

Prepared by:
Server.py

REGRESSION REPORT

DEC 2021



Regression Report

User Defined Parameters

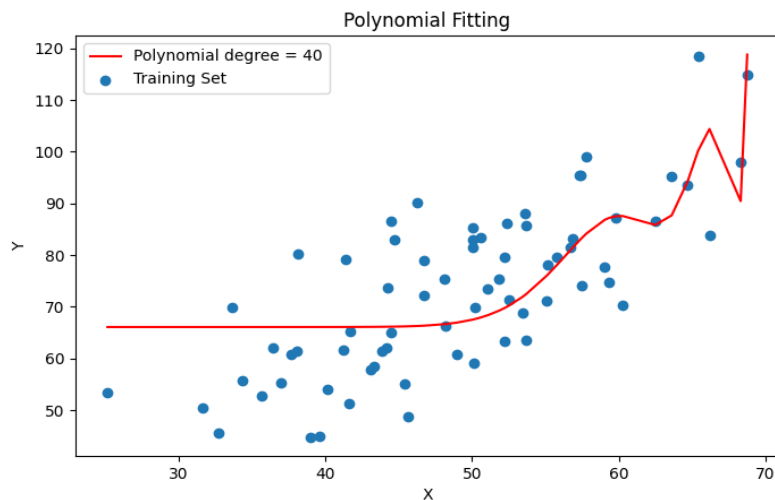
Dataset: received.txt
Polynomial Degree = 40
Train-test-split = 30%

Dataset Exploration

Total Points = 100
Training Points = 30
Testing Points = 70
Max = 70.34607561504933
Min = 25.128484647772304
Mean = 48.95834146155929
Median = 49.63496984153295
Standard Deviation = 9.746379302515331

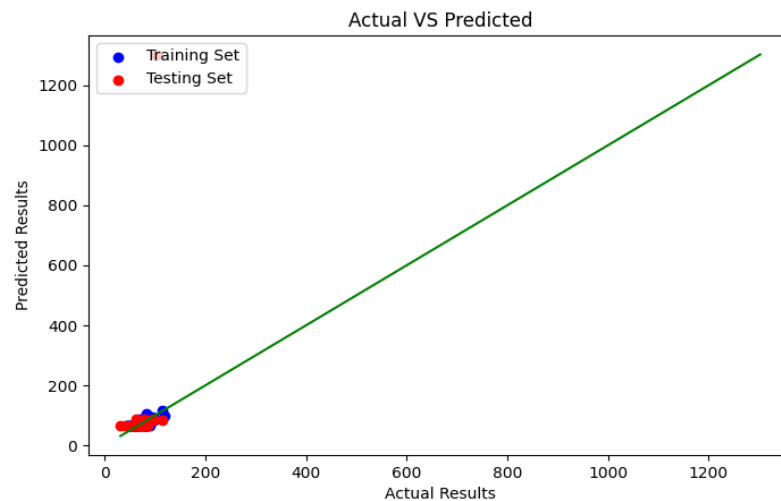
Results

1. Polynomial Fitting



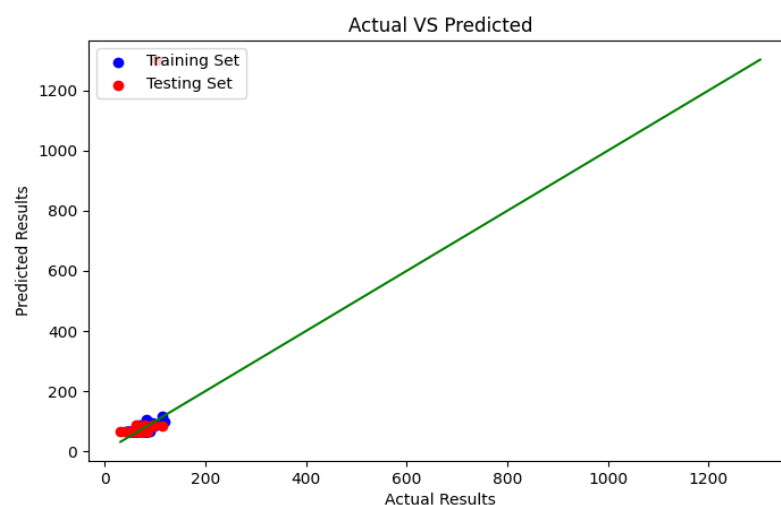
The above figure represents the regression line being fitted to the training instances. If the regression line passes through all the points then the polynomial degree chosen overfits the data. Similarly, if the line does not touch even a single point, you might be underfitting the training instances.

2. Actual vs Predicted



The above scatter plot represents the spread of predicted and actual scores. Assuming our model perfectly fits the data, we can expect the points spread across the $y = x$ line.

3. Residual vs Predicted



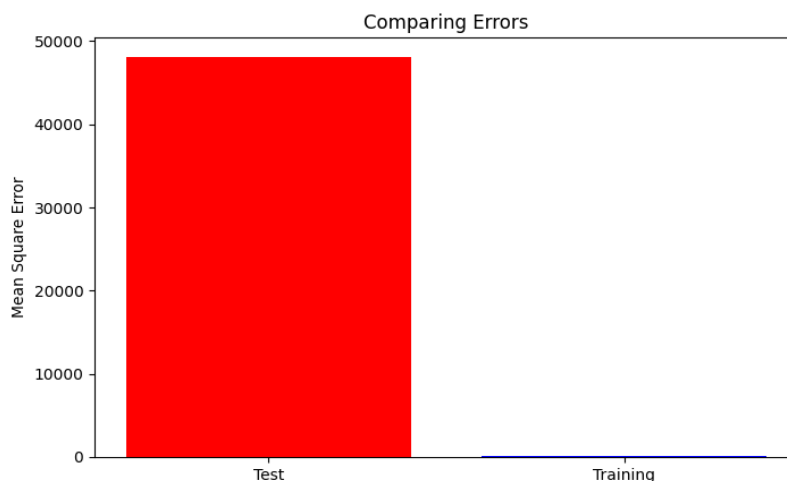
$$\text{Residual} = h(x) - f(x)$$

where,

$h(x)$ is the predicted value and $f(x)$ is the actual value

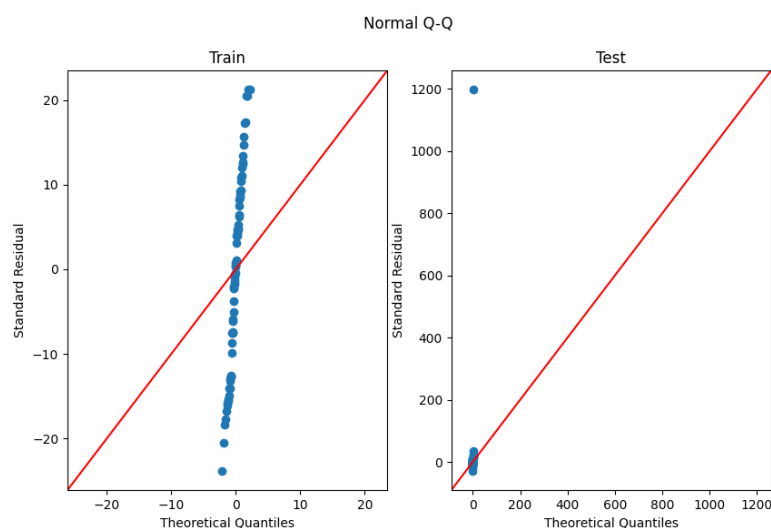
The fitted vs residuals plot is mainly useful for investigating: Whether linearity holds. This is indicated by the mean residual value for every fitted value region being close to 0. In R this is indicated by the red line being close to the dashed line. Whether homoskedasticity holds. The spread of residuals should be approximately the same across the x-axis. Whether there are outliers. This is indicated by some extreme residuals that are far from the rest.

4. Mean Squared Error



The mean squared error is calculated as the sum of differences in actual and predicted result squared. The lower the MSE the better the model performs.

4. Normal Q-Q Plots



The Q-Q plot, or quantile-quantile plot, is a graphical tool to help us assess if a set of data plausibly came from some theoretical distribution such as a Normal or exponential. For example, if we run a statistical analysis that assumes our dependent variable is Normally distributed, we can use a Normal Q-Q plot to check that assumption. It's just a visual check, not an air-tight proof, so it is somewhat subjective. But it allows us to see at-a-glance if our assumption is plausible, and if not, how the assumption is violated and what data points contribute to the violation.

A Q-Q plot is a scatterplot created by plotting two sets of quantiles against one another. If both sets of quantiles came from the same distribution, we should see the points forming a line that's roughly straight. Here's an example of a Normal Q-Q plot when both sets of quantiles truly come from Normal distributions.



Contact

IP Address: 192.168.1.18

Port: 2254

Just connect you socket to the above address.

```
socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

Our customer server.py will get back to you shortly.