Statistical Analysis of Real-World Operations

Alexander Young

November 10, 2025
WMQE-15 ASAI Assessment 2

Question 1: Regression Modeling of Logistics Bill and GDP

(a) Plot of Logistics Bill vs GDP

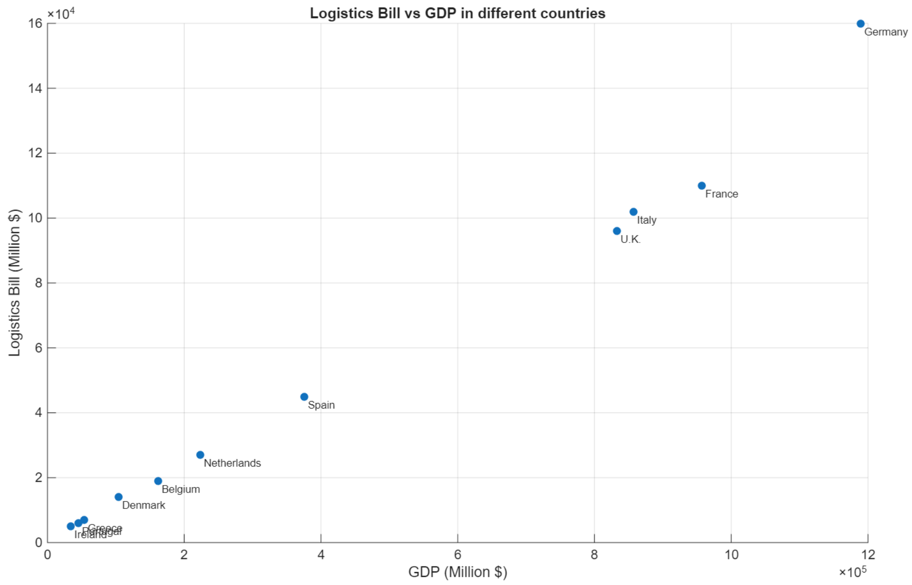

Figure 1: Scatter plot of Logistics Bill vs GDP across 11 countries.

After creating a scatter plot to better visualize the data, it is always a good idea to compute descriptive statistics to understand the spread of the data.

Table 1: Summary Statistics

	Min	Median	Max	Mean	Std
GDP	33 900	223 600	1 189 100	4.3938e+05	4.3215e+05
Logistics_Bill	5 000	27 000	160 000	5.3727e+04	5.3841e+04

It is important to gather insight into the relationship between features before modeling to aid in reliable model selection. It is easy to see in Figure 1 that the data appears to have a strong positive linear relationship, further confirmed in Table 2 by conducting correlation analysis.

Table 2: Correlation Analysis

	Rho	pValue
Pearson	0.9948	3.1666e-10
Spearman	1	0

Pearson correlation measures the linear relationship with Rho ranging from -1 to 1, indicating the strength and direction of the relationship. Having a p-value < 0.05 further indicates that this relationship is statistically significant.

Spearman correlation reinforces the results of the Pearson test and measures the monotonic relationship between features, meaning it is useful for relationships that may not be linear.

(b) Modeling the Relationship between Logistics Bill and GDP

Now that the relationship between Logistics Bill and GDP has been confirmed both visually and through correlation tests, it is time to fit a model. It is best to fit a model of simplest order first to prevent overfitting, since a model that is overfit can be unreliable if used with new data.

The built-in MATLAB function `fitlm` can be used to fit a simple linear regression model in the form $y \sim 1 + x_1$ (*Appendix: Code 1.1, Figure 1.2*).

The model can then be interpreted as:

$$\textbf{Logistics Bill} = 0.12384 \cdot \textbf{GDP} - 727.56$$

where the predicted slope suggests an increase of \$1 million in GDP is associated with about a \$0.124 million increase in Logistics Bill.

Below in Table 3 the values for the slope and y-intercept can be found along with their standard error, test statistic, and p-value. The standard error for the intercept seems large, and the p-value is greater than 0.05, suggesting that the intercept is not statistically different from 0. On the other hand, the slope can be considered statistically significant from the given **p-value = 3.1666e-10** < 0.05, further confirming the dependence of Logistics Bill on GDP.

Table 3: Linear Regression Model ($R^2 = 0.99$)

	Estimate	SE	tStat	pValue
Intercept	-727.56	2557.2	-0.28452	0.78245
x_1	0.12394	0.0042453	29.194	3.1666e-10

Below, Figure 2 shows the fitted regression line over the data, along with the 95% confidence interval of the mean (*green*) and the 95% prediction interval (*red*).

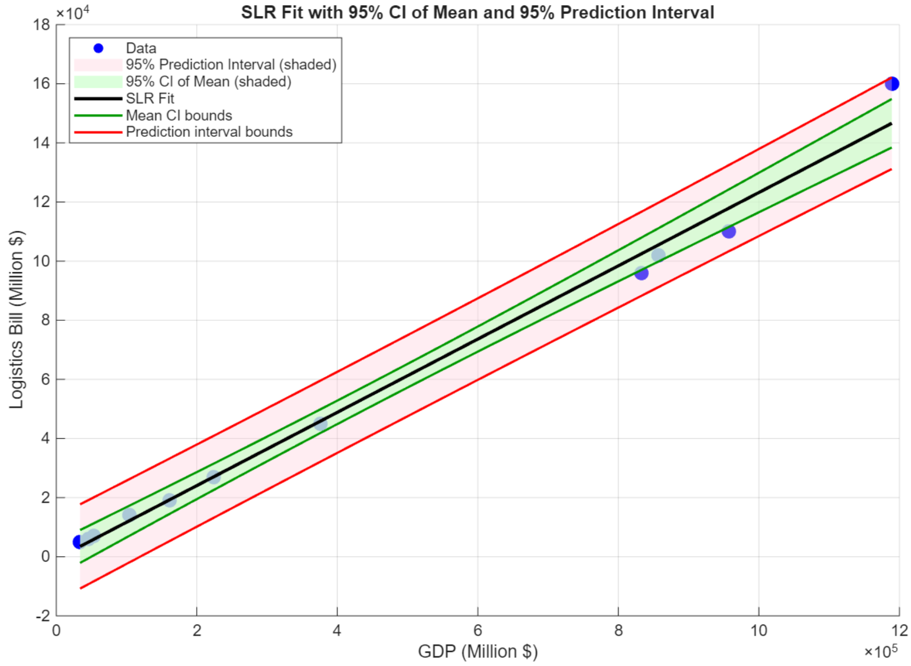

Figure 2: Simple linear regression model with 95% confidence interval for mean response and 95% prediction interval.

(*Appendix: Code 1.2*) There is 95% confidence that the slope and y-intercept of the fitted regression model fall within these ranges:

The 95% confidence interval for **Slope:** [**0.114332, 0.133539**], is positive throughout and reinforces that Logistics Bill increases with GDP, giving a possible range of this increase.

The 95% confidence interval for **Intercept:** [-6512.2430**, 5057.1324**], includes 0 in its possible range; however, the intercept itself is less valuable as a country having 0 GDP is outside the range of realistic observations. This can however speak to the reliability of the model far outside the data it was trained on.

The error metrics of the regression model are below (Table 4):

Table 4: Error Metrics — Linear Regression Model

$R^2 = 0.9896$	RMSE = 5801.4822	MAE = 3444.8698

These metrics indicate that about 99% of the variability in logistics bill is explained by the model and prediction errors are roughly around 3,500–5,800 million $. It is important to mention that RMSE is heavily influenced by outliers, so it will be critical to evaluate outliers using methods such as box plots (see Figure 3).

(c) Logistics Bill Prediction for a Country with GDP of $600 Billion

From the simple linear regression model, a country with a $600 billion GDP is predicted to have a Logistics Bill of **$73,633.50 million**. The 95% confidence interval for mean Logistics Bill at this GDP is: **[69,386.50, 77,880.51] million dollars**,

and the 95% prediction interval for a single new country with this GDP is: **[59,839.56, 87,427.45] million dollars**.

The distinction between these two intervals is important. The **confidence interval** captures uncertainty about where the *true average* Logistics Bill lies for all countries at this GDP level; it reflects only the estimation uncertainty in the regression line. The **prediction interval** captures where a *single new observation* might fall, which adds the natural variability of individual countries on top of the model uncertainty. This is why the prediction interval is substantially wider: it accounts for both where the line sits and how far any one country may deviate from it.

(d) Concerns and Peculiarities as a Data Analyst Advisor

After creating a model and getting predictions, it is also important to conduct a residual analysis. Residuals are the errors in the model, in this case the actual values from the Logistics Bill minus the predicted values, then standardized by applying a Z-Score transformation. By subtracting the mean of the residuals and dividing by their standard deviation, errors are then on a scale which is easy to interpret.

The first visualization that is useful is a box plot that enables the identification of outliers in the prediction errors. It can be seen in Figure 3 that there is one outlier detected, classified as being greater than 75% of the data (Q_3) + 1.5 times the Interquartile Range of the data. This prediction error comes from the model's struggle to predict the Logistics Bill of Germany, which makes sense as Germany had a comparably large GDP and Logistics Bill in relation to the rest of the data. Although removing Germany from the data before training a model should lead to better error metrics, this is usually not the best practice as the model would then be less reliable in predicting values inside the GDP range between France and Germany.

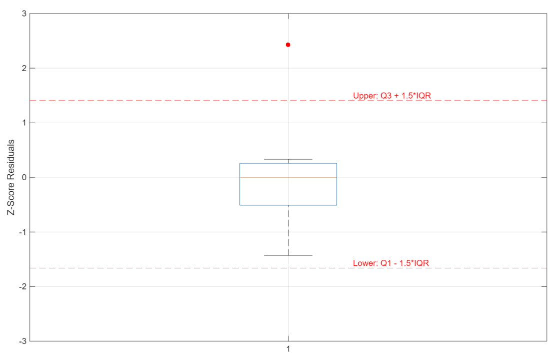

Figure 3: Box plot of residuals from simple linear regression model.

A subsequently important indication of a reliable regression, especially when the sample size of training data is small, is to confirm that residuals from the model are normally and randomly distributed about the mean (in this case 0 as the residuals have been standardized). Visually this is easy to inspect by a QQ-Plot as seen (*Appendix: Figure 1.4*), where the residuals tend to follow the 45-degree line representing a perfectly normal distribution. An additional visual check for confirming the assumption of normality in the residuals is done by a histogram plot as seen in (*Appendix: Figure 1.5*). This histogram is created by classifying the residuals into prospective bins which are optimally determined by Sturges' formula:

$$k = \lceil 1 + 3.322 \cdot \log(n) \rceil$$

The histogram of the residuals should follow the shape of the normal distribution (bell curve), and this is made easier to see by overlaying a perfectly normal curve calculated using MATLAB's built-in `normpdf` function over the length of the bins calculated from Sturges' formula.

Although visually the residuals appear to be normally distributed, it is often useful to utilize a quantitative check. This can be done via a chi-squared hypothesis test (*Appendix: Code 1.3*). At a significance level of $\alpha = 0.05$, the test returned a **p-value = 0.2486**, meaning there is not enough statistical evidence to reject the null hypothesis of the residuals being normally distributed.

Table 5 shows the chi-squared breakdown used for this normality test. Each bin compares the observed frequency of standardized residuals against the expected frequency under a normal distribution. The test statistic is computed as $\chi^2 = \sum (O_i - E_i)^2 / E_i$.

Table 5: Chi-Squared Normality Test Breakdown (SLR Residuals)

Bin Center	Observed	Expected
-1.0436	2	1.9667
-0.27241	4	3.1877
0.49878	4	2.9336
1.27	0	1.5328
2.0412	1	0.45416
$\chi^2 = 2.784$,	df $= 2$,	$p = 0.2486$

The next and arguably more important characteristic for residual analysis of a model is homoscedasticity. Residuals should have constant variance because this directly impacts error metrics, p-values in hypothesis testing, and confidence intervals. If there is heteroscedasticity in the residuals (non-constant variance), this could signal that the model might be under or over estimating uncertainty, which could become problematic and misleading for inferences that are gathered from the model. A good way to first diagnose this characteristic is by visualizing a scatter plot of the residuals against their fitted values as seen below (Figure 4).

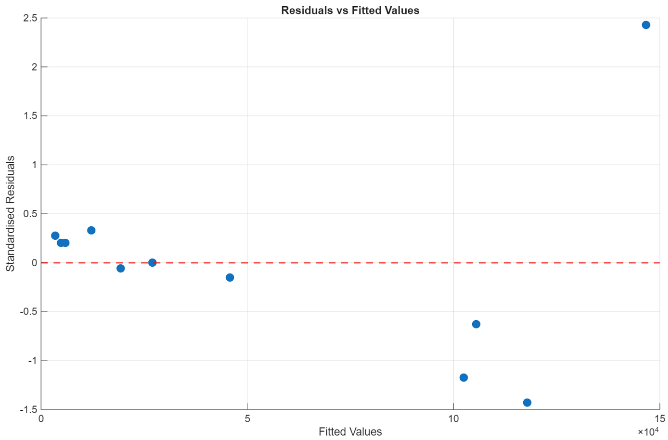

Figure 4: Scatter plot of residuals vs fitted values from simple linear regression model.

The key attributes to look for in this visualization would be a curved pattern indicating a non-linear relationship, or in this case a funnel shape, where it seems that variance is growing with X. Although the plot (Figure 4) does seem to depict variance growing with X, it is important to note that this is over a small sample size, and the pattern is harder to distinguish without recognizing the outlier mentioned earlier.

After some research into methods of verifying this relationship quantitatively, it was settled upon to utilize a method from Gregory Gundersen, in which a test called the Breusch–Pagan Test was implemented (*Appendix: Code 1.4*). The test involved squaring the residuals from the earlier linear regression and then fitting an auxiliary linear regression model between the predictor, GDP in this instance, and those squared residuals. Under the null hypothesis assumption of constant variance, the Breusch–Pagan test statistic is calculated by multiplying the sample size and the R^2 value from the auxiliary model. The p-value is then calculated as the probability of observing this test statistic under the chi-squared distribution.

The attained **p-value was approximately equal to 0.009**, in which case it is concluded that there is statistical evidence to reject the null hypothesis, and assume that heteroscedasticity is evident in the residuals.

To combat concerns of potential unreliability in the earlier linear regression model, which showed heteroscedasticity in residuals, it is recommended to fit a weighted regression model. After additional research, a method of fitting a weighted least squares regression model was attained from Penn State University. This method involved obtaining the absolute values of the residuals from the earlier regression model and were once again used for fitting an auxiliary, this time between the GDP and absolute residuals. After fitting the auxiliary model, weights were defined from its fitted values and were then used to fit a weighted regression (*Appendix: Code 1.5*).

Additional considerations on measurement bias and model assumptions. The GDP and logistics cost data underlying this analysis are reported by individual countries, and reporting standards vary in quality, methodology, and timeliness. Differences in how countries define and measure logistics expenditures (e.g. inclusion or exclusion of warehousing, customs overhead, or informal transport) can introduce systematic measurement bias that the model treats as random noise. Furthermore, the simple linear model assumes a constant marginal relationship between GDP and logistics costs across the entire GDP range. Extrapolation beyond the training range—such as predicting for a country with substantially higher or lower GDP than Germany or Romania—is unreliable because there is no evidence the linear trend continues outside observed values. Finally, with only $n = 11$ observations, all inferences carry wide uncertainty, and the model's generalizability to countries outside this sample is limited.

A plot comparing both regression models can be seen below (Figure 5):

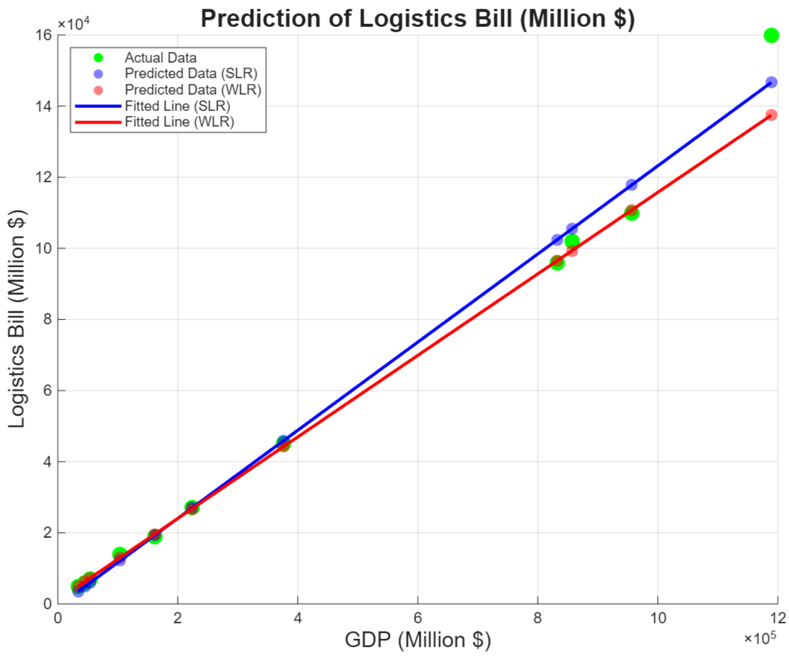

Figure 5: Comparison of simple linear regression (SLR) and weighted least squares (WLS) regression models.

Or just a plot of the weighted regression with confidence and prediction intervals in (*Appendix: Figure 1.14*).

The Weighted Model can then be interpreted as: **Logistics Bill = 0.11464 · GDP + 1111.1** from (Table 6).

Table 6: Weighted Regression Model

	Estimate	SE	tStat	pValue
Intercept	1111.1	134.34	8.2707	1.6951e-05
x_1	0.11464	0.0039241	29.214	3.1473e-10

The error metrics of the weighted regression model are below (Table 7):

Table 7: Error Metrics — Weighted Regression Model

$R^2 = 0.9896$	RMSE = 6874.6418	MAE = 2712.4213

Residual analysis of the weighted regression model was conducted in the same manner as before with the simple linear regression. This included gathering descriptive statistics, a QQ-plot (*Appendix: Figure 1.9*), a Histogram Plot (*Appendix: Figure 1.10*), a Chi-Squared test (*Appendix: Code 1.3*), a Residual vs Fitted Values Plot (*Appendix: Figure 1.12*) and a BP test (*Appendix: Code 1.4*).

It was interesting to note that visually the residuals from the weighted model followed normality better on the QQ plot overall but the residual error from predicting Germany became more pronounced. This also seemed to skew the histogram plot more and cause the chi-squared test to conclude in rejecting the null hypothesis of normality, giving an extremely small p-value close to 0. However, this interpretation could be heavily influenced by that mentioned residual from predicting Germany, and that should be taken into consideration. The residuals vs fitted values plot showed a more constant variance amongst the residuals outside of predicting Germany, this was also reflected when conducting the BP test again for the weighted model and obtaining a **p-value = 0.054042** which showed there was not enough statistical evidence to reject the null hypothesis of constant variance. Overall, the weighted regression model had a slightly worse RMSE but provided a better MAE metric and residuals with more consistent variance, this could indicate the weighted regression to be more reliable, but it does not mean that the weighted model should be considered blindly better in comparison to the simple linear regression.

(e) Does Data Suggest the Regression Line Passes Through the Origin?

This was an interesting consideration and could be determined through the statistical significance of the y-intercept, and whether the 95% confidence interval for the y-intercept contained 0 for each model.

As mentioned earlier for the initial linear regression model, the 95% confidence interval for the intercept ran from about -6500 **to** 5000 which includes 0 in its possible range, and the **p-value for the intercept was 0.78245** > 0.05 suggesting the regression line passes through the origin or at least that it is not statistically different to one that does.

On the other hand, the weighted regression model's 95% confidence interval for the y-intercept ran from about **800 to 1400** and the **p-value for the intercept was 1.6951e-05** < 0.05 suggesting that it is indeed statistically different from 0.

If the maintained assumptions are that the weighted regression is a more reliable model for the data, the data would not suggest a regression line passing through the origin but in a simpler model it would, either way it is very unreasonable in the real world for a country with a GDP of 0 or close to 0 and the need of predicting its Logistics Bill.

Question 2(i): Testing Whether Line Speed Influences Fill Level Deviation in Soda Manufacturing

(a) Effect of Line Speed on Fill Level Deviation

Because the initial interest is only on how line speed influences fill level deviation without considering carbonation, it is a good idea to get some basic descriptive statistics on the entire dataset. This will also give a good baseline for further analysis. A basic summary of the data is given below (Table 8):

Table 8: Fill Level Deviation Summary Statistics Across Line Speeds (All Carbonation Levels)

	Min	Median	Max	Mean	Std
Speed_210	0.4	2	3	1.93	0.7103
Speed_240	0.5	1.95	6	2.35	1.7822
Speed_270	1.2	3.5	4.7	3.51	1.081
Speed_300	2.4	3.7	5.1	3.66	0.7947

It can be noted that the average fill level deviation is increasing as line speed increases. Next, some simple visualizations of the combined data can reveal some influences and relationships. Below (Figure 6) is a scatter plot of all the data with a superimposed least-square line created from the built-in MATLAB function `lsline`. There are also box plots of the data for each line speed to realize any outliers and the spread of the data (Figure 7), and a line plot of each of the average fill level deviations per line speed (Figure 8).

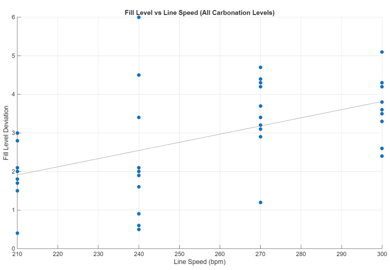

Figure 6: Scatter plot of fill level deviation vs line speed (all carbonation levels) with regression line.

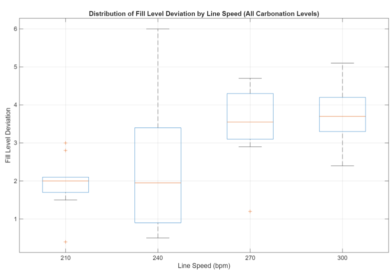

Figure 7: Box plots of fill level deviation by line speed (all carbonation levels).

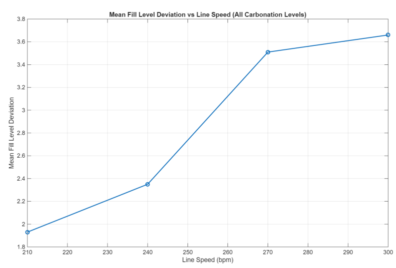

Figure 8: Line plot of mean fill level deviation vs line speed (all carbonation levels).

We can notice from the plots that there seems to be an increasing relationship between line speed and fill level deviation where the medians and overall distribution move upwards with line speed, the biggest changes seem to happen between line speeds of 240 and 270. It also looks like there are some outliers in the groups with line speeds 210 and 270, however none of these values are too extreme in relation to the rest of the data among the groups, so it is probably not necessary to remove them.

Running some quick correlation tests will help quantify these relationships:

Table 9: Correlation Analysis: Fill-Level Deviation & Line Speed

	Rho	pValue
Pearson	0.537	0.0003576
Spearman	0.583	8.029e-05

Once again, these results from the Pearson and Spearman tests indicate a statistically significant $p < 0.05$, moderately strong positive linear relationship between fill level deviation and line speed. Because the relationship looks roughly linear, it can be reasonable to fit another linear regression using the MATLAB `fitlm` function from earlier (*Appendix: Code 2.1*).

The simple linear regression model that fits will be in the same format as in the first question and is given below (Table 10):

Table 10: Linear Regression Model ($R^2 = 0.29$)

	Estimate	SE	tStat	pValue
Intercept	-2.535	1.38887	-1.8254	0.075802
x_1	0.021167	0.0053994	3.9202	0.00035757

Hypothesis for the Slope:

H_0: ($x_1 = 0$) Line Speed has no effect on mean fill level deviation

H_1: ($x_1 \neq 0$) Line Speed does affect the mean fill level deviation

Because the **p-value for x_1 is approximately 0.00036**, the data does provide strong evidence that line speed influences the fill level deviation; higher line speeds are associated with larger deviations from the target fill level.

Although fitting a linear model is sufficient enough to determine there is a relationship between fill level deviation and line speed, showing that about 29% of variability in fill level deviation is explained in line speed alone ($R^2 = 0.288$), fitting a nonlinear model, such as a polynomial, would likely be better for predicting fill level deviations.

Either way, this model is reasonable enough for this exploration; a scatter plot and QQ plot to check the residuals look random and normally distributed, as was done in Question 1, aid in the reliability of using a linear model for this exploration (Figure 9, 10).

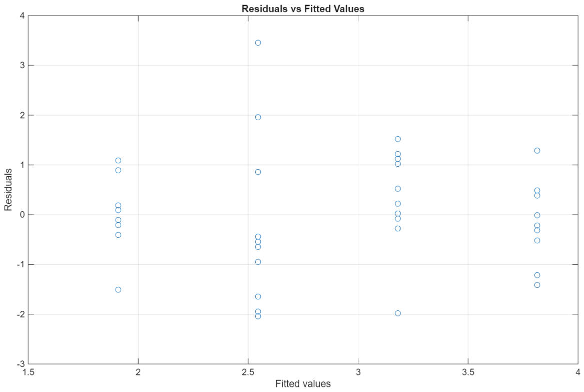

Figure 9: Scatter plot of residuals vs fitted values from SLR of fill level deviation and line speed.

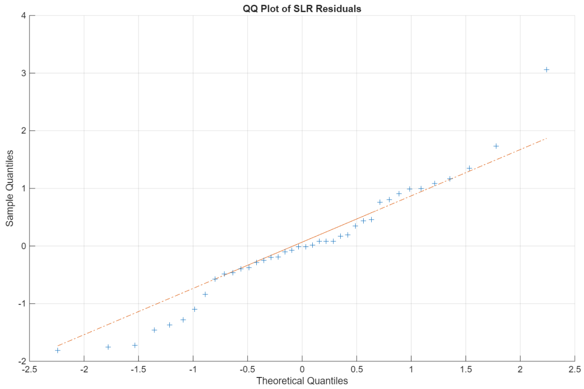

Figure 10: QQ-plot of SLR residuals (fill level deviation and line speed, all carbonation levels).

(b) Interaction Between Carbonation and Line Speed

Now it is known that there certainly exists a relationship between fill level deviation and line speed, the next interest would be to analyze how carbonation plays a role in the process. The data was given so that it includes the fill level deviation for each line speed, but it is also split, where half of the fill level deviation data above is from soda of 10% carbonation and the other half 12%. This can be shown in a scatter as before but treating the data separately based on carbonation (Figure 11).

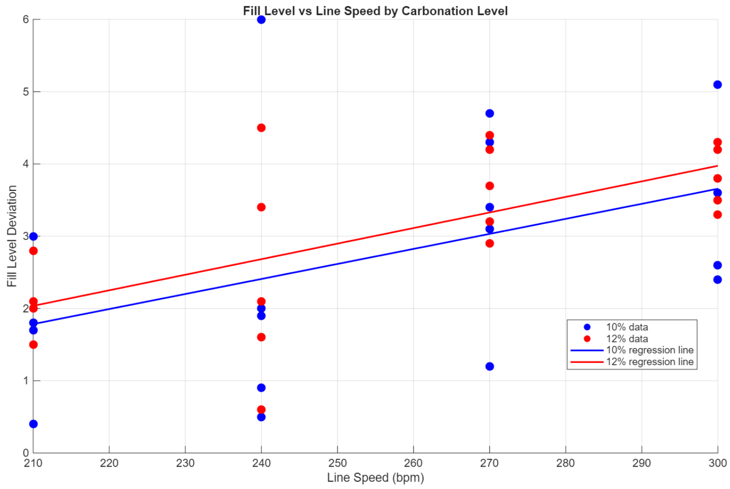

Figure 11: Scatter plot of fill level deviation vs line speed by carbonation level with regression lines.

It is interesting to see the linear relationships compared for 10% and 12% carbonation in the figure; seemingly they run parallel, just with 12% carbonation having a slightly higher average fill level deviation per line speed.

Also, if interested, individual scatter plots, box plots and line plots for both 10% and 12% carbonation as done before were used for individual analysis and can be found (*Appendix: Figures 2.4, 2.5, 2.6, 2.7*).

In tables of summary statistics below (Table 11, 12), as well as (*Appendix: Figures 2.4, 2.5, 2.6, 2.7*) mentioned above, it seems as though the fill level deviation data for 12% is more tightly spread around its average, and the average fill level deviation appears to be larger for soda with 12% carbonation; however, this should be further explored and verified quantitatively.

Table 11: Fill Level Deviation Summary Statistics Across Line Speeds (10% Carbonation Level)

10% Carbonation	Min	Median	Max	Mean	Std
Speed_210	0.4	1.8	3	1.78	0.9284
Speed_240	0.5	1.9	6	2.26	2.187
Speed_270	1.2	3.4	4.7	3.34	1.3612
Speed_300	2.4	3.6	5.1	3.5	1.0817

Table 12: Fill Level Deviation Summary Statistics Across Line Speeds (12% Carbonation Level)

12% Carbonation	Min	Median	Max	Mean	Std
Speed_210	1.5	2	2.8	2.08	0.4658
Speed_240	0.6	2.1	4.5	2.44	1.5307
Speed_270	2.9	3.7	4.4	3.68	0.638
Speed_300	3.3	3.8	4.3	3.82	0.4324

To determine quantitatively whether carbonation actually has an influence on fill level deviation, a two-way ANOVA can be implemented which treats Carbonation and Line Speed as factors where each sub class, (10%, 12%) and (210, 240, 270, 300) respectively, are categorical labels for the fill level deviation in that column. ANOVA is useful because it will compare the mean fill level deviations between the four different line speeds and two different carbonation groups. This is done in using MATLAB's built-in `anovan` function (*Appendix: Code 2.2*). The results are below (Table 13).

ANOVA assumptions. Before interpreting the results, it is important to verify the assumptions underlying the two-way ANOVA:

- **Normality:** The chi-squared goodness-of-fit test on the regression model residuals returned $p = 0.1039$, failing to reject normality at $\alpha = 0.05$.
- **Equal variance:** Levene's test (*Appendix: Code 2.3*) gave $p = 1.46 > 0.05$, providing no evidence against equal variances across the Carbonation $\times$ LineSpeed groups.
- **Independence:** Observations are assumed independent, which is reasonable given that each fill level measurement comes from a separate bottle on the production line.

The design is balanced with 5 observations per cell (4 line speeds $\times$ 2 carbonation levels $\times$ 5 replicates = 40 observations total).

Table 13: Two-Way ANOVA for the Effects of Carbonation, Line Speed, and Their Interaction on Fill Level Deviation

	Sum Sq	Df	Mean Sq	F	pVal
Carbonation	0.81225	1	0.8122	0.5497	0.4639
Line Speed	21.875	3	7.2916	4.9342	0.0063
Carbonation:Line Speed	0.03875	3	0.0129	0.0087	0.9989
Error	70.014	32	1.4778	—	—

From the results of ANOVA, there is a statistically significant effect of line speed on fill level deviation (**p-value = 0.006** < 0.05) which verifies the earlier findings; however, there seems to be no statistically significant main effect of carbonation on fill level deviation having a **p-value = 0.46** > 0.05. There also seems to be no evidence of an interaction between carbonation and line speed which has a **p-value = 0.999** > 0.05. The non-significant interaction term means that the effect of line speed on fill deviation is consistent across both carbonation levels—increasing line speed increases deviation by roughly the same amount regardless of whether the soda is carbonated at 10% or 12%.

An additional check is done by fitting another regression model, using MATLAB's `fitlm` function, with the main effects of carbonation, line speed and their interaction, this model is like previous regression models, just with categorical predictors and interaction terms, where MATLAB has chosen to use 10% carbonation and the 210 line speed as a baseline that is incorporated into the intercept value below in the model output (Table 14).

Table 14: Estimated Coefficients for the Carbonation and Line Speed Linear Regression Model

	Estimate	SE	tStat	pValue
Intercept	1.78	0.544	3.27	0.00255
Carbonation_12%	0.3	0.768	0.39	0.69897
LineSpeed_240	0.48	0.769	0.62	0.53684
LineSpeed_270	1.56	0.769	2.03	0.05084
LineSpeed_300	1.72	0.769	2.24	0.03237
Carbonation_12%:LineSpeed_240	−0.12	1.087	−0.11	0.91281
Carbonation_12%:LineSpeed_270	0.04	1.087	0.04	0.97088
Carbonation_12%:LineSpeed_300	0.02	1.087	0.02	0.98544

Here it would be useful to consider the model's adjusted R^2, which penalizes the fit for the number of parameters in the model.

$R^2_{\text{adj}} = 0.177$, showing that the model explains some variation in fill level deviation but is not very strong. More importantly, the **coefficient p-values** < 0.05 show that increasing line speed has a statistically significant effect on fill level deviation, whereas the carbonation term and all other interaction terms are not significant (**p-values** > 0.05). This further confirms the conclusion that changing the line speed impacts fill level deviation but fill level deviation does not depend on whether the soda is carbonated at 10% or 12%. Once again, these findings are reinforced by another ANOVA table on the regression model using MATLAB's `anova` function, showing how variation in fill level deviation is split between the two main effects, their interaction, and residual error.

(*Appendix Table 2.8*)

The next verification that is done after gathering insights from conducting an ANOVA on the regression model is to analyze the model's residuals once again, first visually checking a scatter plot of the residuals versus fitted values and a QQ-plot to make sure they seem random and normally distributed (Figure 12, 13).

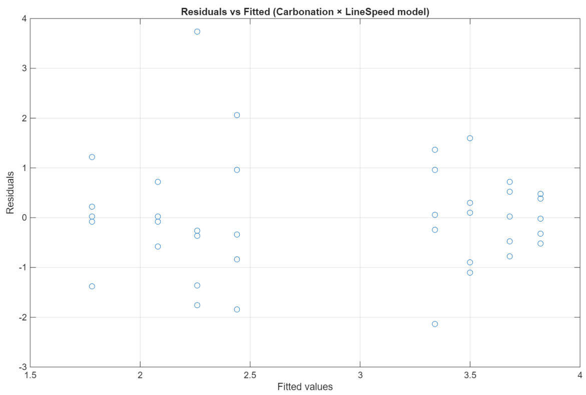

Figure 12: Scatter plot of residuals vs fitted values from (Carbonation × Line Speed) regression model.

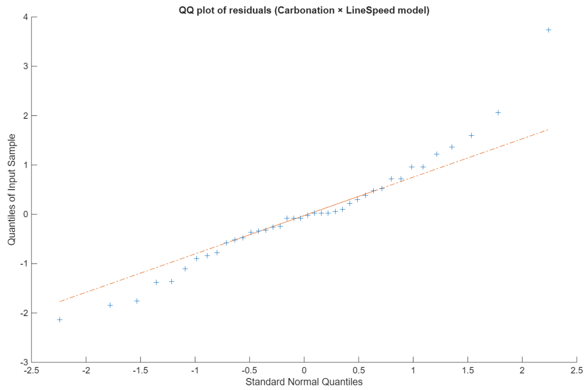

Figure 13: QQ-plot of residuals from (Carbonation × Line Speed) regression model.

The residuals look to be roughly homoscedastic and approximately normal with a few outliers. For an extra quantitative check that there is equal variance among the residuals, Levene's test can be implemented (*Appendix: Code 2.3*).

H_0: All Carbonation × LineSpeed groups have equal variances

H_1: At least one group has a different variance

Levene's test gave **p-value = 1.46** > 0.05, so there is no strong evidence to reject equal variances across the carbonation and line speed groups, supporting the reliability of implementing the two-way ANOVA.

The same checks of using Sturges' to calculate optimal bins, generating a histogram with a normal curve overlay (Figure 14), and running a Chi-Squared Goodness-of-Fit Test for normality were also used to supplement the visual inspection from the QQ-plot (Figure 13).

The Chi-Squared test returned a **p-value = 0.1039**, meaning that at a significance level of 5%, the residuals can be considered normal.

Finally, a post-hoc test can be useful. Since the main effect of line speed was significant, and the carbonation interaction between was not, a Tukey multiple comparison on differing line speeds can be used to see how specific speeds influence mean fill level deviation (Figure 15).

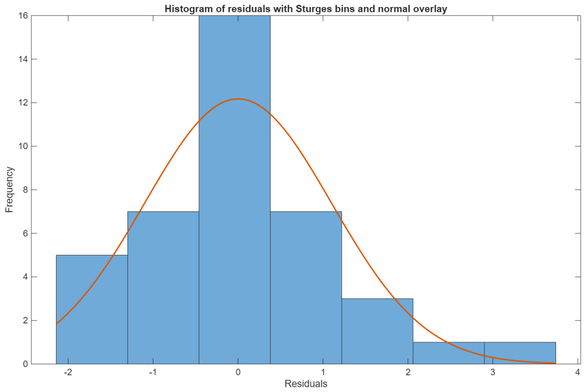

Figure 14: Histogram plot of residuals from (Carbonation × Line Speed) regression model.

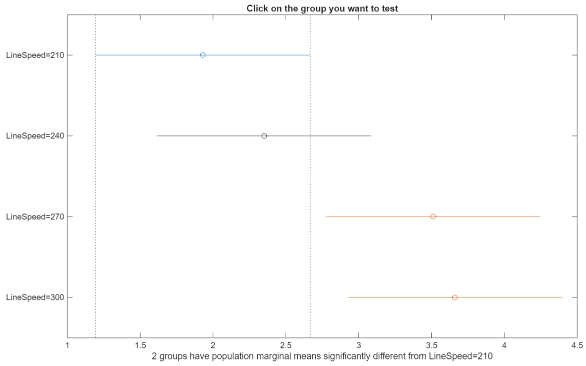

Figure 15: Post-hoc Tukey multiple comparison plot for line speed levels.

The post-hoc test showed that average fill-level deviations for groups with line speed 270 and 300 are statistically different from the group with a line speed of 210, and that no groups' average fill-level deviation were statistically different from having a line speed of 240. This implies that if the goal is to minimize deviation and maximize efficiency, line speeds of 270 and 300 might want to be reconsidered; a line speed of 240 is not statistically different in fill level deviation from other speeds and could be more optimal.

Question 2(ii): Poisson Model for Breakdowns

(a) Justification and Model Setup

A Poisson model is appropriate for modeling this data, where $\lambda = 1.5$ breakdowns per shift. This is because the goal is to count the number of breakdowns in a discrete time interval (one 8-hour shift). The average rate is given and is considered constant (an average of 1.5 breakdowns per 8-hour shift). It is also important that breakdowns can be assumed random and independent over time.

Why Poisson is appropriate:

- The response variable is a *discrete count* of events (breakdowns) in a fixed interval (one shift).
- Breakdowns are *rare events* relative to the total operating time within a shift.
- The average rate λ is assumed *constant*—there is no evidence of wear-out or increasing failure rate within a shift.
- Individual breakdowns are assumed to occur *independently* of one another; one breakdown does not make the next more or less likely.

$$X \sim \text{Poisson}(\lambda = 1.5)$$

(b) Probability of Exactly Two Breakdowns on the Night Shift

There should be approximately a **25.1021%** chance of exactly two breakdowns during the night shift (*Appendix: Code 2.4*).

P(X) %
25.1021

X	P(X) %	Cumulative %
0	22.3130	22.3130
1	33.4695	55.7825
2	25.1021	80.8847
3	12.5511	93.4358
4	4.7067	98.1424
5	1.4120	99.5544
6	0.3530	99.9074
7	0.0756	99.9830
8	0.0142	99.9972

(c) Probability of Fewer Than Two Breakdowns on the Afternoon Shift

There should be approximately a **55.782%** chance of having fewer than two breakdowns during the afternoon shift (*Appendix: Code 2.4*).

(d) Probability of No Breakdowns Over Three Consecutive Shifts

There should be approximately a **1.1109%** chance of exactly no breakdowns during three consecutive shifts (*Appendix: Code 2.5*).

Over three consecutive shifts the total expected breakdowns become $\lambda_{\text{total}} = 3 \times 1.5 = 4.5$, so $X_{\text{total}} \sim \text{Poisson}(4.5)$, and we seek $P(X_{\text{total}} = 0) = e^{-4.5} \approx 1.11\%$.

X	P(X) %	Cumulative %
0	1.1109	1.1109
1	4.9990	6.1099
2	11.2479	17.3578
3	16.8718	34.2296
4	18.9808	53.2104
5	17.0827	70.2930
6	12.8120	83.1051
7	8.2363	91.3414
8	4.6329	95.9743

Practical interpretation for maintenance planning. The cumulative probabilities translate directly into staffing and preparedness decisions. For a single shift, the cumulative probability of 3 or fewer breakdowns is approximately 80.9% (from the single-shift table above), meaning that if maintenance staff are equipped to handle up to 3 breakdowns per shift, they will be adequately prepared roughly 4 out of 5 shifts. Over 3 consecutive shifts, the probability of *zero* breakdowns is only 1.1%, meaning it is virtually certain that at least one breakdown will occur over a full day of operations. This implies that continuous maintenance coverage is essential—scheduling maintenance staff only "as needed" would be impractical, as the probability of a completely breakdown-free day is negligible.

Question 3: Statistical Evaluation of Training Methods and Spring Production

Before proceeding to the detailed analysis, Table 15 consolidates all hypothesis test results from this section for quick reference.

Table 15: Consolidated Hypothesis Test Results for Question 3

Test	Purpose	p-value	Decision
Chi-sq (Computer)	Normality	0.2313	Fail to reject
Chi-sq (Group)	Normality	0.1548	Fail to reject
F-test	Equal variance	0.0461	Reject
Welch t (two-sided)	Equal means	0.12	Fail to reject
Welch t (one-sided)	CAL faster	0.06	Fail to reject (borderline)

(i) Computer Assisted & Group-Based Learning

Summary statistics of the data show that the computer-assisted group has a lower mean assembly time than group-based, 17.85 and 20.1675, respectively. Also, the median values are larger and spread of the data is more varied for group-based learning, which can easily be seen by plotting a box plot (Figure 16). This would initially suggest that the computer-assisted group is faster on average and less variable when compared to group-based; however, this will need to be confirmed in hypothesis testing.

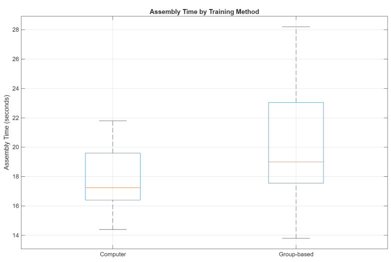

Figure 16: Box plots for assembly times by training method.

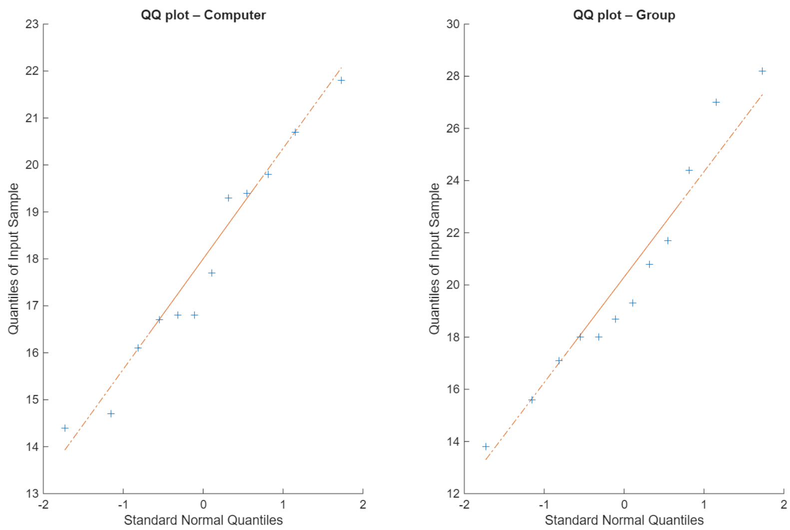

Figure 17: QQ-plots of assembly times by training method to assess normality.

The box plots (Figure 16) for each group show no extreme outliers, and the QQ-plots (Figure 17) for both groups show points lying relatively close to the 45-degree line suggesting that the assembly times for both groups have approximately normal distributions. This is further confirmed using the method of generating histograms (*Appendix: Figure 3.3,* *3.4*) with Sturges' bins and running a Chi-Squared test the same way as before. The Chi-Squared test for the computer-assisted group returned a **p-value = 0.2313**, and the group-based learning returned a **p-value = 0.1548**, both of which are greater than the 0.05 significance level, and in turn do not provide strong enough statistical evidence to reject the null hypothesis of normality.

Since the assembly times in each group appear to be approximately normal based on the Chi-squared test and QQ-plots, it is reasonable to use an F-test to compare the population variances of the two groups (*Appendix: Code 3.1*). A two-tailed F-test displayed a **p-value = 0.0461** < 0.05, meaning there is enough evidence to reject the null hypothesis of equal variance and conclude that the two groups differ in variability, with group-based learning showing larger variability in assembly times than the computer-assisted group.

Why Welch's *t*-test instead of pooled: Because the F-test established that the two populations have unequal variances, a standard pooled t-test is inappropriate. A pooled test assumes equal variances and combines both sample variances into a single pooled estimate of σ^2; when this assumption is violated, the pooled standard error can be too small, inflating the test statistic and producing misleadingly low p-values (increased Type I error). Welch's t-test avoids this by estimating each group's variance separately and adjusting the degrees of freedom downward via the Welch–Satterthwaite equation, yielding a more robust and conservative test.

To determine if computer-assisted learning is better on average than group-based learning, a two-sample t-test is needed. However, because the variances are unequal, it is best practice to utilize Welch's t-test specifically as this test is supported when groups have unequal variance (*Appendix: Code 3.2*). Welch's t-test returned a **two-sided p-value = 0.12** > 0.05, so there is not enough evidence to reject the null hypothesis of equal average assembly times. For the specific hypothesis test that computer-assisted learning leads to shorter average assembly times, the **p-value = 0.06**, which is borderline at the 5% significance level, but still does not provide enough statistical evidence to conclude that the computer-assisted group is significantly better than group-based learning.

The two-sided result ($p = 0.12$) shows no significant difference in either direction, while the one-sided result ($p = 0.06$) suggests a promising but inconclusive trend in favor of computer-assisted training. A larger sample size would be needed to determine whether this borderline effect is real or due to chance. If other metrics besides the average assembly times are also considered, computer-assisted learning could still be preferred for lower variability, implying more consistency in the times that are observed.

(ii)(a) Proportion of Springs Outside Specification

Assuming that the spring rates are normally distributed with a sample mean of 44.175 Newtons needed to compress a spring one centimeter with a sample variance of approximately 1.017 Newtons (used to calculate the sample standard deviation), MATLAB's built-in `normcdf` function can be used to calculate the probabilities a spring falls above the upper spec limit of 46.5 Newtons or below the lower spec limit of 41.5 Newtons (*Appendix: Code 3.3*). The probability the spring falls out of spec is then just the sum of the probabilities of it being above or below the respective spec limits. The probability of being outside of spec was calculated to be **1.46%** and the probability of being within the spec range was **98.54%**.

Justification for the normal distribution. Spring forces are continuous measurements resulting from many small, independent manufacturing variations (material properties, coiling tolerances, heat treatment fluctuations). By the Central Limit Theorem, the aggregate effect of these numerous small sources of variation produces an approximately normal distribution, consistent with the QQ-plots and histogram analysis. Since only about 1 or 2 of 100 springs seem to fall outside of spec, the springs could be considered to perform pretty well. **Whether this is acceptable depends on context:** in safety-critical applications (e.g. automotive suspension, medical devices), a 1.46% defect rate may be unacceptable and warrant process tightening. In lower-criticality applications, this rate may be economically acceptable. The focus could then be on maintaining current metrics, with cost–benefit analysis guiding whether process improvements are justified.

(ii)(b) Is the Mean Spring Rate on Target?

To determine if the mean spring rate is on target, a one-sample t-test using MATLAB's `tcdf` (*Appendix: Code 3.4*) is applied to the single normal sample with an unknown population variance (only given sample variance, otherwise could also use a z-test).

$$\text{Standard Error:} \quad SE = \frac{s}{\sqrt{n}}, \quad \text{where } s = 1.017,\ n = 100$$

$$\text{Test Statistic:} \quad t = \frac{\bar{x} - \mu_0}{SE}, \quad \text{where } \bar{x} = 44.175,\ \mu_0 = 44 \text{ N}$$

$$\text{Degrees of Freedom:} \quad df = n - 1 = 99$$

Hypothesis & Confidence Interval:

$$H_0 : \mu = \mu_0 = 44 \text{ N}; \qquad H_1 : \mu \neq \mu_0; \qquad CI = \bar{x} \pm t_{\text{crit}} \times SE$$

The calculated **two-sided p-value = 0.0858** > 0.05, and the 95% confidence interval for the true mean is [**43.975, 44.375**], which includes the target value of 44 N. Therefore, there is not sufficient evidence that the population mean spring rate is off target.

(ii)(c) Estimating Population Mean Spring Rate With 95% Confidence

The best estimate of the population mean is the sample mean 44.175 N. Using the t-based confidence interval from the previous section, the 95% confidence interval is approximately [**43.975, 44.375**], which estimates the true mean spring rate lies somewhere within this range with 95% confidence.

The graph below (Figure 18) shows the t-distribution of the test statistic under the null hypothesis that the mean is equal to 44 N and there are 99 degrees of freedom. The black curve is the probability density function of a t-distribution with 99 degrees of freedom. The red shaded regions beyond the two blue dotted lines at $\pm t_{\text{critical}}$ values are the 5% rejection regions for a two-sided t-test with a 5% significance level, 2.5% in each region.

The red dotted line marks the observed t-value from the sample ($t_{\mathbf{stat}} = 1.73$). Because the red line lies between the blue critical values, the observed statistic is not extreme enough to fall in the rejection region. This visually confirms the numerical result (**p-value = 0.0858**), therefore there is not enough statistical significance to reject the null hypothesis, and it should be concluded that the sample does not provide strong enough evidence that the true mean spring rate differs from the target value of 44 N.

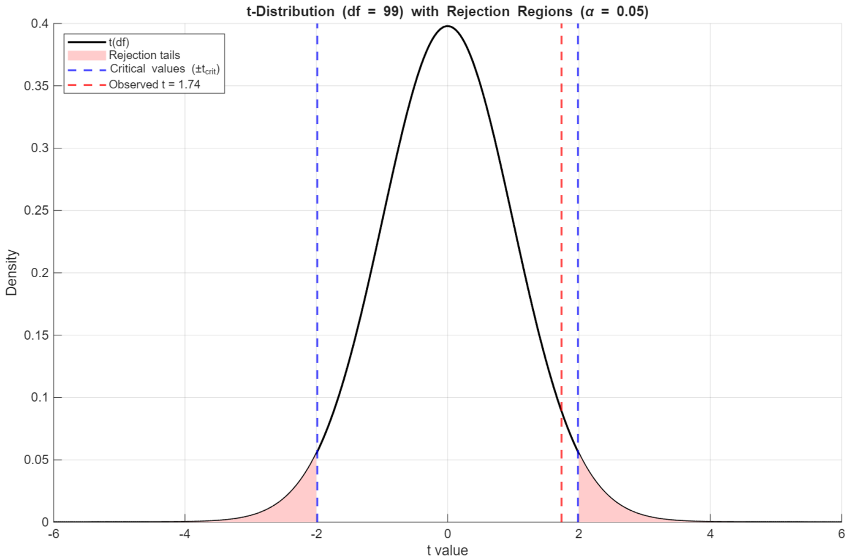

Figure 18: t-distribution for one-sample t-test on mean spring rate with critical regions and observed t-stat.

Reference List

[1] Gundersen, G. (2022). *Breusch–Pagan Test for Heteroscedasticity.* [online] gregory-gundersen.com. Available at: https://gregorygundersen.com/blog/2022/01/31/breusch-pagan/.

[2] Penn State Eberly College of Science: Statistics Online Courses. (n.d.). *13.1 — Weighted Least Squares — STAT 501.* [online] Available at: https://online.stat.psu.edu/stat501/lesson/13/13.1.

Appendix (MATLAB Code, Additional Figures, and Tables)

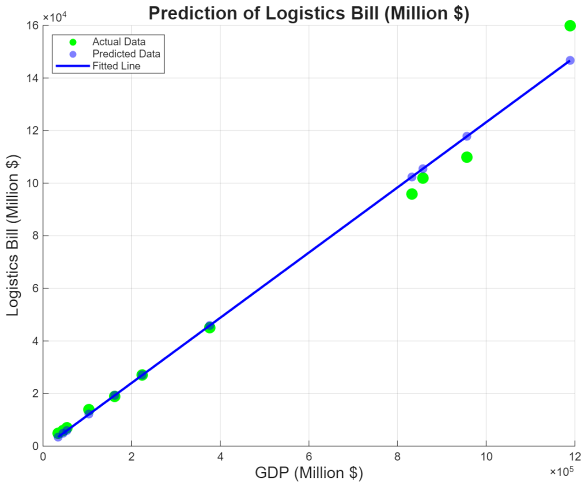

Figure 19: First plot of simple linear regression model.

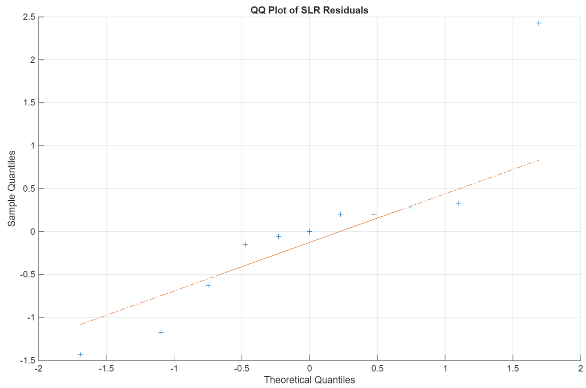

Figure 20: QQ-plot of residuals from simple linear regression model.

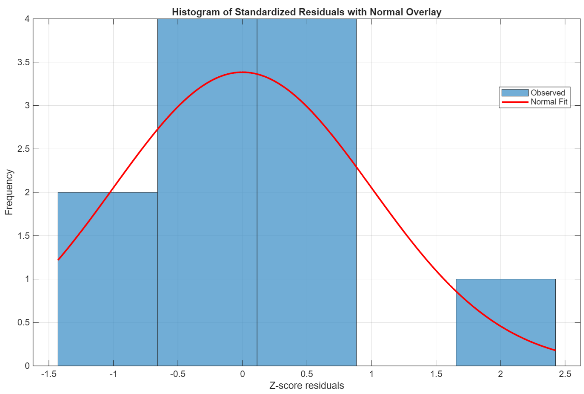

Figure 21: Histogram plot of residuals from simple linear regression model.

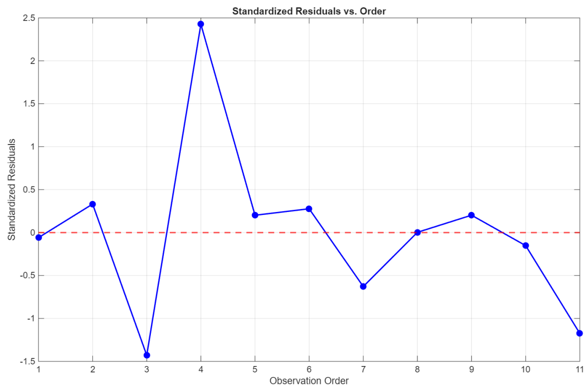

Figure 22: Line plot of residuals vs observation order from simple linear regression model.

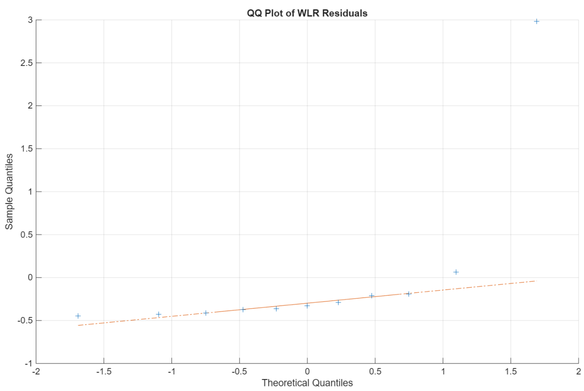

Figure 23: QQ-plot of residuals from weighted least squares regression model.

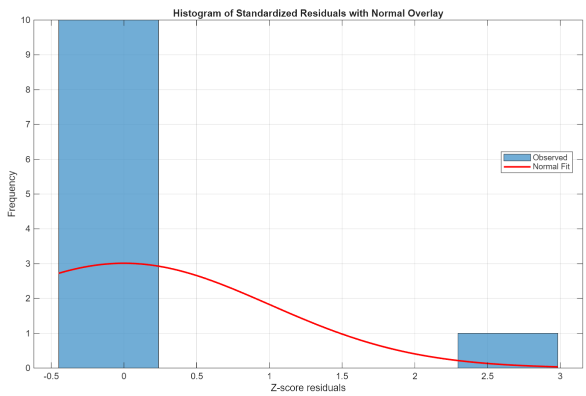

Figure 24: Histogram plot of residuals from weighted least squares regression model.

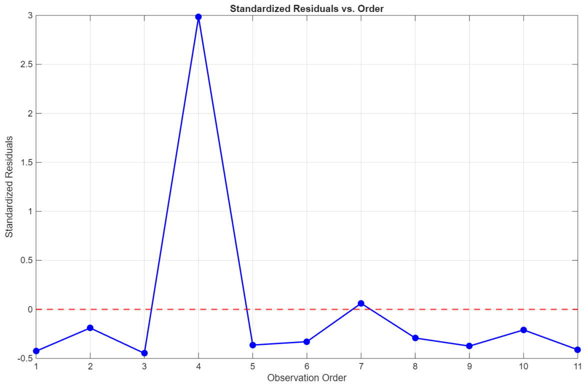

Figure 25: Line plot of residuals vs observation order from weighted least squares regression model.

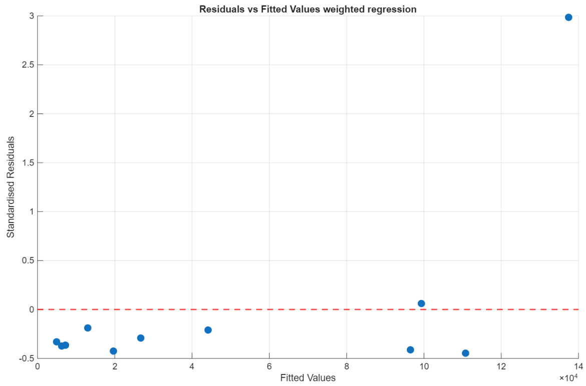

Figure 26: Scatter plot of residuals vs fitted values from weighted least squares regression model.

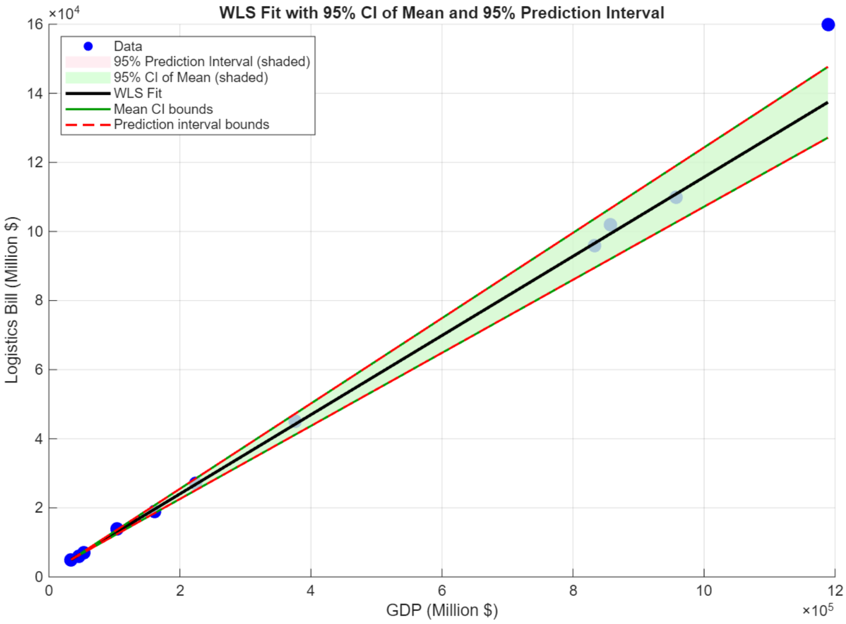

Figure 27: Weighted least squares regression model with 95% confidence interval for mean response and 95% prediction interval.

```
% fit linear regression model

mdl = fitlm(GDP, Logistics_Bill);
```

Code 1.1 (Fit Simple Linear Regression of Logistics Bill on GDP)

```
% 95% confidence interval for regression coefficient
confInt = coefCI(mdl, 0.05);
disp('95% Confidence Intervals for Coefficient:');
disp(confInt);
|
% Extract and print CI for intercept and slope separately
CI_intercept = confInt(1,:);   % row 1: intercept
CI_slope     = confInt(2,:);   % row 2: slope for GDP
```

Code 1.2 (Finding Confidence Intervals)

```
% Chi-square statistic for distributional check
chiSqValues = ((Observed - Expected).^2) ./ Expected;
chiSqTotal  = sum(chiSqValues);  % overall chi-square statistic

% Degrees of freedom
df = k - 1 - 2; % Subtract 2 estimated parameters (μ, σ)

% Chi-square right-tail p- using chi2cdf
pValue = 1 - chi2cdf(chiSqTotal, df);
```

Code 1.3 (Chi-Squared Normality Test)

```
% get residuals
residuals_test = mdl.Residuals.Raw; % Get the raw residuals

% Square the residuals
squared_residuals_test = residuals_test.^2;

% Run an auxiliary regression of squared residuals on the predictor
aux = fitlm(GDP, squared_residuals_test);

% Calculate the test statistic and p-value
n_test = length(residuals_test); % Sample size
R2_test = aux.Rsquared.Ordinary; % R-squared from the auxiliary regression
stat_test = n_test * R2_test; % test statistic

% dof
df_test = size(GDP, 2);

% Calculate the p-value
pValue_test = 1 - chi2cdf(stat_test, df_test);

% Display the results
fprintf('Breusch-Pagan Test Statistic: %f\n', stat_test);
fprintf('Breusch-Pagan p-value: %f\n', pValue_test);
```

Code 1.4 (Breusch–Pagan Test)

```
% get absolute residuals
absolute_resid = abs(residuals_tbl.Residuals);

% fit auxillary regression
mdl_aux = fitlm(GDP, absolute_resid);

% fitted values from aux
aux_fitted= mdl_aux.Fitted;

% calculating weights
weights = 1 ./ (aux_fitted .^ 2)

% fitting weighted regression model
mdl_2 = fitlm(GDP, Logistics_Bill, 'Weights', weights);
```

Code 1.5 (WLS Regression)

```
mdl_speed = fitlm(LineSpeed_All, FillLevel);
disp(mdl_speed);
```

Code 2.1 (Simple Linear Regression of Fill Level Deviation on Line Speed)

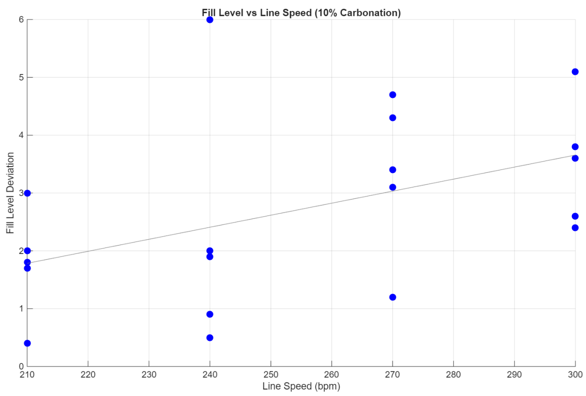

Figure 28: Scatter plot of fill level deviation vs line speed (10% carbonation) with regression line.

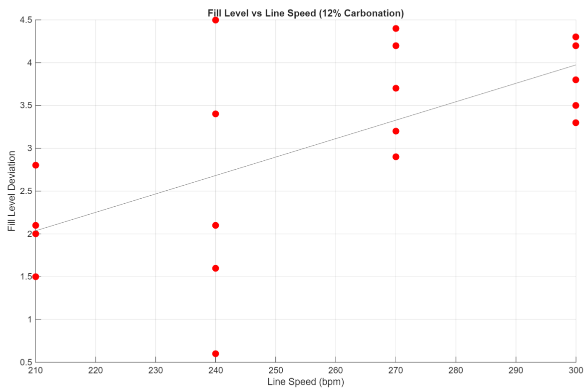

Figure 29: Scatter plot of fill level deviation vs line speed (12% carbonation) with regression line.

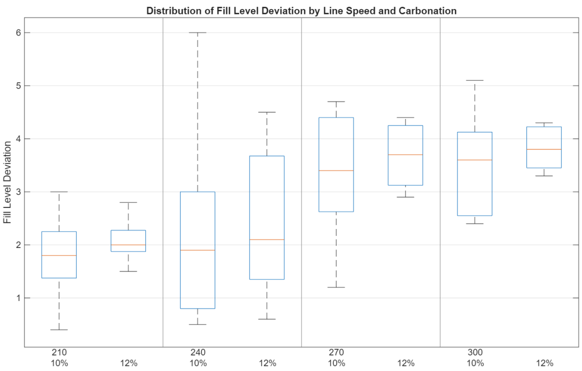

Figure 30: Box plots of fill level deviation by line speed and carbonation level.

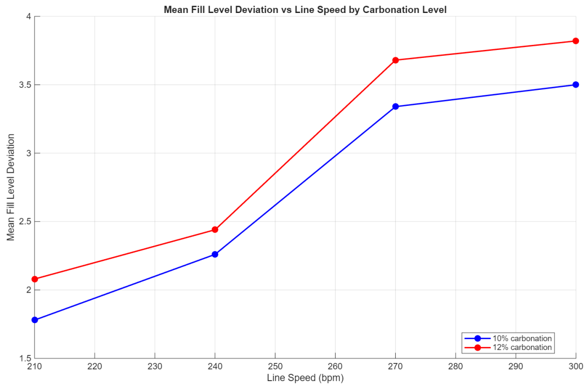

Figure 31: Line plots of average fill level deviation across line speeds for each carbonation level.

```
[p_anova, tbl_anova, stats_anova] = anovan( ...
    Fill_all, ...
    {Carb_all, Speed_all}, ...
    'model', 'interaction', ...
    'varnames', {'Carbonation', 'LineSpeed'}, ...
    'display', 'off');
```

Code 2.2 (Two-Way ANOVA for Carbonation and Line Speed Effects)

```
group_CS = strcat(string(Carb_all), "_", string(Speed_all));

[p_lev, stats_lev] = vartestn( ...
    Fill_all, ...
    group_CS, ...
    'TestType', 'LeveneAbsolute', ...
    'Display', 'off');

fprintf('Levene test p-value = %.4g\n', p_lev);
```

Code 2.3 (Levene's Test for Equal Variances)

Table 16: ANOVA Output for Carbonation and Line Speed Regression Model

	SumSq	Df	MeanSq	F	pValue
Total	70.014	39	107952	—	—
Model	22.726	7	3.2465	2.1969	0.061124
Linear	22.687	4	5.6718	3.8381	0.011707
Nonlinear	0.03875	3	0.012917	0.0087408	0.99885
Residual	47.288	32	1.4778	—	—

```
lambda = 1.5; % avg breakdowns per shift
x_values = 0:8;

% initialize probabilities
P_x = zeros(size(x_values));

% calculate probabilities
for i = 1:length(x_values)
    x = x_values(i);

    % poisson probabilities
    P_x(i) = poisspdf(x, lambda);
end

% calculate the cumulative probability for threshold
cumulative_P = cumsum(P_x);
```

Code 2.4 (Poisson Probabilities for Breakdowns in a Single Shift)

```
% over 3 consecutive shifts the average breakdowns would be
lambda_total = 3 * lambda;

x_values_new = 0:24;

% initialize probabilities
P_x_new = zeros(size(x_values_new));

% calculate probabilities
for i = 1:length(x_values_new)
    x_new = x_values_new(i);

    % poisson probabilities
    P_x_new(i) = poisspdf(x_new, lambda_total);
end

% calculate the cumulative probability for threshold
cumulative_P_new = cumsum(P_x_new);
```

Code 2.5 (Poisson Probabilities for Breakdowns Over Three Consecutive Shifts)

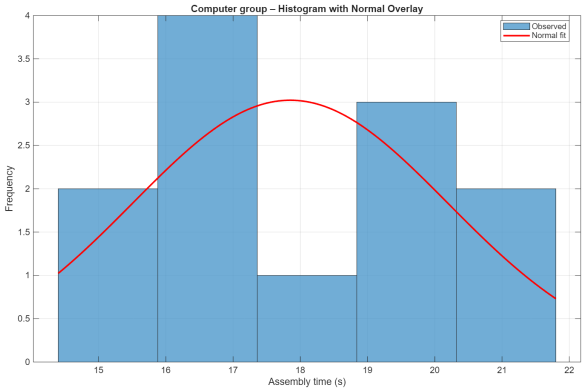

Figure 32: Histogram plot of assembly times for computer-assisted learning with normal distribution.

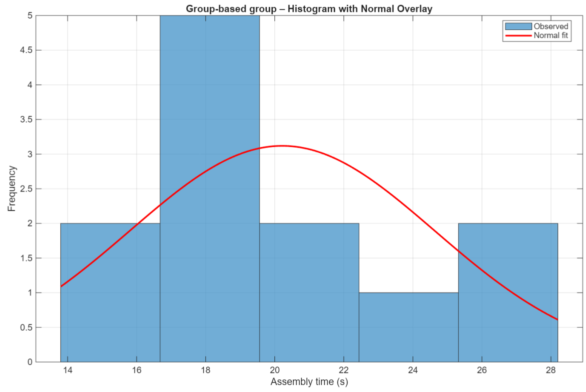

Figure 33: Histogram plot of assembly times for group-based learning with normal distribution.

```
% Put the larger variance in the numerator
if varG >= varC
    F_stat = varG / varC;
    df1 = nG - 1;
    df2 = nC - 1;
    numLabel = 'Group / Computer';
else
    F_stat = varC / varG;
    df1 = nC - 1;
    df2 = nG - 1;
    numLabel = 'Computer / Group';
end

% Two-tailed p-value
p_one = fcdf(F_stat, df1, df2);
p_two = 2 * min(p_one, 1 - p_one);
```

Code 3.1 (F-Test for Equal Variances in Assembly Times)

```
% Welch standard error and t-statistic
se2    = varC/nC + varG/nG;
se     = sqrt(se2);
t_stat = (meanC - meanG) / se;

% Welch degrees of freedom
df = se2^2 / ( (varC^2 / (nC^2 * (nC - 1))) + (varG^2 / (nG^2 * (nG - 1))) );

% One-sided p-value for H1: mu_C < mu_G (left tail)
p_one = tcdf(t_stat, df);

% Two-sided p-value (for reference)
p_two = 2 * min(p_one, 1 - p_one);
```

Code 3.2 (Welch's Two-Sample t-Test for Average Assembly Times)

```
p_below   = normcdf(LSL, mu_est, sigma_est);
p_above   = 1 - normcdf(USL, mu_est, sigma_est);

p_outside = p_below + p_above;
p_inside  = 1 - p_outside;
```

Code 3.3 (Normal Probability of Outside/Inside Spring Rates Specification Limits)

```
SE     = s / sqrt(n); % standard error of sample mean
t_stat = (xbar - mu0) / SE; % t statistic
df     = n - 1; % degrees of freedom

% Two-sided p-value
p_two = 2 * (1 - tcdf(abs(t_stat), df));

% 95% t critical value and CI
t_crit = tinv(1 - alpha/2, df);
margin = t_crit * SE;
CI_low  = xbar - margin;
CI_high = xbar + margin;
```

Code 3.4 (One-Sample t-Test for Mean Spring Rate)