Report XAI

franc.petr1234

May 2025

1 Introduction

The Dataset in question contains the data about daily count of rental bikes between years 2011 and 2012 in Capital bikeshare system. The attributes that we will analyze are:

- Workingday/Holiday/Other
- Season
- Weather Situation (Clear/Misty/Rainy)
- Temperature
- Humidity
- Windspeed
- Date (days since the first day in the dataset)

The target variable is the number of bikes rented on a given day.

2 Data Preprocessing

To train our linear model we need to first preprocess the data.

Temperature, Humidity and Windspeed are given in the normalized form - we needed to reverse that normalization.

- Temperature = (Normalised Temperature * 47) - 8
- Humidity = Normalised Humidity * 100
- Windspeed = Normalised Windspeed * 67

For 'season' and 'weathersit' we need to use one hot encoding, since the values don't respresent any numerical relationship, but distinct categories.

This will yield us 3 categories for season - spring, summer, fall (since one class is the baseline class, in this case - winter), and 2 categories for weather - rainy and misty (with baseline class being clear weather)

For calculating 'days_since_2011' we substract 01-01-2011 from the given date to get which day in the sequence it is.

3 Interpreting the Data

3.1 Linear Model Coefficients

Feature	**Estimate**	**Std. Error**
(Intercept)	2399.442211	238.306592
fall	425.602853	110.819879
spring	899.318156	122.283253
summer	138.215432	161.703690
workingday	124.920938	73.266572
holiday	-686.115442	203.301472
misty	-379.398530	87.553162
rain	-1901.539915	223.639973
temp	110.709582	7.043267
hum	-17.377199	3.169416
windspeed	-42.513472	6.891706
days_since_2011	4.926432	0.172816

Table 1: Linear model feature estimates and standard errors

Linear models use the sum of intercept and values from the data multiplied by their corresponding coefficients to return the final prediction. This type of model is easily understandable by humans - given the coefficients we can understand which features are important to the output and which aren't, which is different to other types of models, for example neural networks. Using the standard error gives us insight into which features are consistent and which aren't - for example, wokingday is not a good predictor because of it's high standard error (73) compared to its coefficient (125), and temperature is a good predictor because it has low standard error (7) compared to its coefficient (110). The high standard error in comparison to the estimate means that the feature is not highly correlated with the outcome and that there is little meaningful connection between the two variables, or that the relationship between the two variables is non-linear.

The coefficient also gives us a rough approximation on how an event may impact the amount of bikes rented on a given day. For example:

- Given an increase of 1 degree, we can expect about 110 more bikes to be rented
- Given an increase in wind of 10km/h, we can expect around 420 less bikes to be rented on average
- If both of those changes were applied at the same time, we have to add them both, equaling around 310 less bikes rented on the given day. This additive property makes linear models straightforward to interpret but assumes no interaction between features.

3.2 Weight Plot

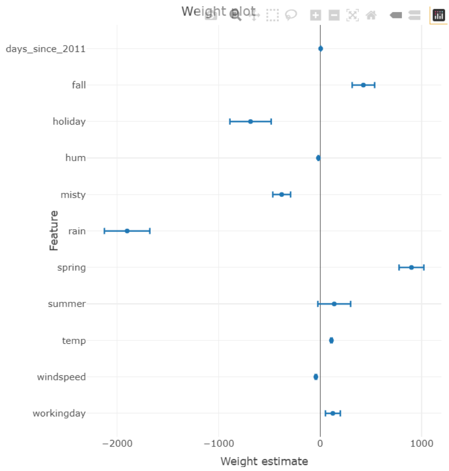


Figure 1: Weight Plot

We can show the coefficients with the standard error on the plot. The unscaled weight plot is visually clear, but misleading in terms of feature importance due to differing input scales. The main strength of graphs is the clarity of data, but in this form it doesn't really tell the whole story - the weights matter in the context of the input numerical values. For example, the change of 5 degrees in temperature is different to a change of 5% in humidity. To see which features contribute the most to the prediction, it's far more useful to show a graph containing normalised data.

The following graph contains a model trained on data scaled to zero mean and unit standard deviation.

This graph is far more interesting for our purposes - it tells us how important are the variables in relation to each other. For example, we can notice that temperature weight estimate is close to 1000, while humidity is around -200. This gives us an empirical idea of how important temperature is, without needing to compare the units. We can just say that whether it's hot is more important than whether the humidity is high.

3.3 Effect Plot

Another way of plotting the data is by using an effect plot. Effect plot uses the effect metric:

$$\text{metric} = \text{weight} * \text{value}$$

Effect plot is useful when trying to learn how changing the variable changes the outcome, if the rest of the variables remain unchanged. We will calculate the effect and create the plot.

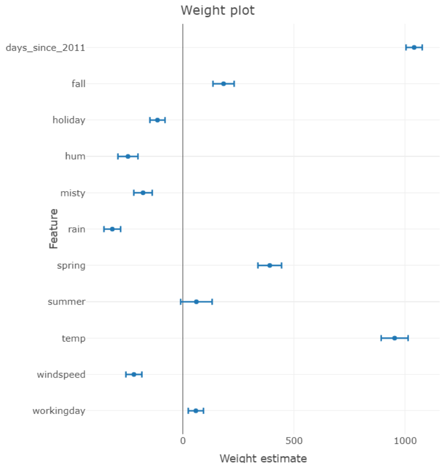


Figure 2: Weight Plot on scaled data

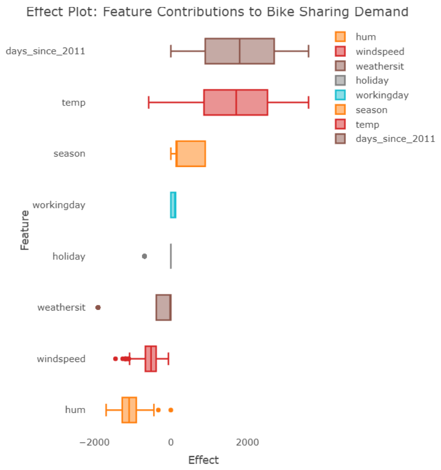
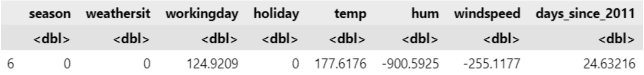


Figure 3: Effect plot

The plot shows that the metrics that contribute the most to the amount of bike rentals are temperature and days_since_2011. Temperature consistently shows a positive effect across observations, with most data points contributing between 800-2500 additional rentals. Feature days_since_2011 displays a steady positive trend, with effects ranging from 900 to 2700. It also shows which metrics are the most volatile - for wind there are a lot of observations that have a very different impact on the prediction than the range predicted by the model. This means that temperature has a consistently positive impact, while impact of the wind is more varied across different days. This variability suggests wind speed has a nonlinear relationship with the number of bikes rented — minor breezes have minimal impact, but strong winds significantly decrease number of bikes rented.

3.4 Analyzing the individual contributions

Now let's calculate how the model calculates its predictions. We'll inspect the 6th instance of the dataset.



	season	weathersit	workingday	holiday	temp	hum	windspeed	days_since_2011
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
6	0	0	124.9209	0	177.6176	-900.5925	-255.1177	24.63216

Figure 4: Effects of 6th instance

In order to get the prediction of the model we sum the effects:

$$124.92 + 177.6 - 900.6 - 255.1 + 24.6 + 2399.44 = 1571$$

The prediction of the model is equal to 1571, while the real value is 1606. We can see the values of the features on the boxplot.

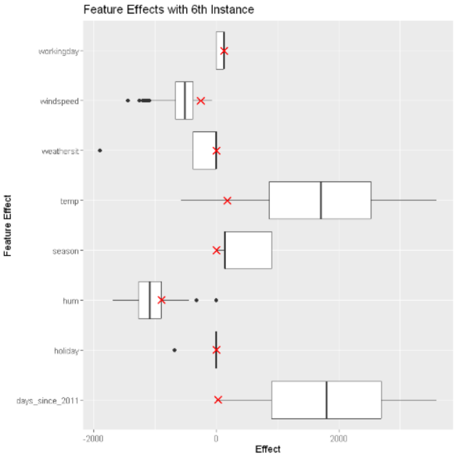


Figure 5: Plot of Coefficients and Standard Error

From the boxplot (and numerical data) we see that the main contributors to the prediction are humidity and windspeed.

Using similar calculations we can find out the average prediction of the model - it turns out that on average the model predicts that 4504 bikes are rented every day. This baseline helps contextualize individual predictions — the 6th instance's prediction of 1571 bikes is significantly below average, likely due to the substantial negative effects from high humidity (-900.6) and windspeed (-255.1) observed in this particular case.

4 Takeaways

Interpreting the linear model trained using the Bike Sharing Dataset is much easier than other forms of models. The model uses math and statistics to predict its outputs, which is easily understandable by humans. We can learn from the linear correlations that the model finds in the data to make our own predictions.

In the given dataset, the two variables had the biggest effect on the final outcome - the temperature and days_since_2011. In the extremes we also see humidity, windspeed and weather situation having huge negative impact on the number of bikes rented - which makes sense, given that wind speed is not something that most people think about when going cycling, unless being unbereably high.