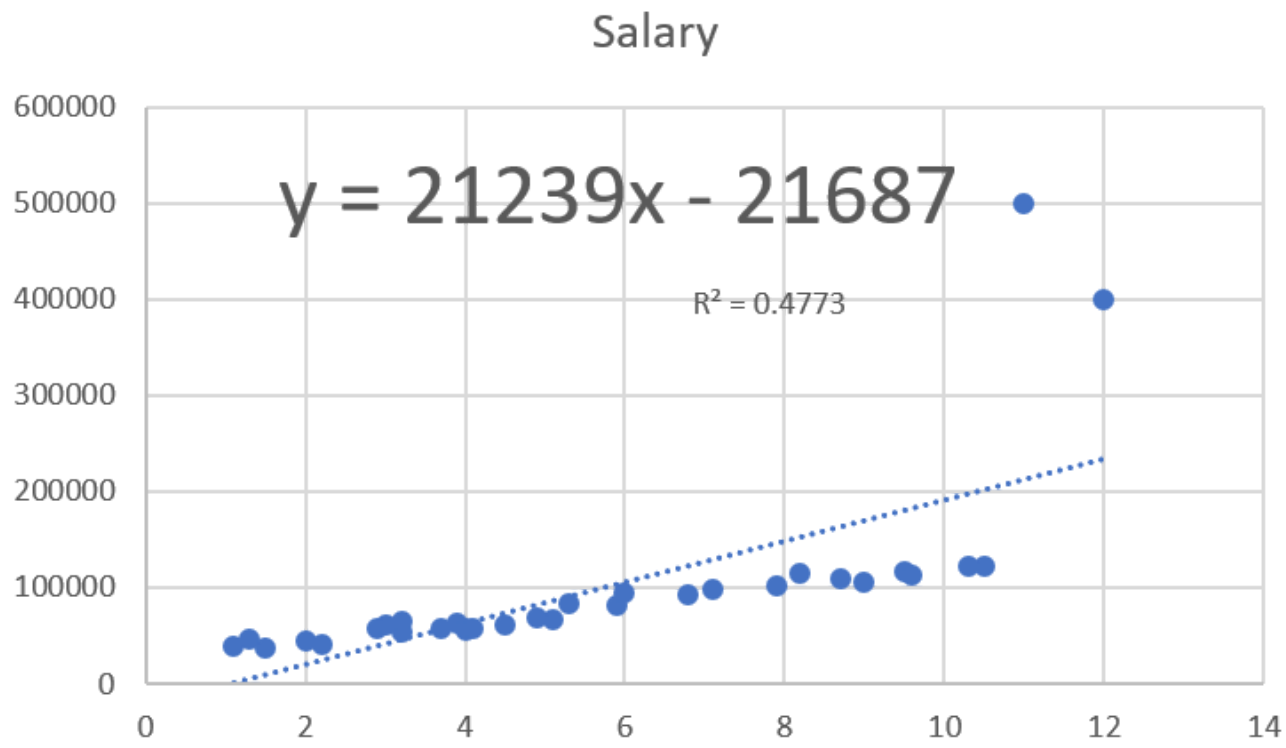
Prashant Nair

Follow 36 Followers About

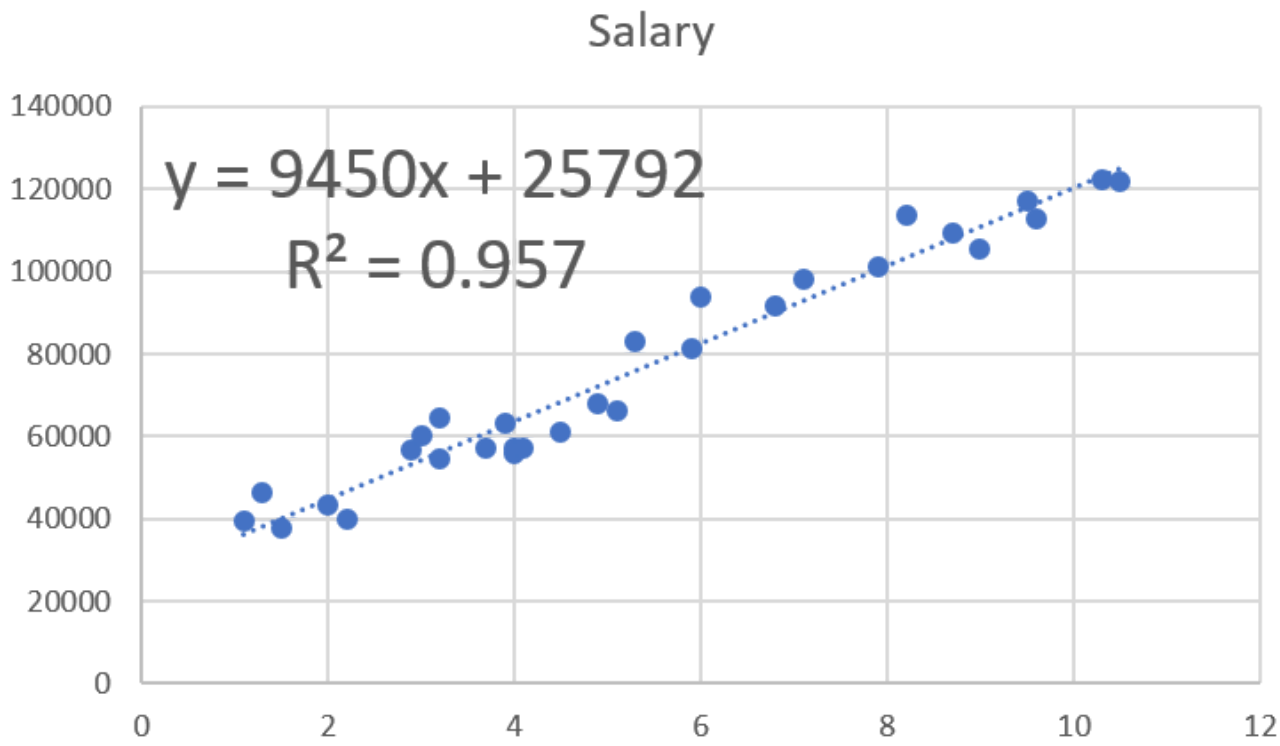
Hands-on : Outlier Detection and Treatment in Python Using 1.5 IQR rule

Prashant Nair Jun 11, 2019 · 3 min read

One of the biggest downfall for any model performance is the outliers present in the data. Outliers ideally are the extreme values for the specific column which affects the generalization of the data and model. Outliers mostly affect the regression models as it changes the equation drastically as shown in the below scatter plot,



You can very well observe here that just two points affected the Linear Regression model (R^2 score is just 47%)



However when the outlier is removed, you see the performance of the model is improved drastically from 48% to 95%. Isn't this awesome !

The intention of Outlier detection and treatment is to ensure you get the best model out of the data considering the fact that your data is qualified to work with the algorithm. In this case, the data is linear and is compatible with the Linear Regression Algorithm. So lets see how to detect and remove outliers from your data inPython using 1.5 IQR rule.

IQR stands for Inter-Quartile Range. Lets see the wikipedia definition of IQR.

In descriptive statistics, the interquartile range, also called the midspread or middle 50%, or technically H-spread, is a measure of statistical dispersion, being equal to the difference between 75th and 25th percentiles, or between upper and lower quartiles, $IQR = Q_3 - Q_1$.

Source: Wikipedia

Steps to perform Outlier Detection by identifying the lowerbound and upperbound of the data:

1. Arrange your data in ascending order
2. Calculate Q1 (the first Quarter)
3. Calculate Q3 (the third Quartile)
4. Find IQR = (Q3 - Q1)
5. Find the lower Range = Q1 -(1.5 * IQR)
6. Find the upper Range = Q3 + (1.5 * IQR)

Once you get the upperbound and lowerbound, all you have to do is to delete any values which is less than lowerbound or greater than upperbound.

Now lets see the code for the same:

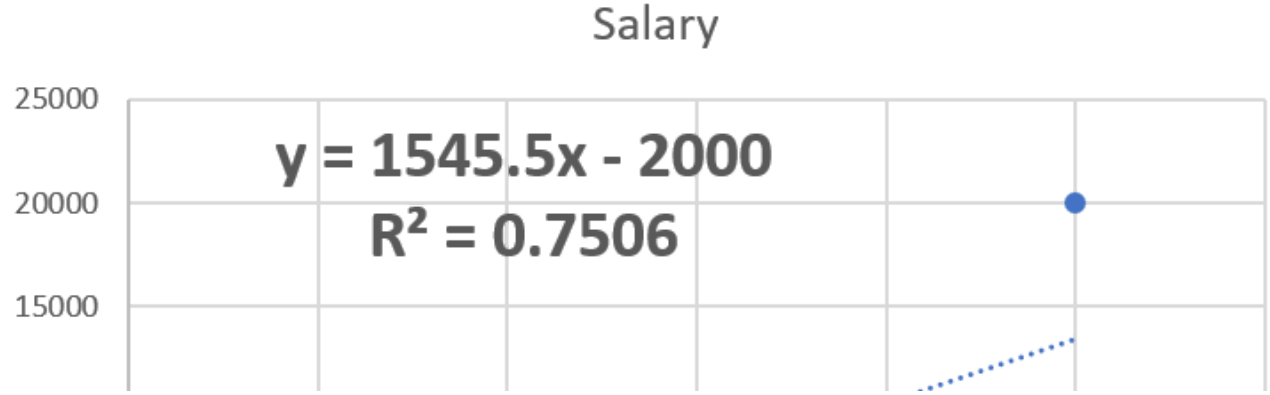
1. Import necessary packages

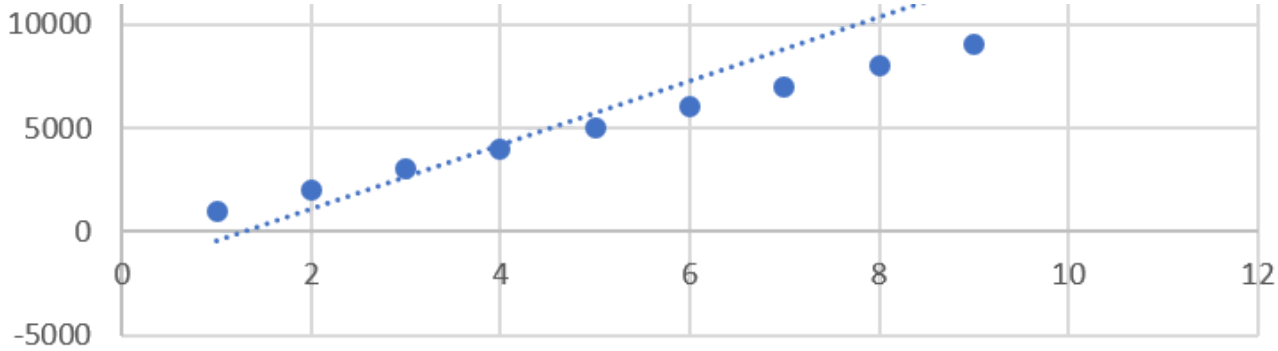
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

2. Create a sample dataset with outliers

```
sample = pd.DataFrame([[1000],[2000],[3000],[4000],[5000],[6000],
[7000],[8000],[9000],[20000]]
 , columns=['Salary'])
```

As you see the dataset I did have added an extreme value i.e. 20000. Lets check the scatter plot to see the outlier,





In this dataset, 20000 is the extreme value. Lets check whether the 1.5IQR rule helps us !

3. Lets write the outlier function that will return us the lowerbound and upperbound values.

```
def outlier_treatment(datacolumn):
 sorted(datacolumn)
 Q1,Q3 = np.percentile(datacolumn , [25,75])
 IQR = Q3 — Q1
 lower_range = Q1 — (1.5 * IQR)
 upper_range = Q3 + (1.5 * IQR)
 return lower_range,upper_range
```

4. Using the above function, lets get the lowerbound and upperbound values

```
lowerbound,upperbound = outlier_treatment(sample.Salary)
```

5. Lets check which column is considered as an outlier

```
sample[(sample.Salary < lower_range) | (sample.Salary >
upper_range)]
```

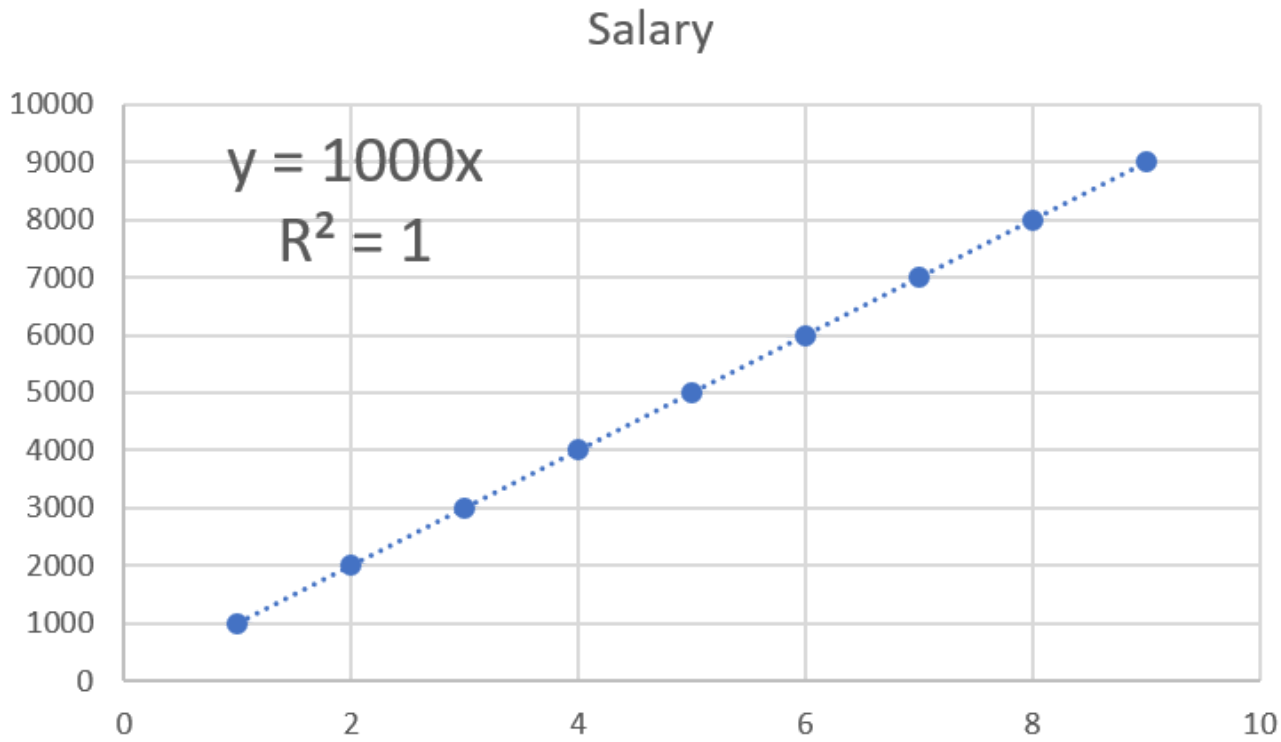
The above code gives the following output,

	Salary
9	20000

5. Now lets remove the outliers from the dataset.

```
sample.drop(sample[ (sample.Salary > u) | (sample.Salary < l)
].index , inplace=True)
```

6. Lets see the scatter plot after outlier removal



As you can observe, after outlier is removed, the data is now well performing with Linear Regression.

Hope this quick tutorial helps. Want to download the jupyter notebook, check out my github link @ https://github.com/aituts/mediumArticles/blob/master/OutlierDetectionAndTreatmentExample.ipynb

Cheers !!!

Data Science Machine Learning Python Outliers Outlier Detection