Introduction to Data Science

Modeling Nonlinear Relationships

Joanna Bieri DATA101

Important Information

- Email: joanna_bieri@redlands.edu
- Office Hours: Duke 209 Click Here for Joanna's Schedule

Announcements

Come to Lab! If you need help we are here to help!

Day 16 Assignment - same drill.

1. Make sure you can **Fork** and **Clone** the Day16 repo from Redlands-DATA101
2. Open the file Day16-HW.ipynb and start doing the problems.
   - You can do these problems as you follow along with the lecture notes and video.
3. Get as far as you can before class.
4. Submit what you have so far **Commit** and **Push** to Git.
5. Take the daily check in quiz on **Canvas**.
6. Come to class with lots of questions!

Paris Paintings Data

To explore the ideas of modeling data we will use the Paris Paintings dataset.

- Source: Printed catalogs of 28 auction sales in Paris, 1764 - 1780 (Historical Data)
- Data curators Sandra van Ginhoven and Hilary Coe Cronheim (who were PhD students in the Duke Art, Law, and Markets Initiative at the time of putting together this dataset) translated and tabulated the catalogs
- 3393 paintings, their prices, and descriptive details from sales catalogs over 60 variables

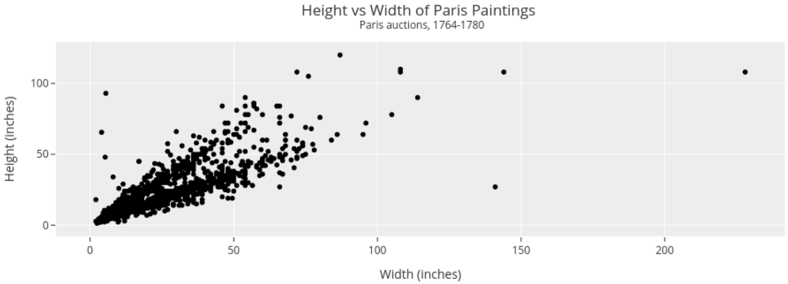
Variables in Paris Paintings Data

Paris Paintings Data

	name	sale	lot	position	dealer	year	origin_author
0	L1764-2	L1764	2	0.032787	L	1764	F
1	L1764-3	L1764	3	0.049180	L	1764	I
2	L1764-4	L1764	4	0.065574	L	1764	X
3	L1764-5a	L1764	5	0.081967	L	1764	F
4	L1764-5b	L1764	5	0.081967	L	1764	F
...	...	...	...	...	...	...	...
3388	R1764-498	R1764	498	0.992032	R	1764	F
3389	R1764-499	R1764	499	0.994024	R	1764	F
3390	R1764-500	R1764	500	0.996016	R	1764	F
3391	R1764-502a	R1764	502	1.000000	R	1764	F
3392	R1764-502b	R1764	502	1.000000	R	1764	F

Paris Paintings Data - Plot Height vs Width

```
Unable to display output for mime type(s): text/html
```



Paris Paintings Data - Linear Regression

```
X = DF_model['Width_in'].values.reshape(-1, 1)
y = DF_model['Height_in'].values

LM = LinearRegression()
LM.fit(X, y)
```

```
LinearRegression()
```

```
Coefficient:
[0.78079641]
Intercept:
3.6214055418381896
Score:
0.6829467672722757
```

Testing for Linearity

How do we know if this is good?

We can always look at the score, R^2, but this only tells us about the average distance away from the prediction each of our data points is. What could cause the metric to be low?

- Data having a high amount of scatter
- Data not actually being linear

How do we tell the difference?

$$Residual = DataValue - PredictedValue$$

Add the prediction and the residual to our data frame!

```
LM.predict(X) = LM.intercept_ + LM.coef_*DF['Width_in']
```

Testing for Linearity

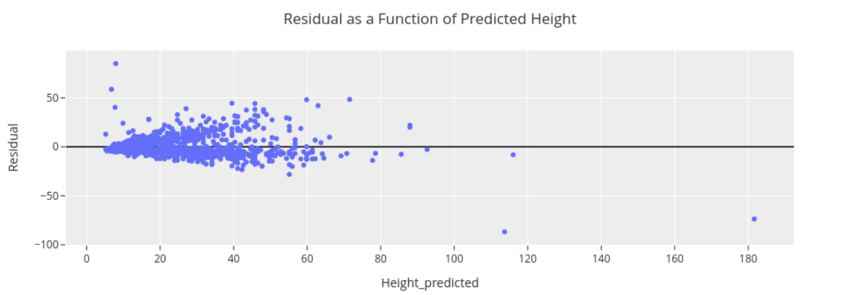
Plot the residual

Now we can plot the residual - this gives us information about whether or not the linear model was appropriate, even in there is a lot of scatter in our data.

- Do a scatter plot of the Residual vs. the Predicted Value (Height).
- Add a line at y=0, to make the residual plot easier to interpret.

Testing for Linearity

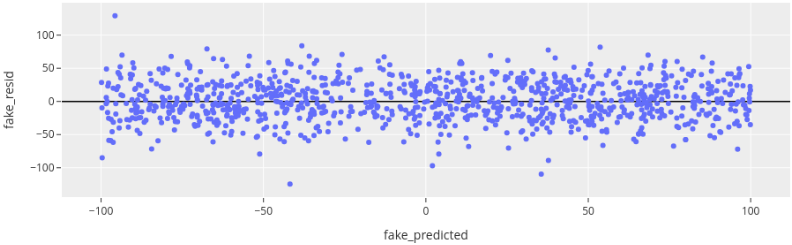
Plot the residual



Interpreting Residual Plots

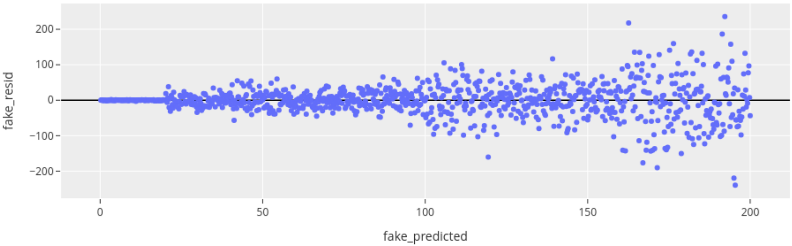
What we are looking for

- Residuals distributed randomly around 0
- With no visible pattern along the x or y axes



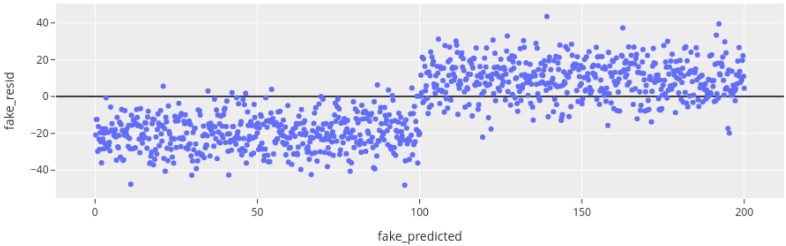
What we don't want

Fan shapes



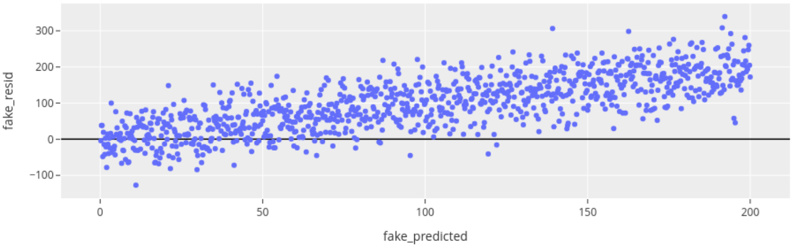
What we don't want

Groups of patterns



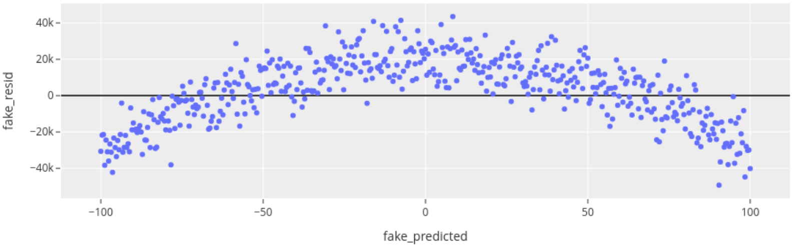
What we don't want

Residuals correlated with predicted values



What we don't want

Any patterns!

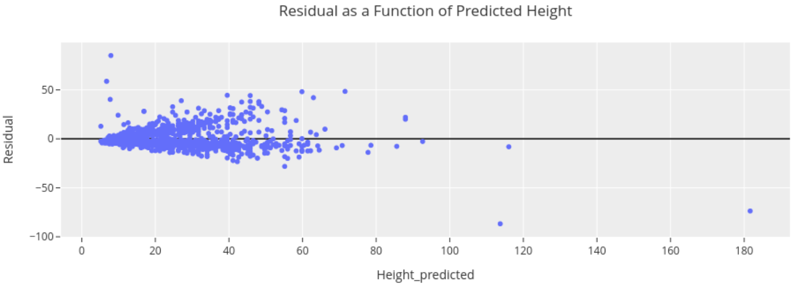


What does the residual plot tell us?

What patterns does the residuals plot reveal that should make us question whether a linear model is a good fit for modeling the relationship between height and width of paintings?

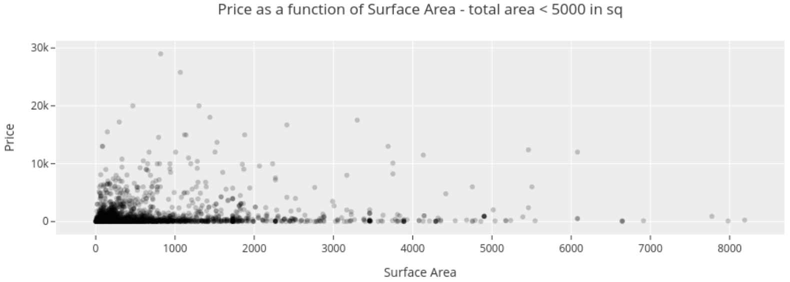
- We don't want to see patterns whatsoever.
- All interesting relationships should have been captured by the linear model
- Any pattern remaining means that the linear model is maybe not the best fit - there is still something going on here.

Look at the residual from Height vs Width



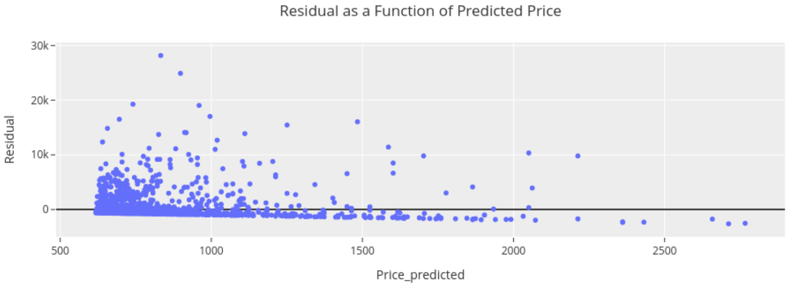
How do we explore linearity?

Model for Price as a function of size with area of less than 10,000 inches squared.



`LinearRegression()`

How do we explore linearity?



What do we do if our data is not linear?

Sometimes we can apply a transformation to our response variable (in this case price) that will help us “unpack” the nonlinearity. Lets look at a histogram of the prices.

```
Unable to display output for mime type(s): application/vnd.plo
```

What do we do if our data is not linear?

The price data is very **skewed**, this means that most of the data is to one side of the histogram. In other words, most paintings sold for less than 5000 (money = livres). Here we see extremely skewed data like this, it looks like a decaying exponential function, so this might influence us to apply a natural log function to the price data!

Use the log()

Remember:

$$\log(e^x) = x$$

The natural log removes the exponential dependence. For us all of these things are the same:

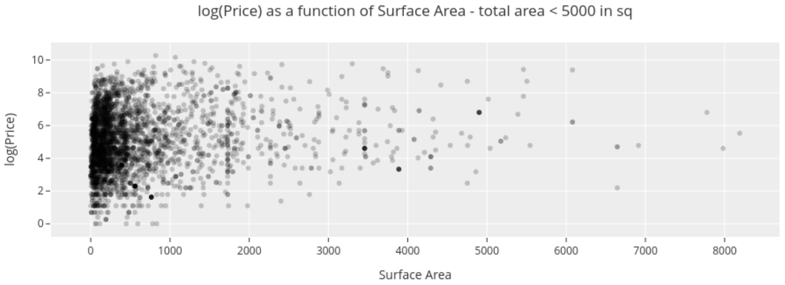
$$\log(x) = \ln(x) = \log_e(x)$$

Below we will apply the natural log to the price column and store that data in a new column in our data frame:

```
DF_model2['log_price'] = np.log(DF_model2['price'])
```

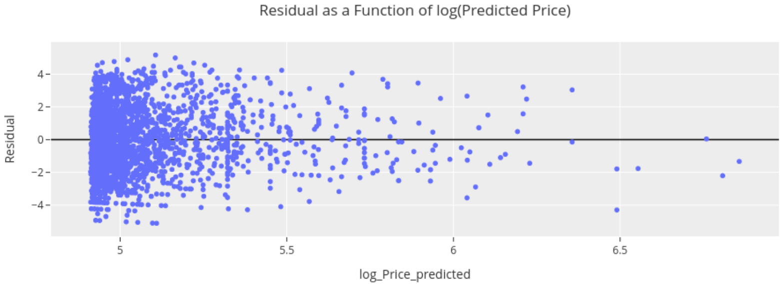
Use the log() - Histogram of Log Prices

Scatter Plot of Log Prices

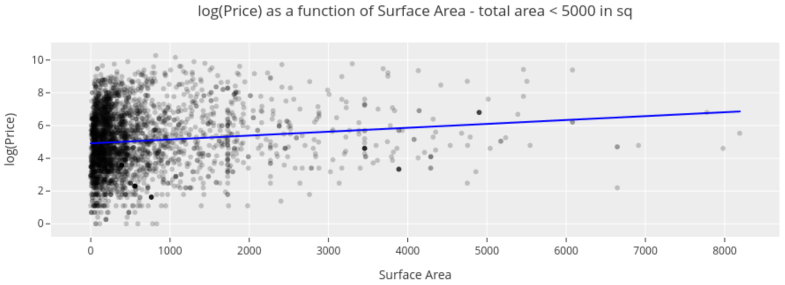


Redo the Analysis for log_price vs size

`LinearRegression()`



Redo the Analysis for log_price vs size




```
Coefficient:
[0.0002376]
Intercept:
4.911544880415433
```

Redo the Analysis for log_price vs size

What is the model telling me?

$$\log(\widehat{price}) = 4.912 + 0.00024(SurfaceArea)$$

How can we interpret this result so it actually makes sense in the context of our problem?

Redo the Analysis for log_price vs size

Properties of exponents and logs:

$$e^{log(x)} = x$$

$$\log(a) - \log(b) = log(a/b)$$

Redo the Analysis for log_price vs size

If our surface area increase by 1 inch squared how much should our price increase? Lets look at the difference in log prices if we increase by an inch squared.

Plugging into our formula:

$$log(SA+1) - log(SA) = [4.898 + 0.00024(SA+1)] - [4.898 + 0.00024(SA)]$$

doing algebra on the right hand side

Redo the Analysis for log_price vs size

$$log(SA + 1) - log(SA) = 0.00024$$

using the log subtraction rule

Redo the Analysis for log_price vs size

$$log\left(\frac{SA+1}{SA}\right) = 0.00024$$

using the exponent undoing the log rule

Redo the Analysis for log_price vs size

$$\frac{SA+1}{SA} = e^{0.00024}$$

calculating the exponent on the right hand side

Redo the Analysis for log_price vs size

$$\frac{SA+1}{SA} \sim 1.00024002880230 41$$

solving for $SA+1$

Redo the Analysis for log_price vs size

$$(SA+1) \sim 1.00024002880230041 * SA$$

So this tells us that increase the area of the painting by one square inch increases the price by a factor of 1.0002400288023041 or about 0.024%.

What did we learn...

There is a small positive increase in the price as the surface area increases, on average. Can we predict the price using the surface area?

`Result of LM.score(X,y):`

`0.013268243020669424`

It does not appear that our logistic regression is a good predictor of the price. Even though it looks like we captured a good linear relationship, we do not have a good predictor. The scatter is still very large!

BUT - we are still able to see a linear trend in the model. There is a relationship here even though the data is very noisy!

Recap

- Non-constant variance is one of the most common model violations, however it is usually fixable by transforming the response (y) variable.
- The most common transformation when the response variable is right skewed is the log transform: $log(y)$, especially useful when the response variable is (extremely) right skewed.
- This transformation is also useful for variance stabilization.
- When using a log transformation on the response variable the interpretation of the slope changes:

"For each unit increase in x, y is expected on average to be higher/lower by a factor of e^{b_1}."

- Another useful transformation is the square root: $\sqrt{y}$, especially useful when the response variable is counts.