8.250151
min	0.000000	16.000000	0.000000
25%	16.665000	49.000000	7.001500
50%	25.000000	65.000000	11.001400
75%	31.060000	81.000000	16.997900
max	50.000000	100.000000	55.998600

https://www.kaggle.com/users/9028/danb/bike-sharing-demand/descriptive-statsimport-pandas-as-pd-imp

Data Exploring



https://www.kaggle.com/users/106206/jos-antonio-lvarez-cubero/bike-sharing-demand/prueba

Data Exploring

Bike Sharing Dataset Data Set

Download: Data Folder, Data Set Description

Abstract: This dataset contains the hourly and daily count of rental bikes between years 2011 and 2012 in Capital bikeshare system with the corresponding weather and seasonal information.

Data Set Characteristics:	Univariate	**Number of Instances:**	17389	**Area:**	Social
Attribute Characteristics:	Integer, Real	**Number of Attributes:**	16	**Date Donated**	2013-12-20
Associated Tasks:	Regression	**Missing Values?**	N/A	**Number of Web Hits:**	40423

Source:

Hadi Fanaee-T

Laboratory of Artificial Intelligence and Decision Support (LIAAD), University of Porto
INESC Porto, Campus da FEUP
Rua Dr. Roberto Frias, 378
4200 - 465 Porto, Portugal

Original Source: http://capitalbikeshare.com/system-data
Weather Information: http://www.freemeteo.com
Holiday Schedule: http://dchr.dc.gov/page/holiday-schedule

https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset