Bike Rental Count

Project Report

Sharang Shrivastava

22 January , 2019

Contents

1.Overview

Bike sharing systems are a means of renting bicycles where the process of obtaining membership, rental, and bike return is automated via a network of kiosk locations throughout a city. Using these systems, people are able rent a bike from a one location and return it to a different place on an as-needed basis.

The objective of this Case is to Predication of bike rental count on daily based on the environmental and seasonal settings. The details of data attributes in the dataset are as follows -

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 or 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
hum- relative humidity
windspeed - wind speed
casual - number of non-registered user rentals initiated
registered - number of registered user rentals initiated
cnt - number of total rentals (Dependent Variable)

2. Data Summary

As a first step let's do three simple steps on the dataset

- Size of the dataset
- Get a glimpse of data by printing few rows of it.
- What type of variables contribute our data

Shape of data : 731 rows , 16 columns

Sample Of First Few Rows

instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered	cnt
1	2011-01-01	1	0	1	0	6	0	2	0.344167	0.363625	0.805833	0.160446	331	654	985
2	2011-01-02	1	0	1	0	0	0	2	0.363478	0.353739	0.696087	0.248539	131	670	801
3	2011-01-03	1	0	1	0	1	1	1	0.196364	0.189405	0.437273	0.248309	120	1229	1349
4	2011-01-04	1	0	1	0	2	1	1	0.200000	0.212122	0.590435	0.160296	108	1454	1562
5	2011-01-05	1	0	1	0	3	1	1	0.226957	0.229270	0.436957	0.186900	82	1518	1600

Variable Summary

feature	count	mean	std	min	25%	50%	75%	max
instant	731	366	211.1658	1	183.5	366	548.5	731
season	731	2.4966	1.1108	1	2	3	3	4
yr	731	0.5007	0.5003	0	0	1	1	1
mnth	731	6.5198	3.4519	1	4	7	10	12
holiday	731	0.0287	0.1672	0	0	0	0	1
weekday	731	2.9973	2.0048	0	1	3	5	6
workingday	731	0.684	0.4652	0	0	1	1	1
weathersit	731	1.3953	0.5449	1	1	1	2	3
temp	731	0.4954	0.1831	0.0591	0.3371	0.4983	0.6554	0.8617
atemp	731	0.4744	0.163	0.0791	0.3378	0.4867	0.6086	0.8409
hum	731	0.6279	0.1424	0	0.52	0.6267	0.7302	0.9725
windspeed	731	0.1905	0.0775	0.0224	0.1349	0.181	0.2332	0.5075
casual	731	848.1765	686.6225	2	315.5	713	1096	3410
registered	731	3656.1724	1560.2564	20	2497	3662	4776.5	6946
cnt	731	4504.3488	1937.2115	22	3152	4548	5956	8714

3. Exploratory Data Analysis

3.1 Feature Engineering

As we see from the above results, the columns "season","holiday","workingday" and "weathersit" should be of "categorical" data type.But the current data type is "int" for those columns. We will transform the dataset in the following ways so that we can get started up with our EDA

Coerce the datatype of "season","holiday","workingday" and weather to category

We don't require dteday column as we already have separate columns as “yr” for year and “mnth” for months . We also don't require instant column , we will drop both of these from our dataset

3.2 Missing Values Analysis

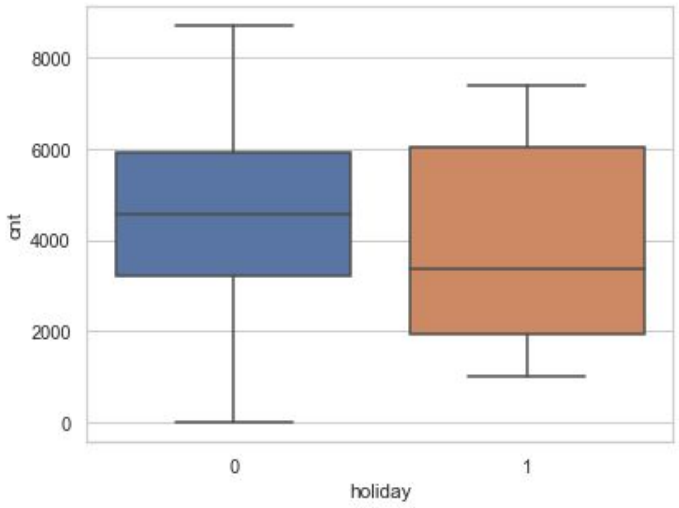
Once we get hang of the data and columns, next step we generally is to find out whether we have any missing values in our data. Luckily we don't have any missing value in the dataset.

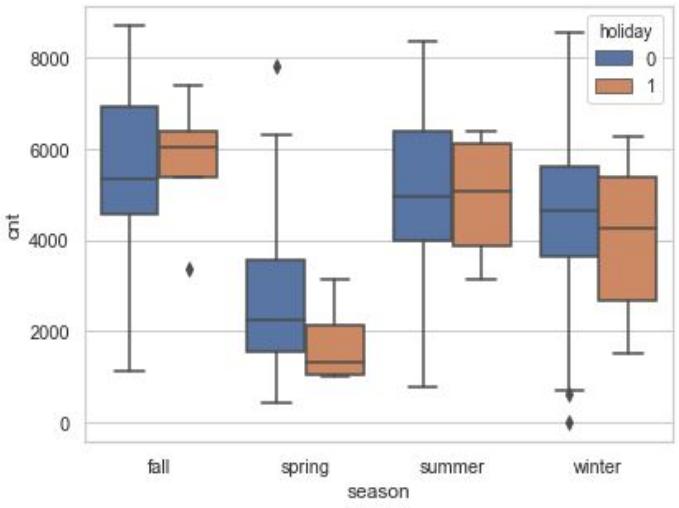
3.3 Outliers Analysis

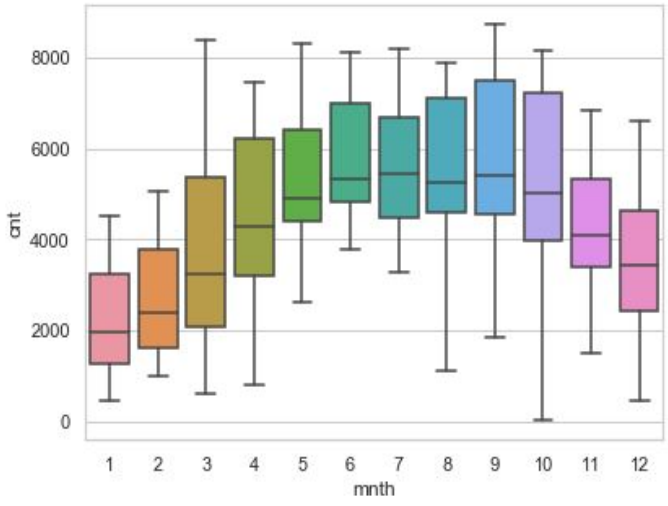
We can see that there are only few outliers present in count variable with respect to season

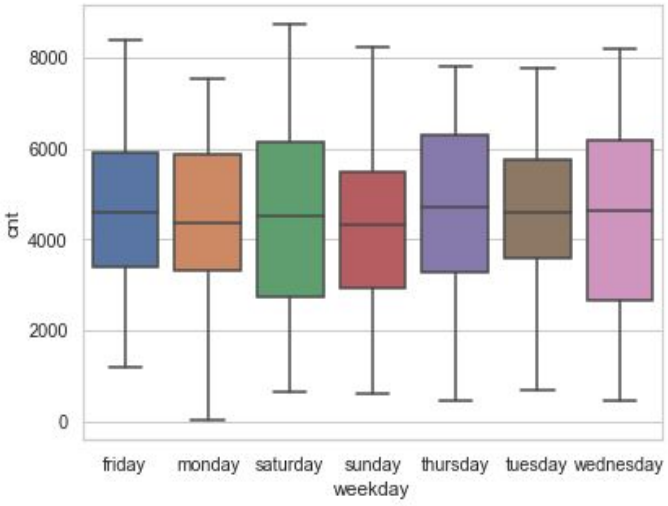
There is one outlier we can see that belongs to spring season having count unexpectedly high , we will remove it..
There is one more outlier in count when season is winter, we will confirm whether that is due to weather conditions. We can see that count = 22 in the last row is really an outlier as with same weather conditions , count for other rows are very much higher than 22 , with least values 627,705 hence we will remove count=22 row .

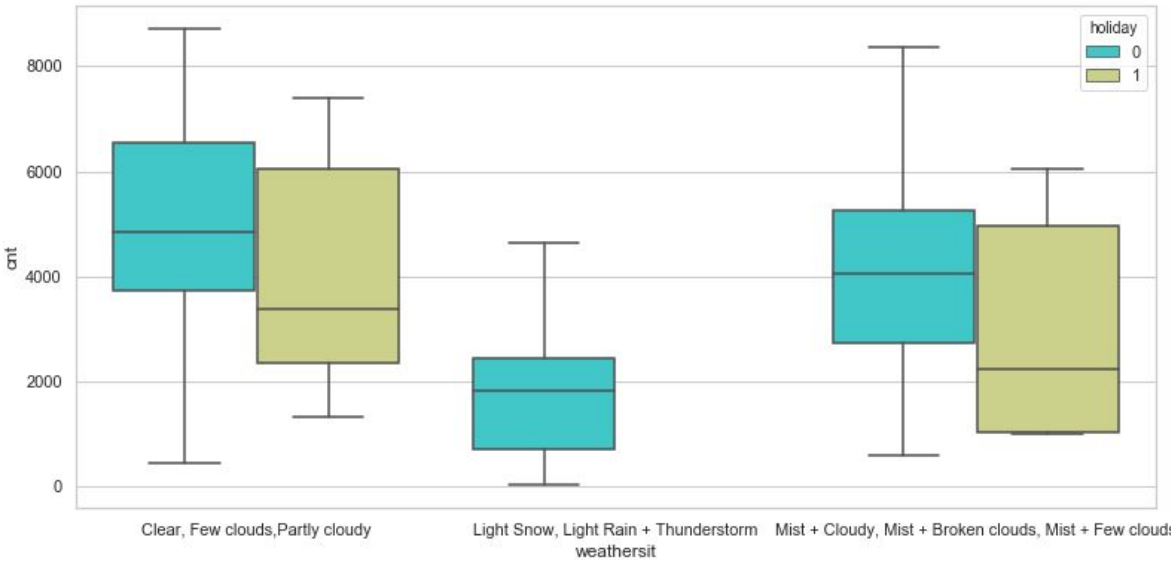
Following inferences can also been made from the simple box plots given below.
Spring season has got relatively lower count.The dip in median value in boxplot gives evidence for it.
The boxplot with “Month” is quite interesting.The median value are relatively higher from august to september .

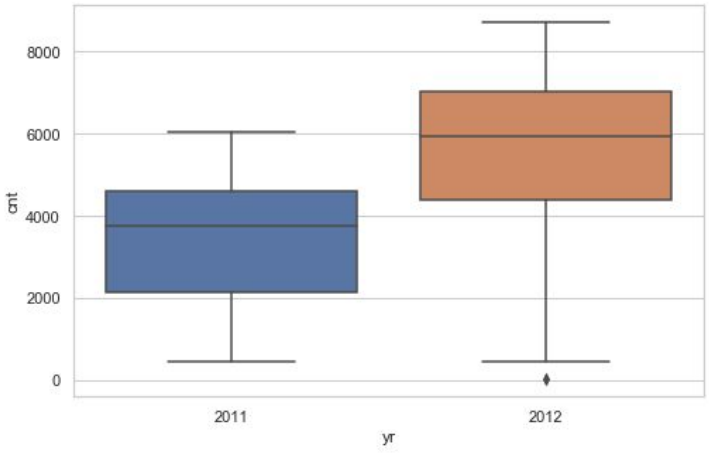

8000
6000
4000
2000
0
cnt
0
1
holiday

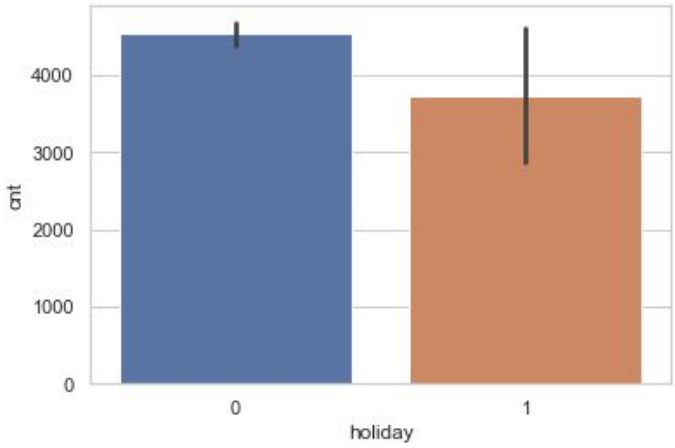

holiday
0
1
8000
6000
4000
2000
0
cnt
fall
spring
summer
winter
season


8000
6000
4000
2000
0
cnt
1
2
3
4
5
6
7
8
9
10
11
12
mnth


8000
6000
4000
2000
0
cnt
friday
monday
saturday
sunday
thursday
tuesday
wednesday
weekday



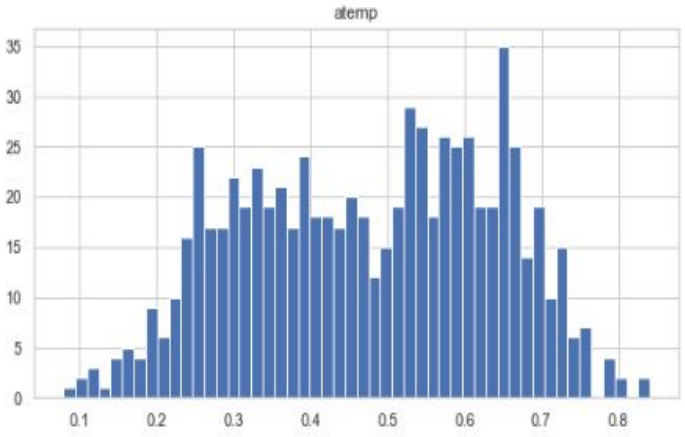


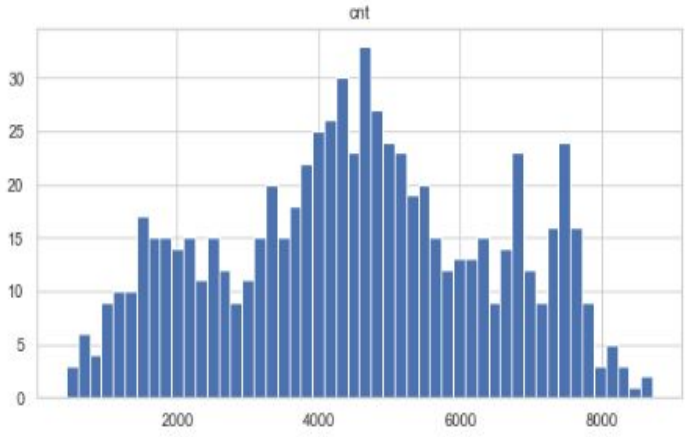


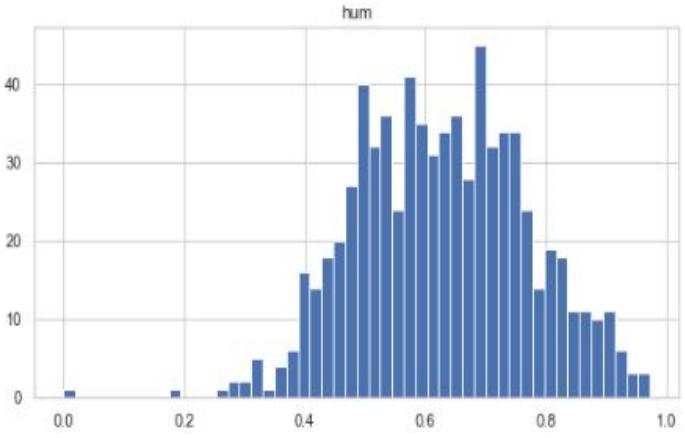
We can infer from the figure above is that

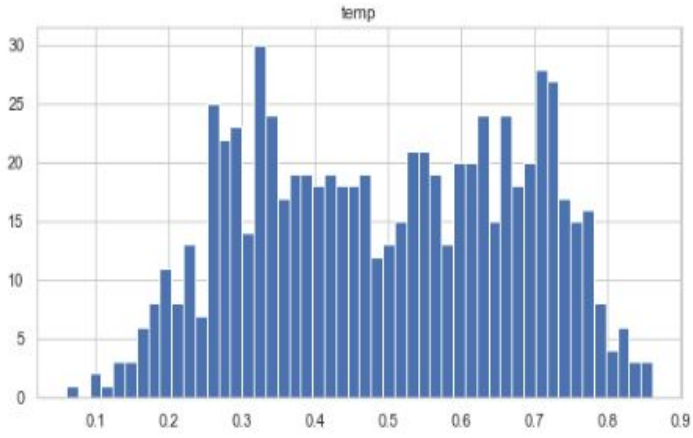
In year 2012 , count value is more relatively than 2011
Value of count is comparatively lesser in the case when weather represents light snow,rain and thunderstorm , as it is quite obvious in extreme weather ,demand drops
There is lesser demand in case of holiday ..

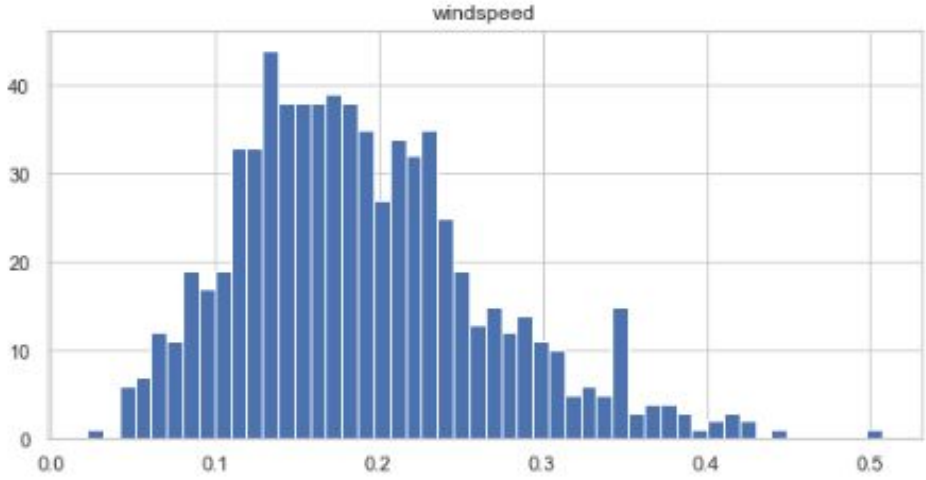
Now will see how the data is distributed in other independent variables









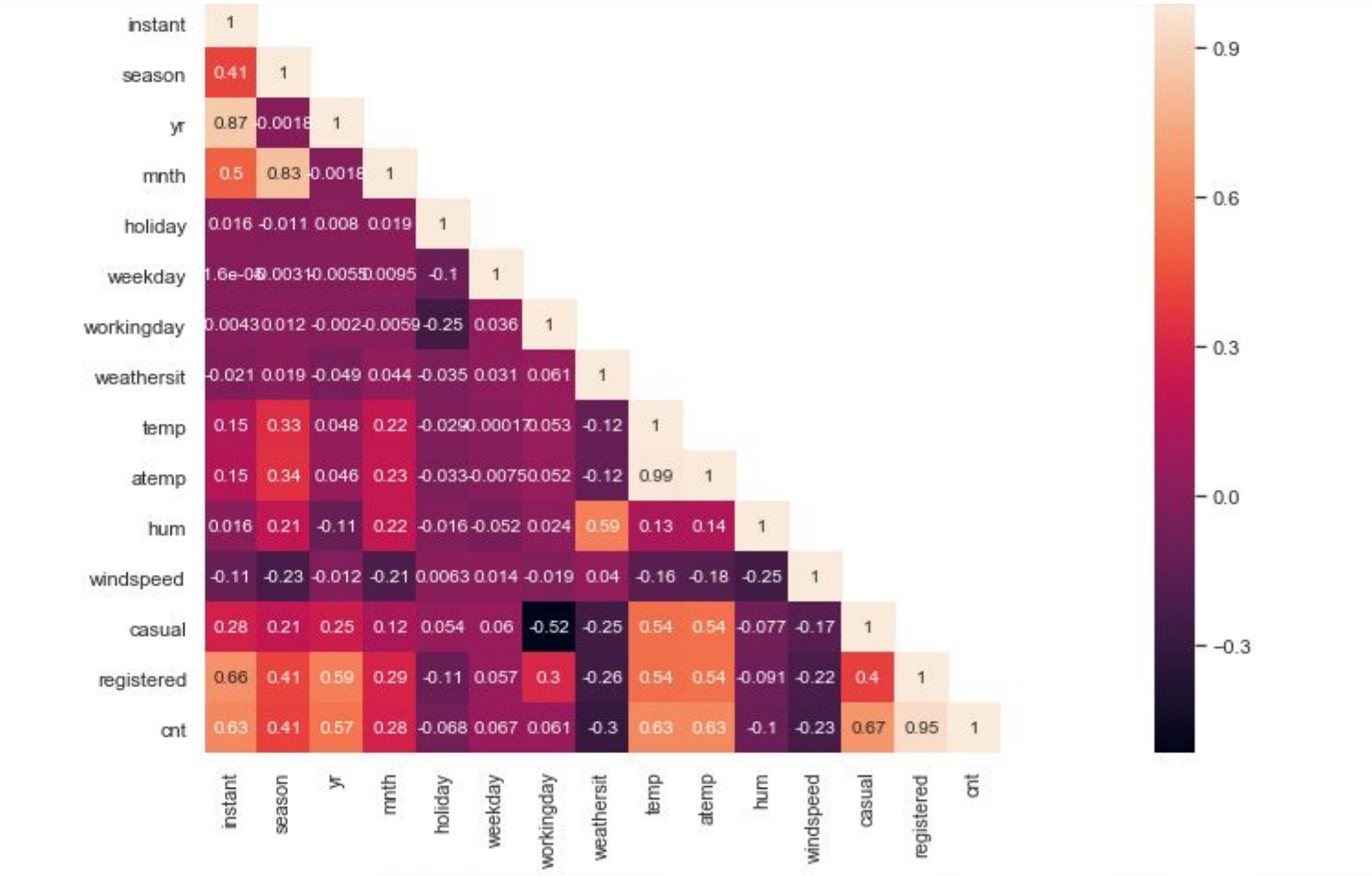


4. Correlation Analysis

One common to understand how a dependent variable is influenced by features (numerical) is to find a correlation matrix between them. Lets plot a correlation plot

The variable we are going to predict is the “count”. So let’s look at how much each independent variable correlates with this dependent variable.

```
cnt           1.000000
registered    0.945111
casual        0.672063
atemp         0.632732
temp          0.628887
yr            0.572372
season        0.412430
mnth          0.284421
weekday       0.064563
workingday    0.063566
holiday      -0.069149
hum          -0.095597
windspeed    -0.229254
weathersit   -0.290843
```

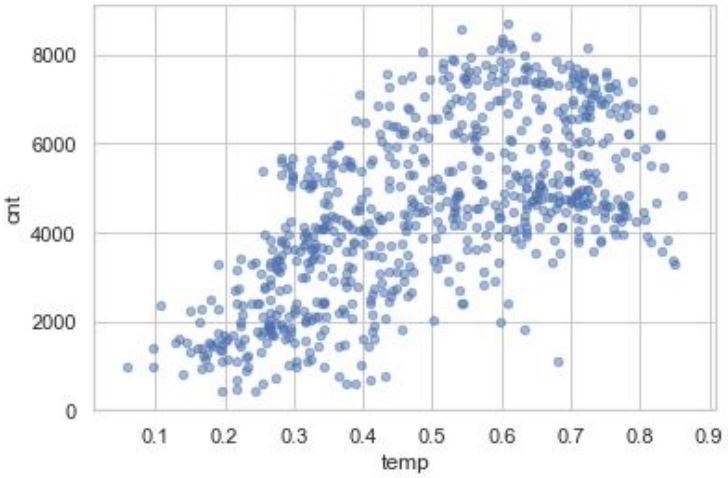


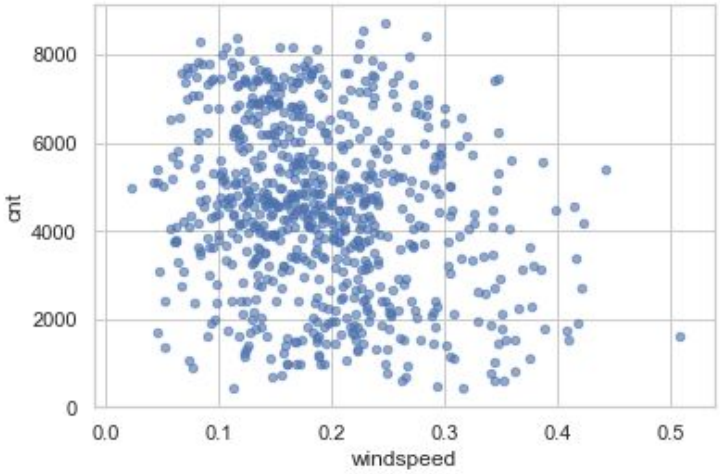
We can see that there is pretty much strong correlation of count with respect to atemp,temp,yr,season and month. We will ignore casual and registered variables as they are directly related with count variable. Windspeed shows negative correlation as it is obvious with high value of windspeed , demand will definitely drop. Same is the case with weather variable.

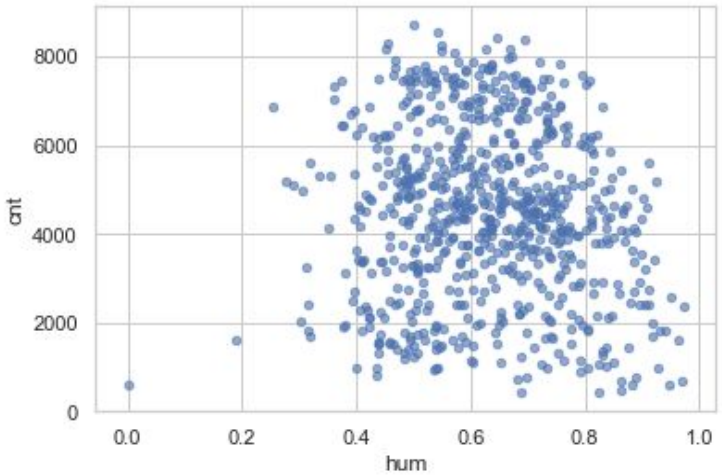
Now we will see how these variables are related with respect to each other.

temp and atemp are highly correlated,correlation value is 0.99. We will drop atemp variable as both are same.

The most promising variable for predicting the count is the temp, so lets look at their correlation scatter plot.







The correlation is indeed strong; you can clearly see the upward trend and that the points are not too dispersed. In case of humidity and windspeed , data is too much dispersed , no correlation at all.

5. Model building

x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=30)

Shape of train and test dataset

Train data (511, 10)
Test data (219, 10)

5.1 Linear regression Model

LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=True)

R squared value ::	0.8113
Linear Regression Root Mean Squared Error :	831.9710
Linear Regression Root Mean Squared Log Error:	0.2442
Linear Regression Mean Absolute Error :	617.5498

5.2 Lasso (**least absolute shrinkage and selection operator**)

Lasso(alpha=1.0, copy_X=True, fit_intercept=True, max_iter=1000,
  normalize=False, positive=False, precompute=False, random_state=None,
  selection='cyclic', tol=0.0001, warm_start=False)

R squared value ::	0.810241940891103
Lasso Root Mean Squared Error :	834.3916
Lasso Root Mean Squared Log Error:	0.2455
Lasso Mean Absolute Error :	619.674127

5.3 Gradient boost Regressor

GradientBoostingRegressor(alpha=0.9, criterion='friedman_mse', init=None,
        learning_rate=0.1, loss='ls', max_depth=3, max_features=None,
        max_leaf_nodes=None, min_impurity_decrease=0.0,
        min_impurity_split=None, min_samples_leaf=1,
        min_samples_split=2, min_weight_fraction_leaf=0.0,
        n_estimators=100, n_iter_no_change=None, presort='auto',
        random_state=None, subsample=1.0, tol=0.0001,
        validation_fraction=0.1, verbose=0, warm_start=False)

Gradient boost regressor R squared value ::	0.8910702019919841
Gradient boost regressor Root Mean Squared Error :	632.1835049178984
Gradient boost regressor Root Mean Squared Log Error:	0.2096
Gradient boost regressor Mean Absolute Error :	461.9865

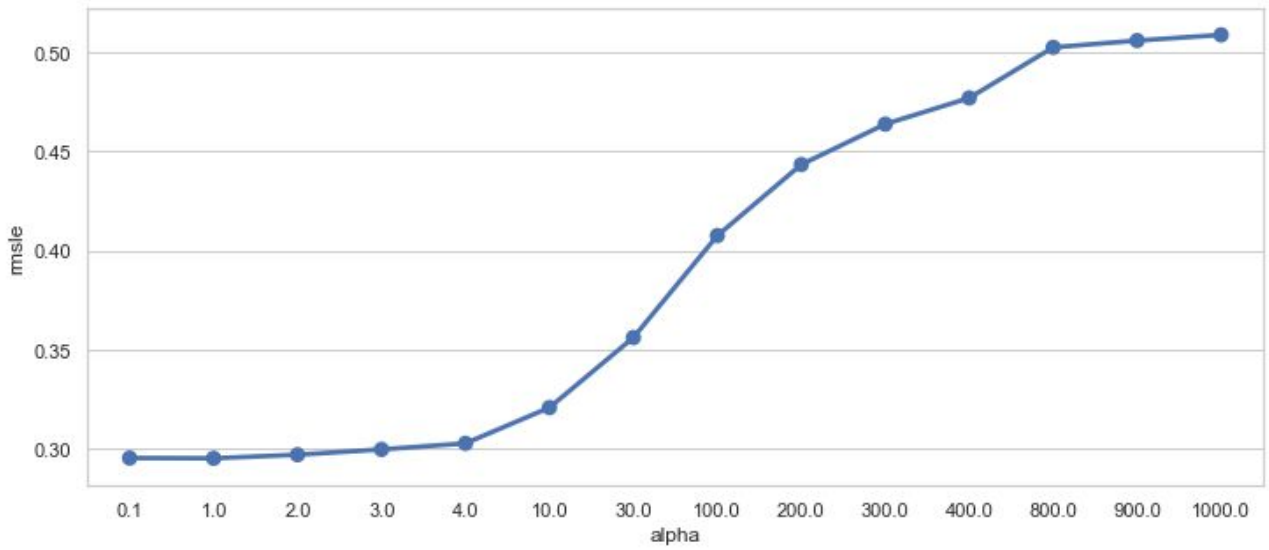
5.4 Regularization Model - Ridge

We will use the grid search also along with this.We have given a list of values for alpha parameter.
ridge_params_= {'max_iter' : [3000] , 'alpha' : [0.1, 1, 2, 3, 4, 10, 30,100,200,300,400,800,900,1000]}
After fitting model to the data, we get

Best Estimators

Ridge(**alpha=1**, copy_X=True, fit_intercept=True, **max_iter=3000**,
normalize=False, random_state=None, solver='auto', tol=0.001)

Ridge Root Mean Squared Log Error : 0.23137216355261897



Model has also given the best value of alpha parameter and that is 1
From the above figure , it is clear that for alpha = 1 , rmsle value is the lowest.

5.5 Ensemble Tree Based Models

We have applied some other ensemble tree based models like **RandomForestRegressor,AdaBoostRegressor,BaggingRegressor, and SVR, KNeighborsRegressor also.**

And we have framed the results in data frame and sorted the values based on greater R_Squared values. We have also plotted feature importance alongside , once again it is proved that temp has contributed a lot in predicting count values.

	Modelling Algo	R_Squared_test	RMSLE	RMSE	MAE
0	GradientBoostingRegressor	0.905677	0.180929	587.911744	441.217186
1	RandomForestRegressor	0.876008	0.206850	667.115145	486.025571
2	BaggingRegressor	0.871050	0.205696	680.321035	486.409589
3	LinearRegression	0.822731	0.225050	797.665254	607.942972
4	Lasso	0.821963	0.225622	799.389697	607.890438
5	AdaBoostRegressor	0.816154	0.246809	812.326404	651.119913
6	KNeighborsRegressor	0.759404	0.266448	929.283195	713.418265
7	SVR	0.009991	0.504275	1885.052240	1541.867575

	features	importance
0	temp	0.480248
1	yr	0.299619
2	season	0.075415
3	hum	0.063461
4	windspeed	0.031682
5	mnth	0.019056
6	weathersit	0.017048
7	weekday	0.007964
8	workingday	0.003164
9	holiday	0.002343

From the above figure , it is clear that Gradient Boosting Regressor is predicting bike rental count values with least error. We will try to tune the parameters of model and optimize it.

6 Model Tuning (Gradient Boosting Regressor)

Now we will tune the parameters of gradient boost regressor to help optimize our model
We will set learning rate = 0.1904 and random_state=30 . We will see whether this improves the model statistics metrics or not

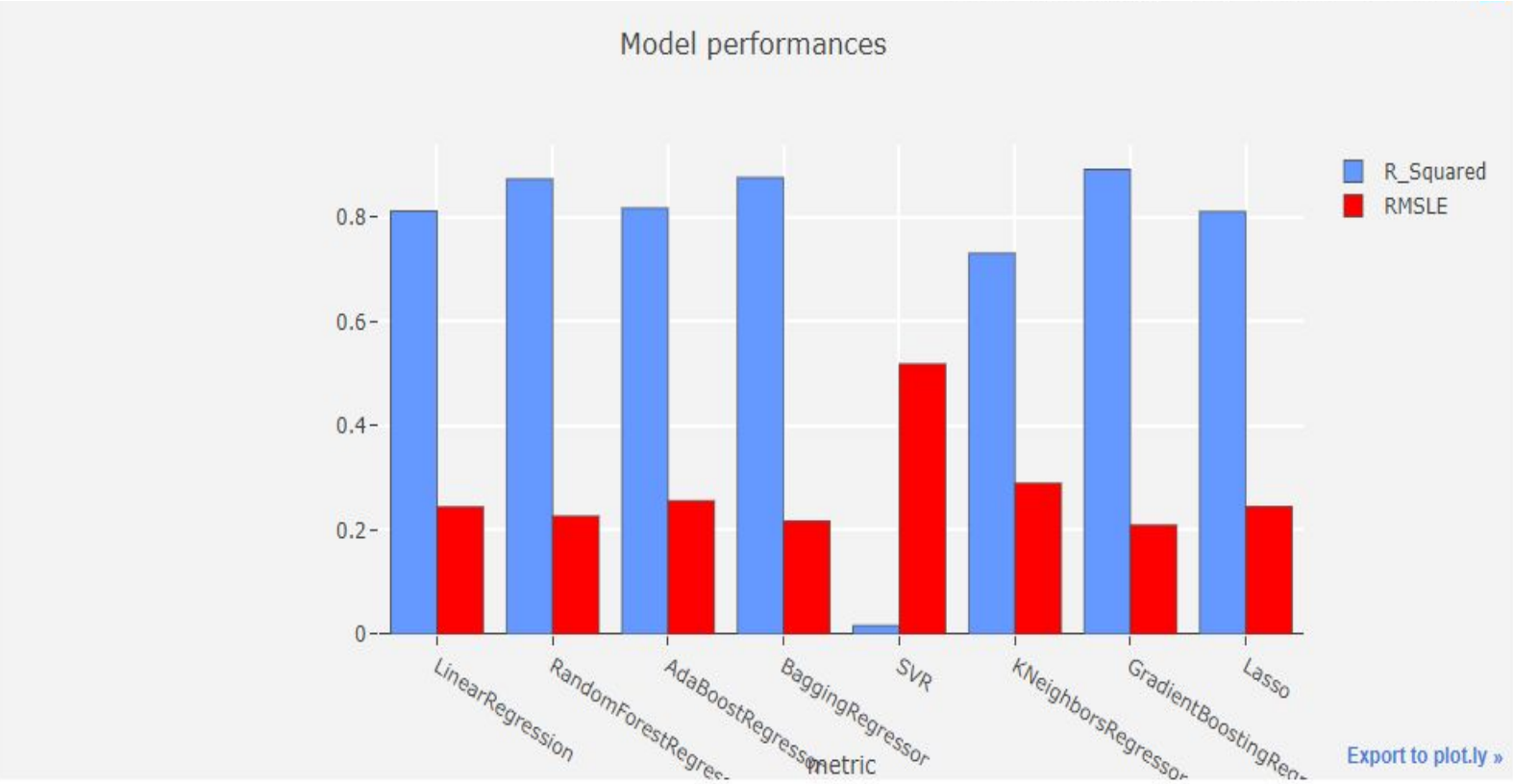
GradientBoostingRegressor(alpha=0.9, criterion='friedman_mse', init=None,
learning_rate=0.1899, loss='ls', max_depth=3,
max_features=None, max_leaf_nodes=None,
min_impurity_decrease=0.0, min_impurity_split=None,
min_samples_leaf=1, min_samples_split=2,
min_weight_fraction_leaf=0.0, n_estimators=100,
n_iter_no_change=None, presort='auto', **random_state=30**,
subsample=1.0, tol=0.0001, validation_fraction=0.1, verbose=0,
warm_start=False)

By fitting the model again with data and new parameters , we get the following values :

	New Values	**Old Values**
R squared value ::	0.905677	0.90370
Root Mean Squared Error :	598.5074334190634	632.1835049178984
Root Mean Squared Log Error:	0.180929	0.18302
Mean Absolute Error :	438.9757	461.9865

We can see that MSE, RMSLE and MAE now have reduced value, that means our tuned model is predicting with less error. This is the best we can get from the model with new parameters.

7. Model Performance Comparison



So we have build several models , and we have chosen the **Gradient Boosting Regressor** as the best model that is going to predict bike rental count with least error.