Predicting Bike Count

Using Selected Features

Data, Qs, Methods

Dataset sources, questions, and methods used

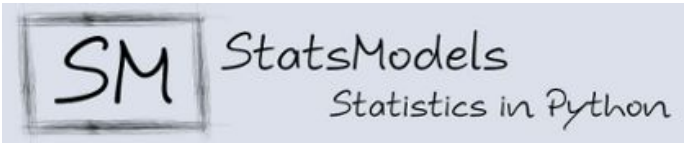
Dataset Sources

- Main set of data concerning the bikes originated from **Capital Bikeshare System**, Washington D.C. ranging from 2011-2012.
- Other data, such as weather, retrieved from **i-weather.com**.
- Found already cleaned and organized on **UCI's Machine Learning Repository**.

[1] Fanaee-T, Hadi, and Gama, Joao, "Event labeling combining ensemble detectors and background knowledge", Progress in Artificial Intelligence (2013): pp. 1-15, Springer Berlin Heidelberg, doi:10.1007/s13748-013-0040-3.

Methods and Libraries

- ScikitLearn (Linear Regression)
- Pandas (Data Frame)
- Statsmodels (OLS summary)
- Matplotlib (scatterplots, line graphs)



Questions:

- How many bikes are there on any given day?
- Based on selected features, are the number of bikes rented predictable?
- Which features are important in predicting bike count?

Why Bike count?

- Distribution of bikes through the system based on predicted demand.
- Profit potential based on bike count under particular conditions.
- Increase advertisement during potentially high bike count days.

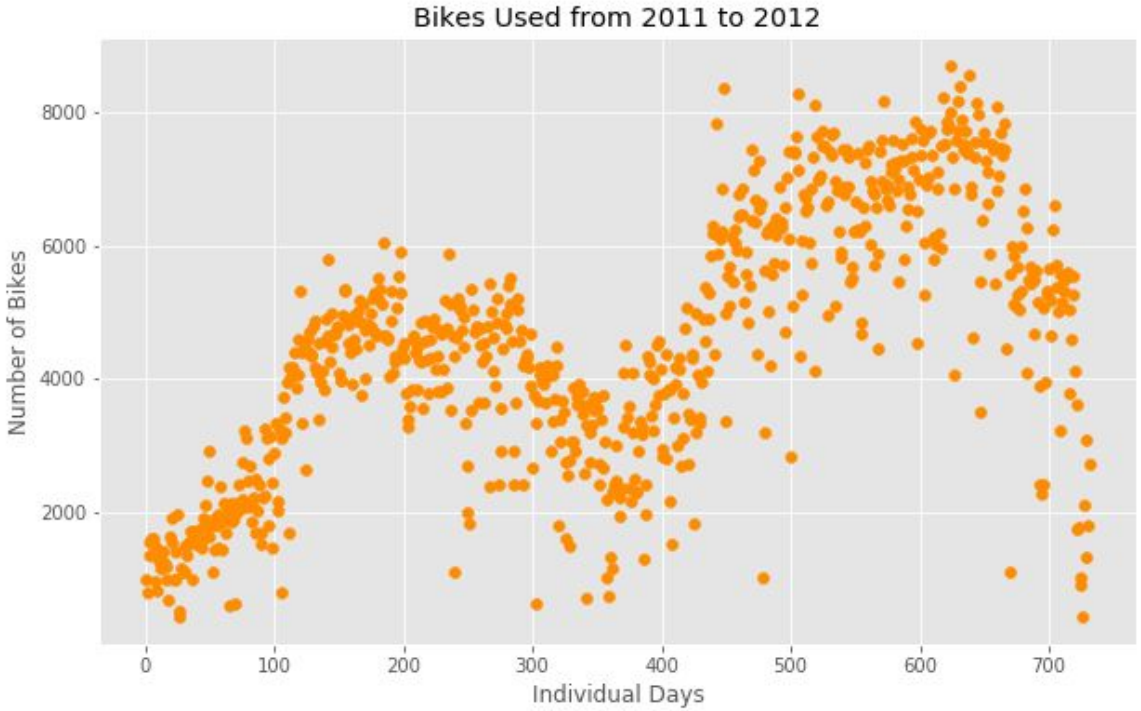
Selected Features

- Temperature
- Humidity
- Weather
- Season
- Working Day

Visualizations

Regressions, Models, and Features

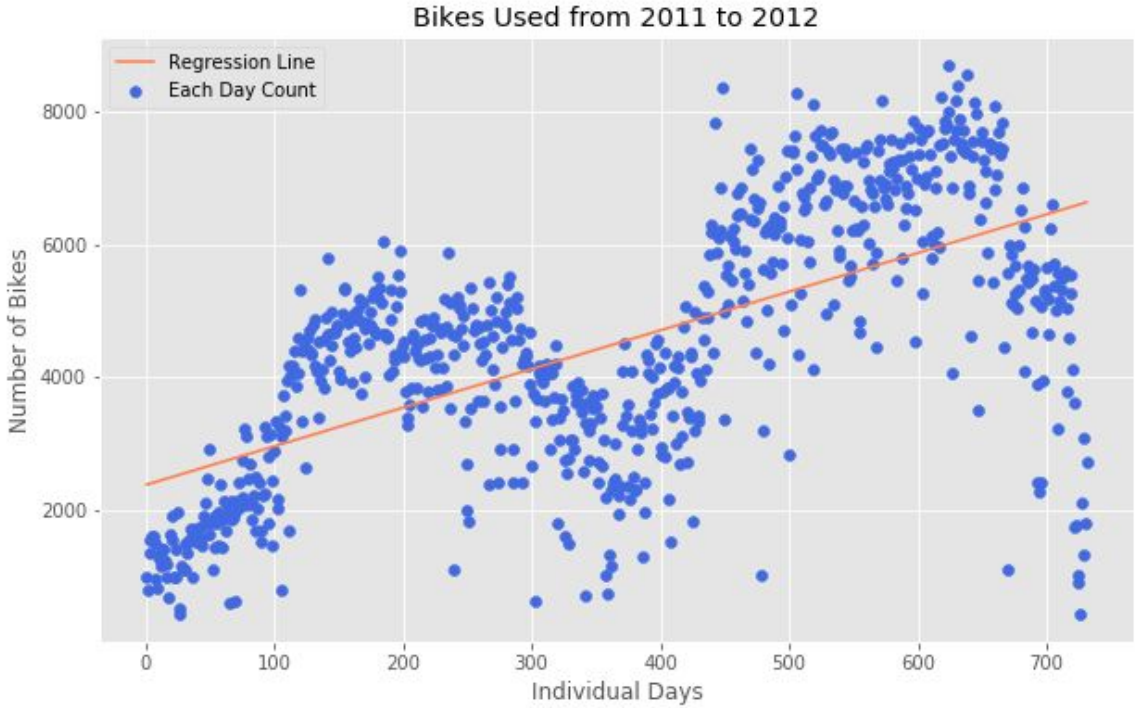
Bikeshare Usage



Normalizing the Trend

- To reduce the effect of the upward trend from one year to the next, the data was normalized using a simple linear regression line.

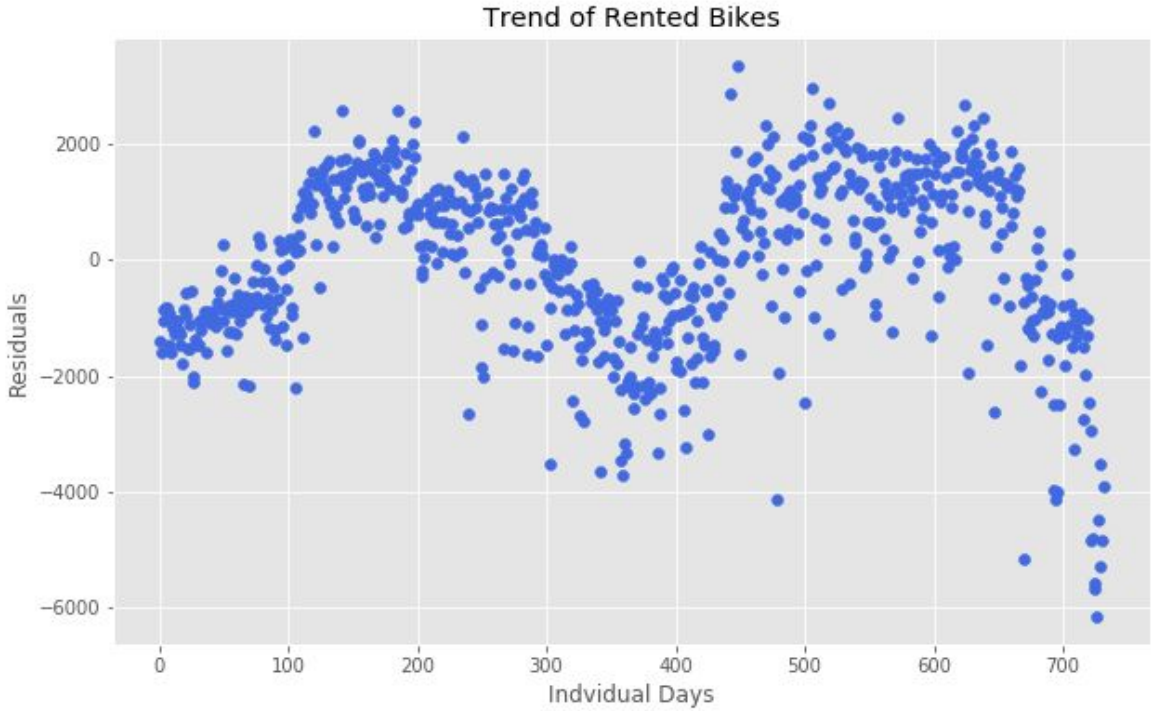
Regression Line for Bikeshare Usage



The Residuals

- Differences between the predicted values on the regression line and the actual values were graphed to normalize the rising trend.

Bikeshare Usage Normalized



Interactions

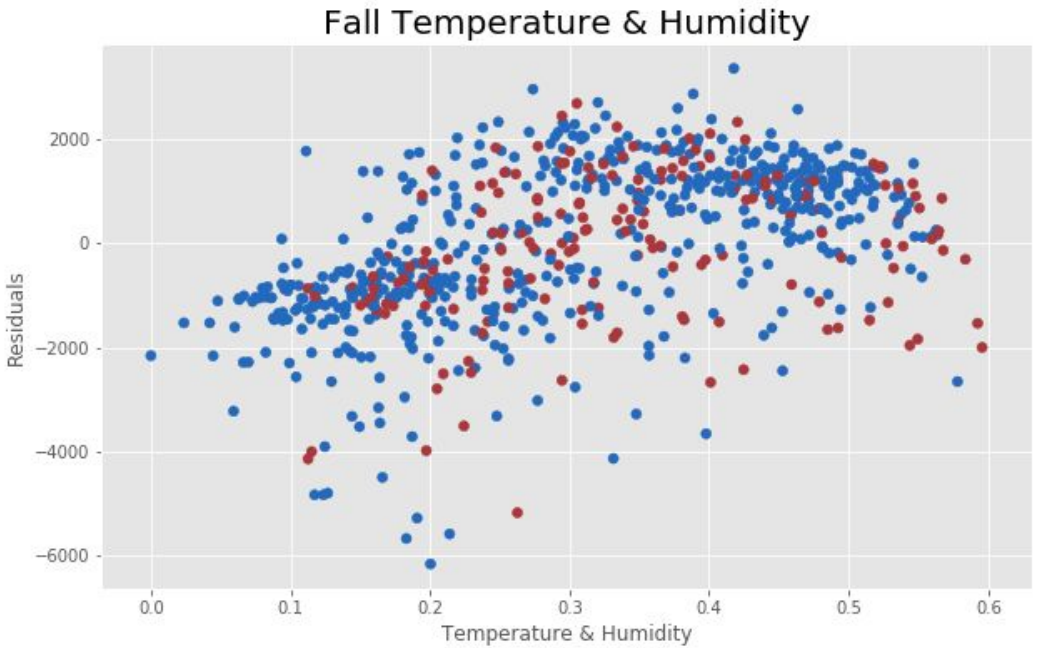
Preview of some of the selected features interacting with one another

Interacted Features

- Temperature and humidity in fall
  - Very large coefficient
  - The only combo of temperature, humidity, season interaction term that was significant
- Humidity & misty weather
  - Has negative coefficient
  - Humidity alone and misty weather alone have positive coefficients
- Humidity on a clear day in summer
  - Humidity in summer alone was not significant
  - Clear day in summer has large positive effect, but add on humidity and bike usage takes a dive
- Temperature on a misty working day
  - Checking how working day might interact with other variables

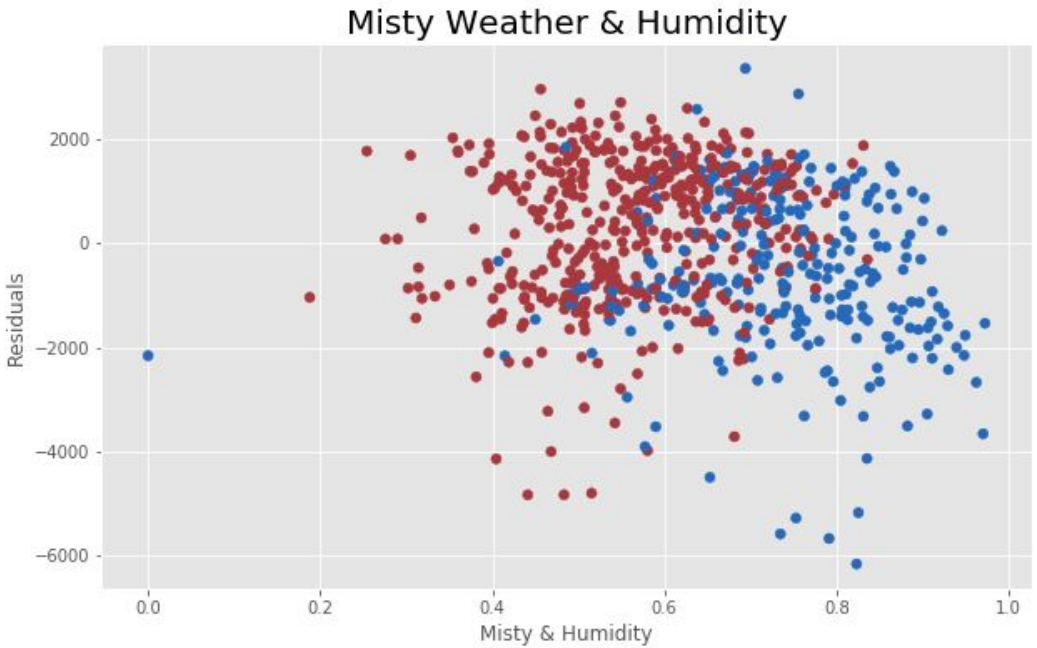
Interactions

- Season: Fall (in red)
- Temperature
- Humidity



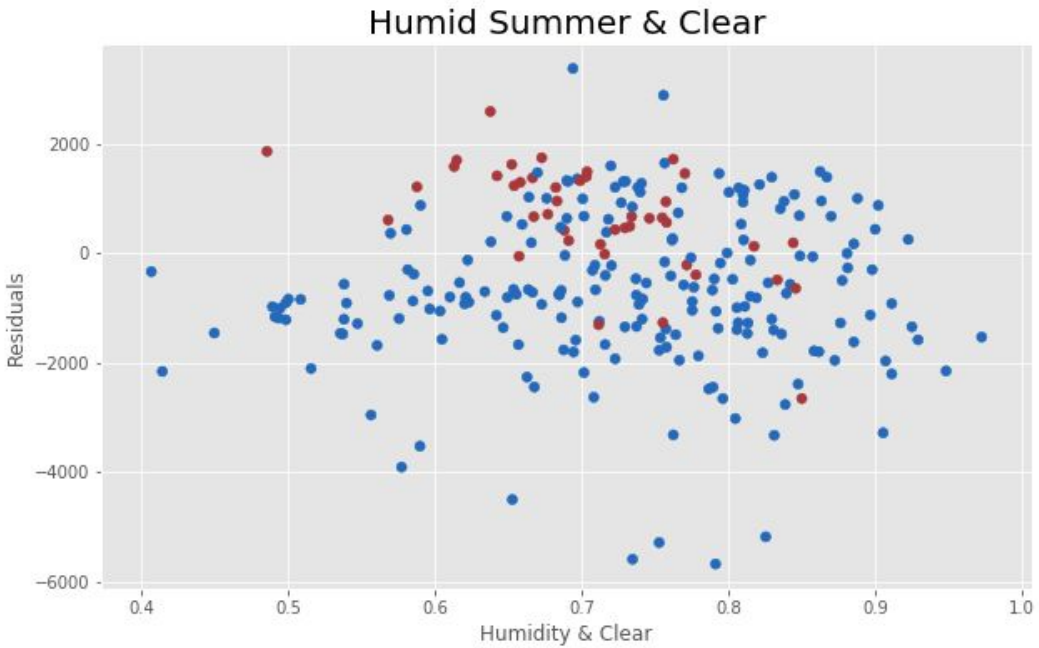
Interactions

- Weather: Misty (in red)
- Humidity



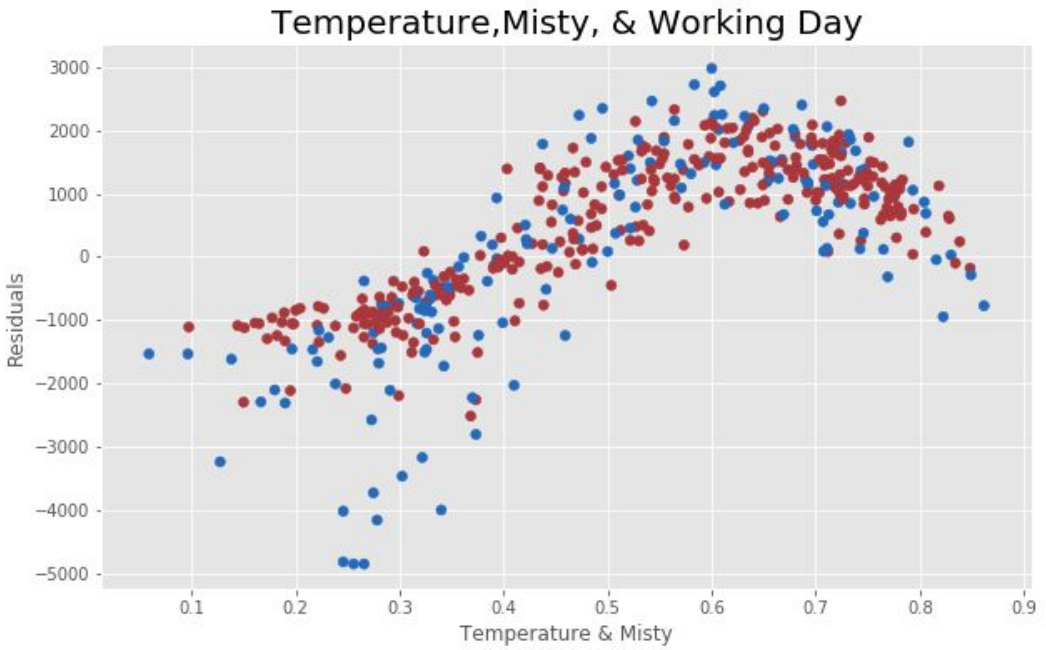
Interactions

- Season: summer (in red)
- Weather: Clear
- Humidity



Interactions

- Weather: Misty
- Working Day (in red)
- Temperature



Final Model

Interacted almost all the variables that were available

- All possible interactions between:
  - Temperature (normalized in Celsius)
  - Humidity (normalized out of 100)
  - Seasons (dummies for summer, fall, winter, spring)
  - Workingday (0 and 1)
  - weather type (dummies for clear, misty, and light storm)

Omnibus:	147.976	**Durbin-Watson:**	1.068
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	399.636
Skew:	-1.020	**Prob(JB):**	1.66e-87
Kurtosis:	5.996	**Cond. No.**	1.27e+16

OLS Regression Results

Dep. Variable:	resids	**R-squared:**	0.738
Model:	OLS	**Adj. R-squared:**	0.723
Method:	Least Squares	**F-statistic:**	49.84
Date:	Fri, 23 Aug 2019	**Prob (F-statistic):**	3.93e-173
Time:	10:39:12	**Log-Likelihood:**	-5879.9
No. Observations:	730	**AIC:**	1.184e+04
Df Residuals:	690	**BIC:**	1.202e+04
Df Model:	39		
Covariance Type:	nonrobust		

Model Preview

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-8148.7443	1199.452	-6.794	0.000	-1.05e+04	-5793.731
clear	6429.2105	1249.895	5.144	0.000	3975.156	8883.265
hum	6852.2732	1572.657	4.357	0.000	3764.505	9940.041
misty	6641.6683	1201.819	5.526	0.000	4282.007	9001.330
summer	6229.1393	1084.862	5.742	0.000	4099.112	8359.166
clear_fall	-7531.8129	1816.557	-4.146	0.000	-1.11e+04	-3965.170
clear_holiday	1.269e+04	3276.052	3.872	0.000	6253.200	1.91e+04

Process of Optimizing the Model

- Started with many features interacting with one another.
- Slowly removed features with high P-Values until only statistically significant features remained.

Predicting Values with the Model

- Model was optimized with residual errors as the dependant value.
- Predicted values were the residual errors of the simple linear regression line.

Converting Predicted Errors for Implementation

- Final model predicts residual errors of the simple linear regression line.
- To improve the prediction of the regression line, add the predicted errors to the regression line.
- **(Linear-Regression-Predictions) + (Final-Model-Error-Predictions)**

Final Model Prediction Comparison