



# **AN ANALYSIS OF UNCERTAINTY IN LEARNING CURVE MODELS**

## **THESIS**

Alexander D. Pollack, First Lieutenant, USAF

AFIT-ENV-MS-21-M-257

**DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY**

**AIR FORCE INSTITUTE OF TECHNOLOGY**

---

---

**Wright-Patterson Air Force Base, Ohio**

**DISTRIBUTION STATEMENT A.**  
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT-ENV-MS-21-M-257

AN ANALYSIS OF UNCERTAINTY IN LEARNING CURVE MODELS

THESIS

Presented to the Faculty

Department of Systems Engineering and Management

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Cost Analysis

Alexander D. Pollack, BS

First Lieutenant, USAF

March 2021

**DISTRIBUTION STATEMENT A.**  
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT-ENV-MS-21-M-257

AN ANALYSIS OF UNCERTAINTY IN LEARNING CURVE MODELS

Alexander D. Pollack, BS

First Lieutenant, USAF

Committee Membership:

Dr. John J. Elshaw  
Chair

Lt Col Clay M. Koschnick, PhD  
Member

Dr. Jonathan D. Ritschel  
Member

### **Abstract**

The dilemma with learning curves and actual costs is "fitting square pegs into round holes." Alternative regression techniques or further modifications to  $Y = aX^b$  may address discrepancies in modeling observed data with learning curves. Adequate data permits further analysis of the constants, variables, and factors that make up a learning curve model. The analysis involves trying a nonlinear solver on the traditional learning curve models (i.e., Wright's cumulative average theory and Crawford's unit learning curve theory) and changing the theoretical first unit cost (T1) to the actual first unit cost (A1). Mergers and acquisitions in the aircraft industry may have reduced certainty in learning curve models since human resources were dispersed to manufacturing facilities worldwide. The results indicated no significant differences between using a nonlinear solver over ordinary least squares. However, the nonlinear solver propagated more uncertainty into the learning curve models. The actual first unit cost (A1) almost rids cumulative average theory models' statistical advantage over unit learning curve theory models. Conversely, mergers and acquisitions reduced uncertainty in learning curve models. Research on program attributes should continue when the appropriate data is available. Cost analysts should request missing data from contractors.

## **Acknowledgments**

Many thanks to my adviser, Dr. John Elshaw, and his colleagues on my thesis committee for making several constructive suggestions. I want to thank Dr. Edward White, Maj Timothy Anderson, and 2d Lt Nicholas Pollack for the helpful discussions on mathematics and statistics in my research. Finally, I would like to thank the Air Force Institute of Technology for providing me with the knowledge and skills necessary to complete my research.

Alexander D. Pollack

## Table of Contents

	Page
Abstract.....	iv
Acknowledgements.....	v
Table of Contents.....	vi
List of Figures.....	viii
List of Tables .....	ix
I. Introduction.....	1
Motivation .....	<b>Error! Bookmark not defined.</b>
Background.....	<b>Error! Bookmark not defined.</b>
Problem Statement .....	2
Research Questions .....	3
Summary.....	4
II. Literature Review .....	6
Cost Estimating Relationships (CERs) .....	6
Learning Curve Theory .....	6
Post-World War II Aircraft Production.....	10
Boone’s Learning Curve .....	12
Summary.....	15
III. Methodology .....	17
Data Collection .....	17
Data Characteristics .....	17
Research Design.....	20
Summary.....	22
IV. Analysis and Results .....	23
Analysis .....	23
Results .....	23
Summary.....	29
V. Conclusions and Recommendations.....	31
Conclusions .....	31

Limitations.....	32
Recommendations.....	33
Appendix.....	36
Bibliography .....	46



## List of Figures

	Page
Figure 1. Wright's CUMAV in C-17A Aircraft Production .....	3
Figure 2. Crawford's ULC in C-17A Aircraft Production .....	3
Figure 3. Wright's v. Boone's CUMAV in T-1A Aircraft Production .....	4
Figure 4. Crawford's v. Boone's ULC in T-1A Aircraft Production .....	4
Figure 5. Regression Analysis (ICEAA Module 8, 2014).....	7
Figure 6. Mean Relative Error Confidence Intervals (Flyaway WBS) .....	20
Figure 7. Mean Relative Error Confidence Intervals (Hardware WBS) .....	21
Figure 8. Mean Relative Error Confidence Intervals with $a = A1$ (Flyaway WBS).....	22
Figure 9. Mean Relative Error Confidence Intervals with $a = A1$ (Hardware WBS).....	23
Figure 10. Mean Relative Error Confidence Intervals Before/Never Merger or Acquisition (Flyaway WBS).....	25
Figure 11. Mean Relative Error Confidence Intervals After Merger or Acquisition (Flyaway WBS).....	25
Figure 12. Mean Relative Error Confidence Intervals Before/Never Merger or Acquisition (Hardware WBS).....	26
Figure 13. Mean Relative Error Confidence Intervals After Merger or Acquisition (Hardware WBS).....	26

## List of Tables

	Page
Table 1: Learning Curve Models .....	2
Table 2: Sample Programs by Mission Type.....	15
Table 3: Sample Programs by Contractor.....	15
Table 4. Mean Relative Errors (Flyaway WBS).....	20
Table 5. Mean Relative Errors (Hardware WBS).....	20
Table 6. Mean Relative Errors with $a = A1$ (Flyaway WBS).....	21
Table 7. Mean Relative Errors with $a = A1$ (Hardware WBS).....	22
Table 8. Mean Relative Errors Before/Never Merger or Acquisition (Flyaway WBS)....	23
Table 9. Mean Relative Errors After Merger or Acquisition (Flyaway WBS).....	24
Table 10. Mean Relative Errors Before/Never Merger or Acquisition (Hardware WBS)	24
Table 11. Mean Relative Errors After Merger or Acquisition (Hardware WBS).....	24



# AN ANALYSIS OF UNCERTAINTY IN LEARNING CURVE MODELS

## I. Introduction

### **Motivation**

Uncertainty in cost estimates coincides with phrases such as "one million dollars, plus or minus ten percent." Cost analysts try to quantify the precision (or more likely imprecision) of their cost estimates using uncertainty (ICEAA Module 9, 2014).

Uncertainty is a critical component of decision analysis. Cost growth in Department of Defense (DOD) weapon system programs compels cost analysts to quantify uncertainty accurately to support decision-making.

Extrapolation of actual past or current costs introduces uncertainty into cost analysis. The expected costs for new programs may be based on information a decade or even half-century old. Cost data is based on a contractor's best estimate of direct labor hours used in production (Alchian, 1963). The DOD's cost estimation field also encounters small sample sizes. Cost analysts are lucky to have more than ten historical programs to choose from to increase reliability and credibility in their estimates (Mislick & Nussbaum, 2015). Cost estimation is the art and science of making meaningful estimates based on imperfect and insufficient data.

### **Background**

A learning curve is a popular method of extrapolating actual cost data to estimate the production cost of future end-items. An end-item is the final production product for an intended use (DAU Glossary, 2021). Learning is based on the fact that workers will gain efficiencies as they build more of the same end item over time. Table 1 lists a

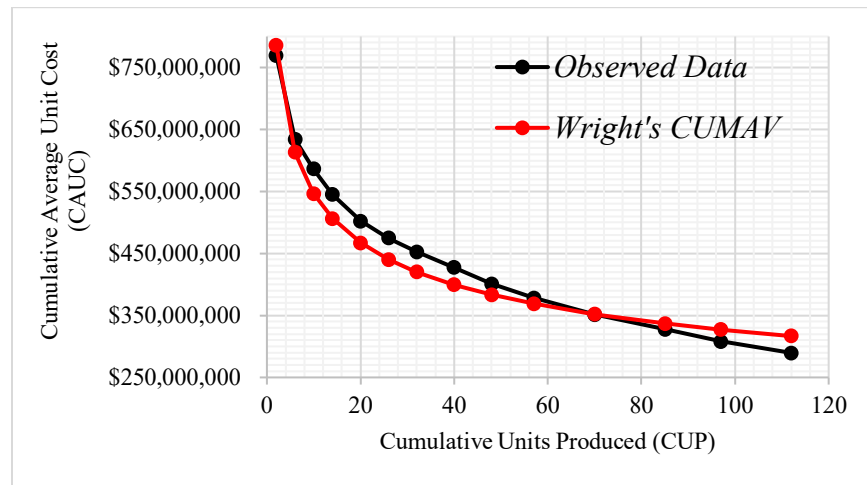
description of four learning curve models that predict the unit cost of an end item given the item's sequential unit number.

<i>Theory Name</i>	Cumulative Average (CUMAV)	Unit Learning Curve (ULC)	Boone's Cumulative Average	Boone's Unit Learning Curve
<i>Credited to</i>	T.P. Wright (1936)	J.R. Crawford (1944)	E.R. Boone (2018)	E.R. Boone (2018)
<i>f(x)</i>	$Y = aX^b$	$Y = aX^b$	$Y = aX^{\frac{b}{(1+\frac{X}{c})}}$	$Y = aX^{\frac{b}{(1+\frac{X}{c})}}$
<i>Independent Variable</i>	Cumulative Number of Units Produced	Sequential Unit Number of Unit Being Calculated	Cumulative Number of Units Produced	Sequential Unit Number of Unit Being Calculated
<i>Dependent Variable</i>	Cumulative Average Cost of $X$ Units (CAUC)	Cost of the $X$ th Unit	Cumulative Average Cost of $X$ Units (CAUC)	Cost of the $X$ th Unit

**Table 1: Learning Curve Models**

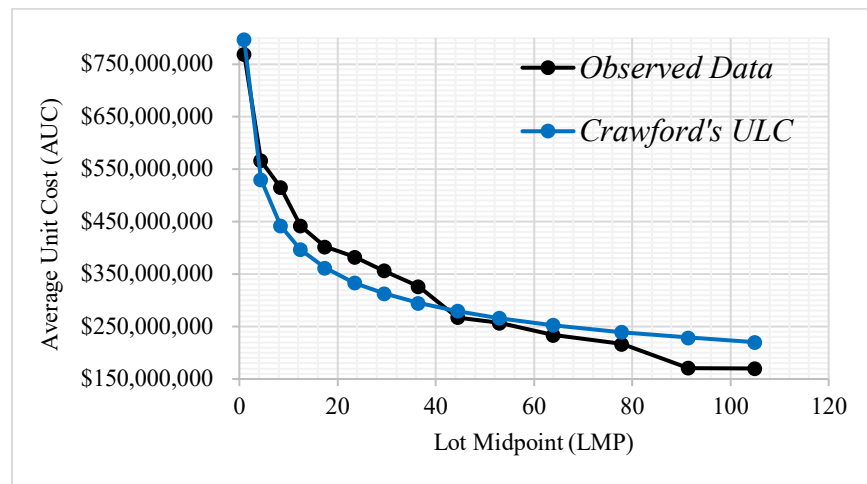
Wright's Cumulative Average (CUMAV) (1936) is the most popular learning curve model used in DOD cost estimation (ICEAA Module 7, 2014). Wright's CUMAV is much easier to use than Crawford's Unit Learning Curve (ULC) (1944) when working with lot data. Often a program's costs are only accounted for in lot data instead of unit data (ICEAA Module 7, 2014). Lot data only has the first and last unit numbers and hence the total number of units and total cost of each batch (i.e., lot) of units. The cumulative number of units produced (CUP) is the independent variable in Wright's CUMAV. The dependent variable in Wright's CUMAV is the cumulative average cost of  $X$  units, also known as the cumulative average unit cost (CAUC). CAUC is the summation of unit costs divided by the total units (ICEAA Module 7, 2014). Crawford's ULC uses lot midpoint (LMP) as its independent variable with lot data. LMP is the unit at which the lot's average cost occurs (ICEAA Module 7, 2014). The average unit cost

(AUC) corresponds to the cost at the lot midpoint. Figure 1 illustrates Wright's CUMAV modeling actual (i.e., observed) costs in the C-17A aircraft production program.



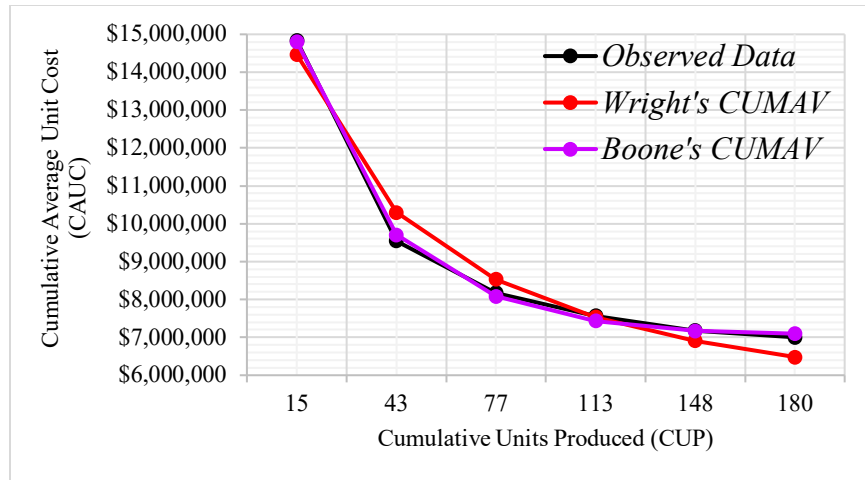
**Figure 1: Wright's CUMAV in C-17A Aircraft Production**

Wright's CUMAV has a “smoother” better fit than Crawford's ULC using lot data. Figure 2 depicts Crawford's ULC modeling actual costs for the same program.



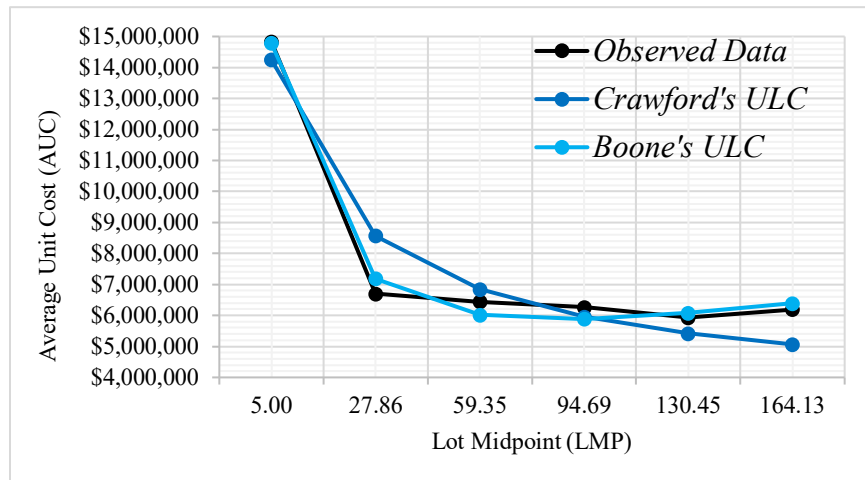
**Figure 2: Crawford's ULC in C-17A Aircraft Production**

Boone's Cumulative Average (CUMAV) is typically more accurate at modeling actual data than other learning curve models. Figure 3 shows how Boone's CUMAV compares to Wright's CUMAV in the T-1A aircraft production program.



**Figure 3: Wright's v. Boone's CUMAV in T-1A Aircraft Production**

Boone's Unit Learning Curve (ULC) is usually more accurate than Crawford's ULC but not as accurate as Wright's CUMAV and Boone's CUMAV. Figure 4 shows how Boone's ULC compares to Crawford's ULC for the same program.



**Figure 4: Crawford's v. Boone's ULC in T-1A Aircraft Production**

## Problem Statement

Cost analysts strive to increase confidence in their estimates by providing a ballpark estimate. A ballpark estimate provides a potential range of costs based on

confidence intervals (ICEAA Module 2, 2014). A confidence interval (CI) suggests the range that contains the estimated mean cost. A significance level, characterized by an alpha  $\alpha$ , indicates the confidence level of an estimate.

$$\text{Confidence Level} = (1 - \alpha) * 100\%$$

The most common confidence interval used in DOD cost estimation is 95%, which corresponds to  $\alpha = 0.05$  (ICEAA Module 8, 2014). The Weapon Systems Acquisition Reform Act of 2009 established a requirement to justify selecting any confidence level less than 80% in establishing a baseline cost estimate for major defense acquisition programs or major automated information system programs (Public Law 111-23, 2009). A consistent and appropriate application of confidence intervals improves a cost estimate's overall quality.

## Research Questions

Ordinary least squares (OLS) is the standard regression technique used to obtain the optimal values for  $a$  and  $b$  of the traditional learning curve models (i.e., Wright's CUMAV and Crawford's ULC). Although  $Y = aX^b$  is a nonlinear function, it can be transformed into a linear function via logarithms:

$$Y = aX^b$$

$$\ln Y = \ln a + b \ln X$$

A nonlinear solver (NLS) is a regression technique that can determine optimal values for  $a$  and  $b$  without a logarithmic transformation of  $Y = aX^b$ . The traditional learning curve models may fit observed data better with NLS than OLS.



1. How does the applied regression technique (i.e., OLS or NLS) affect the accuracy of Wright's CUMAV and Crawford's ULC?

The theoretical first unit cost (T1) is  $a$  in  $Y = aX^b$ . T1 can vary depending on the optimal value that a regression technique assigns to  $a$ . The actual first unit cost (A1) may minimize any statistical advantage CUMAV theory learning curve models have over ULC theory learning curve models using lot data. Regression techniques would only solve for  $b$  ( $b$  and  $c$  in the case of Boone's learning curve models).

2. How does the parameter  $a$  affect a learning curve model's accuracy?

A major defense contractor (i.e., Boeing, Lockheed-Martin, McDonnell-Douglas, etc.) does not have all of its skilled workers and manufacturing facilities located in one region of the world (*Boeing.com*, 2020). Acquisitions and mergers in the aircraft industry spread skilled labor across manufacturing facilities. The methods of estimating direct labor hours varied among manufacturing facilities of the same production program (Alchian, 1963). The learning curve models may be less accurate at modeling learning in production programs completed by a contractor after an acquisition or merger.

3. How does an acquisition or merger affect a learning curve model's accuracy?

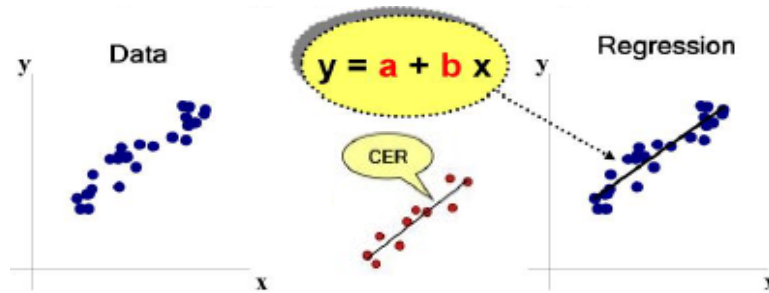
## Summary

Chapter I introduced the importance of providing confidence intervals to gauge uncertainty in cost estimates. Cost data is usually inconsistent and insufficient. Cost analysts should recognize the limitations and advantages of one learning curve model over another. Chapter II reviews relevant literature in learning curve theory.

## II. Literature Review

### Cost Estimating Relationships (CERs)

Four established cost estimating techniques discern patterns in historical data. One technique, parametric, has an advantage over other cost estimating techniques for capturing uncertainty using statistics (ICEAA Module 2, 2014). A parametric technique is based on the concept of using a statistical relationship, called a cost estimating relationship (CER). Regression analysis constructs CERs by providing a best-fit equation and measures of statistical significance and uncertainty (ICEAA Module 8, 2014).



**Figure 5: Regression Analysis (ICEAA Module 8, 2014)**

Cost analysts use regression-based CERs to estimate the future costs of new systems. The regression equation's intercept and slope propagate uncertainty in a CER. A confidence interval quantifies uncertainty and expresses the degree of precision (or imprecision) in the intercept and slope or  $y$  value at a given  $x$ . For example, if adding more data points changes the intercept's value significantly, the intercept likely has a wide confidence interval.

### Learning Curve Theory

A learning curve is a regression-based cost estimating relationship that uses actual costs from a past or current end item to predict future costs for the same end item

(ICEAA Module 2, 2014). T.P. Wright (1936) and J.R. Crawford (1944) expressed the learning curve effect in a simple power function:  $Y = aX^b$

$Y$  represents the cumulative average cost of  $X$  units in Wright's CUMAV and the  $X$ th unit's cost in Crawford's ULC. Performing ordinary least squares (OLS) regression obtains the learning curve's parameters  $a$  and  $b$ . OLS regression requires transforming the learning curve's power function logarithmically to a linear function in log-space before determining the regression's best-fit equation.

$$Y = aX^b$$

$$\ln Y = \ln a + b \ln X$$

OLS regression acquires values for the intercept and slope, which provide the values of  $a$  and  $b$ . Euler's number  $e$  raised to the intercept defines  $a$ .  $b$ 's value remains unchanged from log-space to unit-space.

$$a = e^{\ln a}$$

$$b = \log_2 LCS$$

$a$  is the theoretical first unit cost (T1), which should not be confused with the actual first unit cost (ICEAA Module 7, 2014). When fitting historical data in regression analysis, the regression's best-fit line will likely not pass through any data points. The learning rate is the complement of the learning curve slope (LCS). The learning curve slope is 2 raised to  $b$ .

$$LCS = 2^b$$

$$b = \log_2 LCS = \frac{\ln(LCS)}{\ln(2)}$$

The learning curve slope is a fraction between 0 and 1. If  $LCS = 1$ , a 100% learning curve slope, a worker would spend as much time manufacturing or assembling their 100<sup>th</sup> unit as the 1<sup>st</sup> unit. The base 2 logarithm of a fraction between 0 and 1 is a negative number. Often  $b$  will vary from -6.644 to -0.0144, corresponding to a 1% to 99% decrease in the cost or labor hours every time  $X$  doubles.<sup>1</sup> The learning curve slope garners more attention than the theoretical first unit cost in learning curve analysis. A projected learning curve slope often has the biggest effect on the production portion of a life cycle cost estimate (ICEAA Module 7, 2014). Cost analysts need to be confident when justifying a learning curve slope.

### **Post-World War II Aircraft Production**

Alchian (1950) observed learning rates across different mission-types (i.e., bomber, fighter, trainer, and transport) in airframe manufacturing. After World War II, learning curves in the aircraft production industry expressed a relationship between the labor hours required to produce an airframe and the  $X$ th airframe produced with a weight factor. The weight factor was the labor hours expended per pound of an airframe for the  $X$ th airframe. Learning curves were more accurate at predicting labor requirements for lighter airframes (i.e., fighters and trainers) than heavier airframes (i.e., bombers and transports). However, learning curves varied across manufacturing facilities producing the same airframe. Alchian noticed nonconstant, diminishing learning rates by convexity in log-space graphs of labor hours used and units produced in airframe production. Log-space makes it easier to detect nonconstant learning rates than unit-space since the

---

<sup>1</sup> Program costs are often reported in labor hours as opposed to dollars.

relationship between labor hours (or costs) and units produced should be linear in log-space. Different methods of calculating labor hours across manufacturing facilities may have contributed to the diminishing learning rates.

Asher (1956) proposed causes of diminishing learning rates in his study of post-World War II fighter aircraft. The U.S. Air Force contracted him to investigate why some programs were being underestimated at the end of production. He explained how aggregated learning curves relate to diminishing learning rates by examining learning curves in log-space. Non-aggregated learning curves were linear, but a combination of two learning curves could cause convexity in log-space.

Learning curves are aggregated at different levels. A culmination of several constituent learning curves is present in an aggregated learning curve. At the lowest level of aggregation is the factory worker, who has their own learning curve. One level higher is a team of workers. The next level might be the department. The highest level is the learning curve for an entire organization across all departments. Diminishing learning stems from an aggregation of learning curves. All workers, teams, and departments will have different learning rates that will produce different learning curves.

Asher showed how combining different learning curves into an aggregated learning curve will result in an asymptote or diminishing learning rate at some point in the learning curve. He found diminishing learning rates to be common near the end of production after many units. However, he could not find a common value for the number of units where diminishing learning occurs for most departments. The point at which diminishing learning occurred depended on the complexity of the line of work. The welding department may have reached an asymptote at 500 units while sheet metal

reached their asymptote at 1000 units. He concluded that constant learning rates are inaccurate, especially when a program has produced many units.

### **Boone's Learning Curve**

Boone (2018) proposed a more complex power function that utilizes Wright's CUMAV and Crawford's ULC. Boone's power function is  $Y = aX^{\frac{b}{1+\frac{X}{c}}}$  and it is intrinsically nonlinear. Unlike the simple power function,  $Y = aX^b$ , no transformation will make Boone's function linear.  $c$  is impossible to obtain using OLS regression. Boone incorporated  $c$  in an attempt to account for diminishing learning rates.  $c$  only varies between 0 to 5000 because any value greater than 5000 did not significantly impact error reduction, and 5000 was a binding constraint in the nonlinear solver for several data observations (Boone, 2018).

Boone (2018) only compared his model to Wright's CUMAV. Hogan (2020) compared Boone's ULC to Crawford's ULC. He researched what proportion of observations were more accurately explained by Boone's learning curve than the traditional learning curves (i.e., Wright's CUMAV and Crawford's ULC). He defined percentage error differences that showed significant error reductions using Boone's learning curve. 93% of observations had about the same error or significant error reductions using Boone's learning curve over the traditional learning curves. Therefore, Boone's learning curve performs as well as if not better than the traditional learning curves for 93% of observations. 42% of observations were significantly better explained by Boone's learning curve, while 51% of observations had an approximately equal error. Boone's learning curve was less accurate than the traditional learning curves for only 7%

of observations.<sup>2</sup> The amount of error reduction one can expect from Boone's learning curve was unknown because the descriptive statistics indicated a highly variable error reduction from Boone's learning curve. As a result, Hogan investigated program attributes (i.e., how many units the program produced, aggregated learning curves, and the production rate) to predict the degree of error improvement one can expect from Boone's learning curve. He tried to discern which program attributes contributed to Boone's learning curve model explaining diminishing learning rates. However, he did not find any statistically significant relationships but noted program attributes as an area of future research.

## **Summary**

Chapter II provided a brief explanation of how cost analysts extrapolate historical costs and create a regression-based CER, such as a learning curve. The discovery of nonconstant, diminishing learning rates in post-World War II defense aircraft production programs led to analysts creating alternative learning curves. Boone (2018) amended the traditional learning curves' general form (i.e., Wright's CUMAV and Crawford's ULC) to account for diminishing learning rates. Chapter III outlines the methodology for further analysis of Wright's CUMAV, Crawford's ULC, Boone's CUMAV, and Boone's ULC.

---

<sup>2</sup> The nonlinear solver (NLS) used to determine Boone's learning curve parameters may have contributed to Boone's learning curve being less accurate than the traditional learning curves for 7% of observations.

### **III. Methodology**

#### **Data Collection**

Air Force Life Cycle Management Center (AFLCMC) extracted quantitative data from CADE-DCMS and cost reports 1921 and 1921-1. AFLCMC transferred relevant data for learning curve analysis to a spreadsheet. The data included direct recurring costs in constant year (CY) dollars at the flyaway and prime mission equipment work breakdown structure (WBS) levels. When developing cost estimating relationships, a cost analyst must represent historical costs in constant year dollars to normalize inflation (ICEAA Module 5, 2014). The spreadsheet had included all appropriate calculations of lot midpoint and average unit cost. However, it required additional calculations for cumulative units produced and cumulative average unit cost calculations for lot data.

#### **Data Characteristics**

The data included two WBS levels: Total Recurring Flyaway Costs Less Engine and Total Recurring Hardware (Prime Mission Equipment) Costs. Flyaway costs contain the “cost of the air vehicle, including the airframe, installed engines, electronics, airborne armament, fire control systems, engineering changes, and a prorata share of fabrication tooling and other production startup costs” (ICEAA CEBOK Glossary, 2013). Engineering changes may indicate indirect costs that cannot be modeled with learning curve theory. Only direct costs from manufacturing or assembling physical end items experience learning (Wright, 1936). Prime mission equipment is equipment intended to perform the mission of a program but not “test equipment, spares, or test articles” (ICEAA CEBOK Glossary, 2013). Flyaway costs contain hardware costs, but the



hardware costs may represent a greater proportion of recurring costs directly attributable to the end item.

The original spreadsheet from AFLCMC included lot data of 138 DOD weapon system production programs of complex end-items. Lot data does not include the detailed costs of producing each unit of a program. However, learning curve analysis is possible with an appropriate sample of lot data. The following criteria limited the sample to 49 of the initial 138 production programs:

- 1) Program must have 5 or more consecutive lots.
- 2) Program must have data for its first lot.

Criterion 1 is consistent with previous learning curve analyses and prevents overfitting of the data (Boone, 2018).<sup>3</sup> A requirement of 6 or more consecutive lots would decrease the sample size to 40 programs and limit the diversity of programs attributes (i.e., mission type or contractor) available to the analysis. Criterion 2 is required based on the nature of Wright's CUMAV. The cumulative average unit cost is a function of previous lots' costs and quantities.

$$CAUC \text{ for Lot } 1 = \frac{\text{Lot } 1's \text{ Total Cost}}{\text{Lot } 1's \text{ Quantity}}$$

$$CAUC \text{ for Lot } 2 = \frac{\text{Lot } 1's \text{ Total Cost} + \text{Lot } 2's \text{ Total Cost}}{\text{Lot } 1's \text{ Quantity} + \text{Lot } 2's \text{ Quantity}}$$

The sampled 49 programs had 8 lots on average. Fighter aircraft represented nearly a third of the sample. Fighter aircraft had an average of 10 lots. Tables 2 lists the number of programs by mission type.

---

<sup>3</sup> Boone (2018) and Hogan (2020) used the same criterion.

Fighter	15	31%
Attack	7	14%
Electronic Warfare	7	14%
Cargo	5	10%
Bomber	3	6%
Special Operations	3	6%
Trainer	3	6%
Maritime Patrol	2	4%
Reconnaissance	2	4%
Tanker	1	2%
Utility	1	2%

**Table 2: Sample Programs by Mission Type**

Boeing represented the largest share of programs in the sample. Boeing merged with McDonnell-Douglas in 1997 (Amir & Weiss, 2020). McDonnell Aircraft acquired Douglas Aircraft in 1967 (*Britannica*, 2020). 18/49 of the sampled programs were completed by a contractor that merged or acquired another major contractor in the aircraft manufacturing industry.

Boeing	10	20%
McDonnell-Douglas	6	12%
Lockheed Corporation	5	10%
Lockheed-Martin	5	10%
Douglas Aircraft	4	8%
Grumman Aerospace	3	6%
Northrop-Grumman	3	6%
McDonnell Aircraft	2	4%
Raytheon	2	4%
Bell	1	2%
Fairchild Republic	1	2%
General Dynamics	1	2%
LTV Aerospace	1	2%
Boeing-McDonnell-Douglas	1	2%
McDonnell-Douglas-British Aerospace	1	2%
North American-Rockwell-Collins	1	2%
Northrop	1	2%
Sikorsky Aircraft	1	2%

**Table 3: Sample Programs by Contractor**

## Research Design

The Python programming language hosts various libraries of functions for scientific analysis. SciPy is an extensive library of functions in Python. *curve\_fit* is a simple nonlinear solver (NLS) function from SciPy that fits models to data and determines a model's parameters. The first step to determine the slope and intercept of the best-fit line in the sample data is to define the learning curve functions. The traditional learning curve function:

```
def func(fx,a,b):  
    return a*fx**b
```

Boone's learning curve function:

```
def func(fx,a,b,c):  
    return a*fx**(b/(1+(fx/c)))
```

The second step is to call *curve\_fit* to find the best-fit parameters using a least-squares fit. *curve\_fit* returns two variables, *popt* and *pcov*. *popt* contains the best-fit parameters for *a* and *b* (*b* and *c* in the case of Boone's learning curve). *pcov* contains the covariance matrix, which indicates the uncertainties and correlations between parameters.

```
popt, pcov = curve_fit(func, x, y)
```

*curve\_fit* required additional arguments to optimize Boone's learning curve function. *maxfev* set a limit to the number of calls to *curve\_fit* performing "trial-and-error" to obtain the optimal parameter values. Bounds provided necessary constraints to guide *curve\_fit* to valid parameter values. The largest actual first unit cost (A1) of the sampled programs was just under \$1B. The sample's cost data was in dollars (thousands). \$2,000,000 (thousands) provided a reasonable upper bound for *curve\_fit* to determine the

theoretical first unit cost's (T1) parameter. The learning curve slope (LCS) for any sampled program would be greater than 1% but less than 99%. Thus,  $b$  was allowed to vary between -6.644 and -0.0144, corresponding to 1% and 99% LCS, respectively. Lastly, Boone's decay variable needed bounds defined from 0 to 5000.

```
popt, pcov = curve_fit(func, x, y,
maxfev=50000, bounds=([0,-6.644,0],[2000000,-0.0144,5000]))
```

A logarithmic transformation to the traditional learning curve function allows an ordinary least squares (OLS) to find the best-fit parameters using a least-squares fit.

```
def logfunc(fx,a,b):
    return np.log(func(fx,a,b))
```

OLS regression returns an intercept and coefficient value, which provide the values of  $a$  and  $b$ .

```
df = pd.DataFrame({'x':func(x, *popt), 'y':y})
fit_lm = smf.ols('y~x', data=df).fit()
```

The last step was the calculation of relative errors. The relative error is used to compare estimation methods on data sets of widely differing sizes. For example, the C-17A aircraft production program's costs ranged from \$150M - \$750M, but the T-1A aircraft production program's costs ranged from \$4M - \$15M. The T-1A's costs were a small ratio of the C-17A's costs. Therefore, the costs were on a ratio scale. The relative error is used in measurements that have an absolute zero. \$0 is the absolute zero in costs. *relerr* and *avgrel* are arbitrarily-named variables representing the calculations of relative errors at each data point of a program modeled by the learning curves and the relative errors' mean.

```

relerr = abs(1 - y / func(x, *popt))

print("Relative Errors:", relerr * 100)

avgrel = sum(relerr) / len(y) * 100

print("Mean Relative Error:", avgrel)

```

The mean relative error for Wright's CUMAV, Crawford's ULC, Boone's CUMAV, and Boone's ULC were recorded for later use in statistical analysis.

## Summary

Chapter III described the data's population and sample used for an adequate learning curve analysis. Python allowed for a nonlinear solver (NLS) and an ordinary least squares (OLS) regression technique to obtain parameter values and the learning curve models' relative errors. Chapter IV answers research questions from Chapter I by examining confidence intervals and determining statistically significant differences.

## IV. Analysis and Results

### Analysis

The statistical analysis consisted of determining mean relative error confidence intervals for each learning curve model and regression technique to answer the research questions. Any overlap between mean relative error confidence intervals suggested a lack of sufficient evidence that statistically significant differences exist in a direct comparison of two or more learning curve models. The mean relative error confidence intervals used the Student's t-distribution because the sample size was small (ICEAA Module 10, 2014).

The margin of error is a calculation for the upper and lower bounds of a confidence interval.

$$\text{Margin of Error} = z * \frac{\sigma}{\sqrt{n}} \quad z = 1.96 \quad n = 49$$

$z$  represents the  $z$ -value determined by the confidence interval.  $\sigma$  represents the standard deviation of each sampling distribution for the sample mean.  $n$  represents the sample size of 49 production programs.

### Results

Tables 4 and 5 list results that answered the first research question:

1. How does the applied regression technique (i.e., OLS or NLS) affect the modeling accuracy of Wright's CUMAV and Crawford's ULC?

Tables 4 and 5 list mean relative error descriptive statistics for each learning curve model and regression technique. Each table represents data from a separate work breakdown structure (WBS) level.

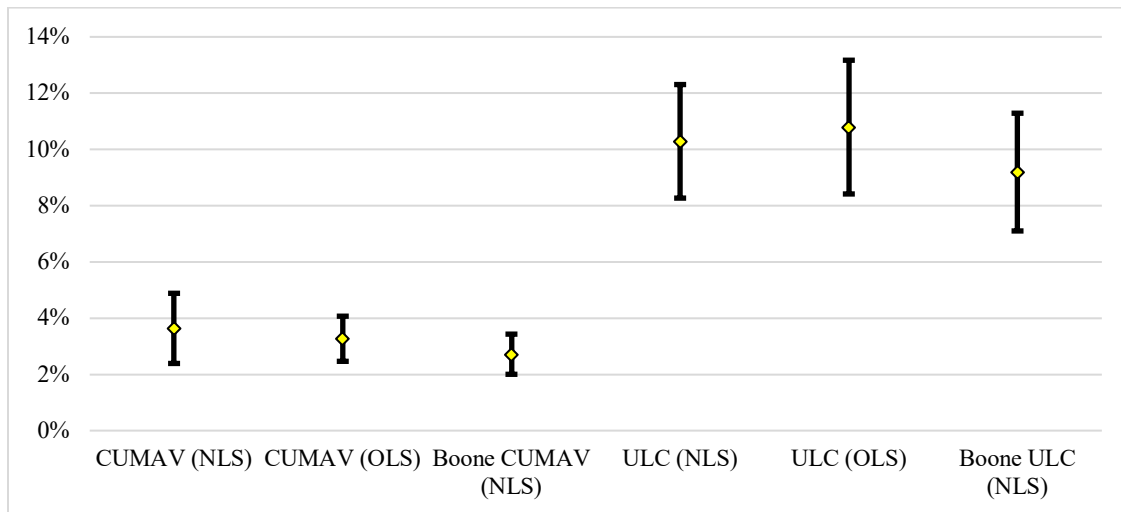
Learning Curve Theory & Regression Technique	Mean	Standard Deviation	Margin of Error	Lower Bound	Upper Bound
CUMAV (NLS)	3.635%	4.374%	1.256%	2.378%	4.891%
CUMAV (OLS)	3.266%	2.846%	0.817%	2.449%	4.083%
Boone CUMAV (NLS)	2.716%	2.519%	0.724%	1.993%	3.440%
ULC (NLS)	10.286%	7.142%	2.051%	8.235%	12.338%
ULC (OLS)	10.792%	8.351%	2.399%	8.394%	13.191%
Boone ULC (NLS)	9.193%	7.292%	2.094%	7.098%	11.287%

**Table 4: Mean Relative Errors (Flyaway WBS)**

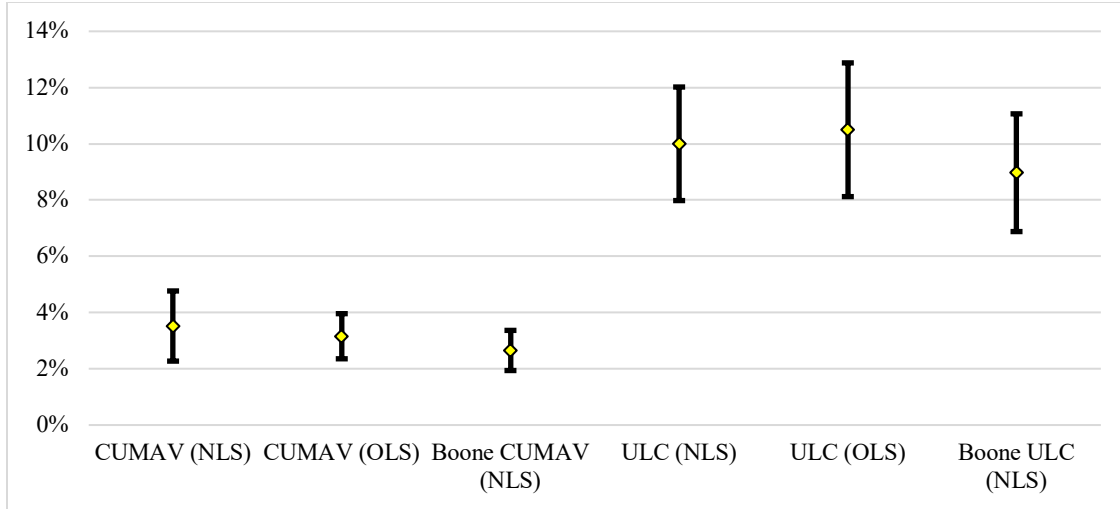
Learning Curve Theory & Regression Technique	Mean	Standard Deviation	Margin of Error	Lower Bound	Upper Bound
CUMAV (NLS)	3.521%	4.342%	1.247%	2.273%	4.768%
CUMAV (OLS)	3.158%	2.793%	0.802%	2.356%	3.960%
Boone CUMAV (NLS)	2.652%	2.485%	0.714%	1.938%	3.366%
ULC (NLS)	10.001%	7.029%	2.019%	7.982%	12.020%
ULC (OLS)	10.500%	8.278%	2.378%	8.122%	12.878%
Boone ULC (NLS)	8.973%	7.288%	2.093%	6.880%	11.066%

**Table 5: Mean Relative Errors (Hardware WBS)**

CUMAV theory learning curve models were statistically more accurate than ULC theory models. Figures 6 and 7 illustrate the statistically significant differences between CUMAV and ULC theory models.



**Figure 6: Mean Relative Error Confidence Intervals (Flyaway WBS)**



**Figure 7: Mean Relative Error Confidence Intervals (Hardware WBS)**

As shown in Tables 4 and 5 and Figures 6 and 7, the mean relative errors are not statistically different between OLS and NLS regression techniques. Moreover, the confidence intervals are wider with NLS, which indicates more uncertainty.

Tables 6 and 7 list results that answered the second research question:

2. How does the parameter  $a$  affect a learning curve model's accuracy?

Tables 6 and 7 list mean relative error descriptive statistics for each learning curve model and regression technique with  $a$  as the actual first unit cost (A1) instead of the theoretical first unit cost (T1). Each table represents data from a separate work breakdown structure (WBS) level.

Learning Curve Theory & Regression Technique	Mean	Standard Deviation	Margin of Error	Lower Bound	Upper Bound
CUMAV (NLS)	11.296%	9.455%	2.716%	8.581%	14.012%
CUMAV (OLS)	10.666%	9.162%	2.632%	8.035%	13.298%
Boone CUMAV (NLS)	10.499%	8.414%	2.417%	8.082%	12.916%
ULC (NLS)	19.322%	21.239%	6.101%	13.221%	25.423%
ULC (OLS)	16.802%	14.324%	4.114%	12.688%	20.916%
Boone ULC (NLS)	15.937%	11.927%	3.426%	12.511%	19.362%

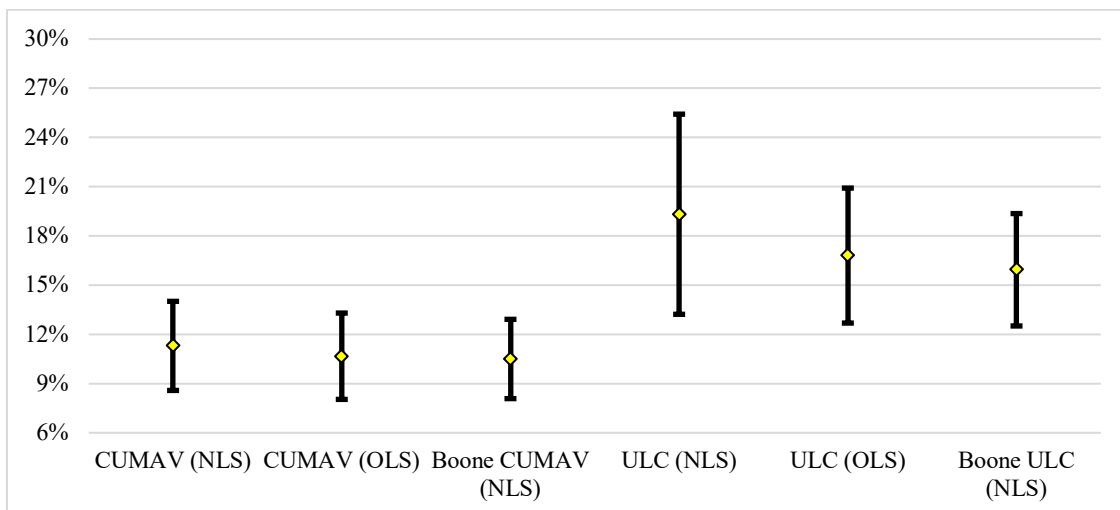
**Table 6: Mean Relative Errors with  $a = A1$  (Flyaway WBS)**



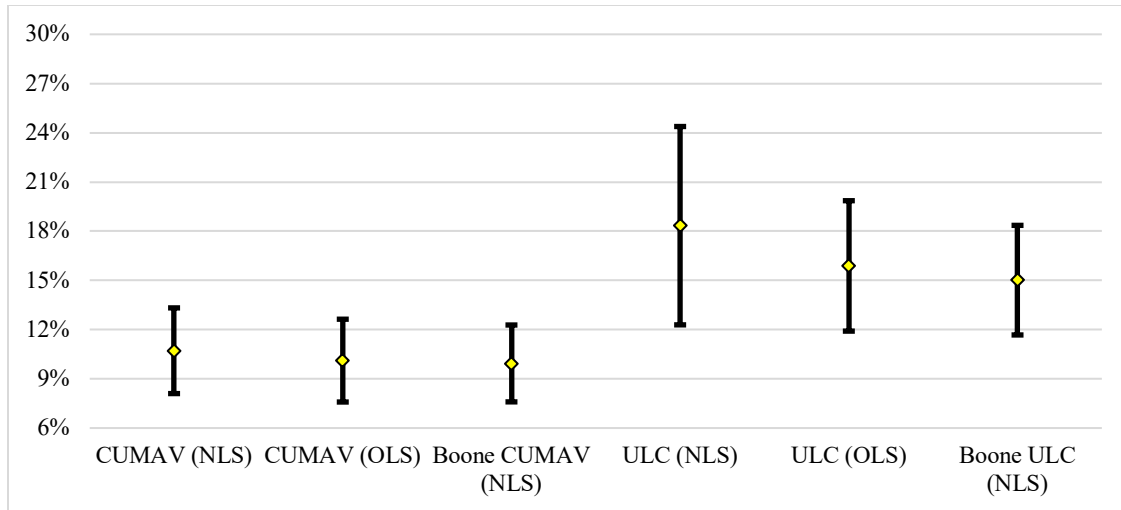
Learning Curve Theory & Regression Technique	Mean	Standard Deviation	Margin of Error	Lower Bound	Upper Bound
CUMAV (NLS)	10.712%	9.094%	2.612%	8.100%	13.324%
CUMAV (OLS)	10.111%	8.787%	2.524%	7.587%	12.634%
Boone CUMAV (NLS)	9.938%	8.157%	2.343%	7.595%	12.281%
ULC (NLS)	18.335%	21.047%	6.046%	12.290%	24.381%
ULC (OLS)	15.883%	13.838%	3.975%	11.908%	19.858%
Boone ULC (NLS)	15.015%	11.631%	3.341%	11.674%	18.355%

**Table 7: Mean Relative Errors with  $a = A1$  (Hardware WBS)**

For flyaway and hardware (PME) WBS levels, every learning curve model except for Crawford's ULC with OLS was statistically less accurate at modeling actual data when  $a$  was the actual first unit cost (A1). CUMAV theory models did not have statistically significant different mean relative errors with ULC theory models when  $a$  was the actual first unit cost. Figures 8 and 9 show an overlap between the upper bounds of CUMAV theory model confidence intervals and the lower bounds of ULC theory model confidence intervals. The wider confidence intervals when  $a$  was the actual first unit cost could be a technical artifact of the regression techniques constrained to optimizing only  $b$  ( $b$  and  $c$  in the case of Boone's learning curve models).



**Figure 8: Mean Relative Error Confidence Intervals with  $a = A1$  (Flyaway WBS)**



**Figure 9: Mean Relative Error Confidence Intervals with  $a = A1$  (Hardware WBS)**

Tables 8-11 list results that answered research question III:

3. How does an acquisition or merger affect a learning curve model's accuracy?

Table 8 lists mean relative error descriptive statistics for each learning curve model and regression technique for production programs completed by a contractor before an acquisition or merger (or never had an acquisition or merger) at the flyaway WBS level.

Learning Curve Theory & Regression Technique	Mean	Standard Deviation	Margin of Error	Lower Bound	Upper Bound
CUMAV (NLS)	4.225%	5.221%	1.500%	2.726%	5.725%
CUMAV (OLS)	3.721%	3.220%	0.925%	2.796%	4.646%
Boone CUMAV (NLS)	2.982%	2.796%	0.803%	2.179%	3.785%
ULC (NLS)	10.941%	7.861%	2.258%	8.683%	13.199%
ULC (OLS)	11.886%	9.525%	2.736%	9.150%	14.621%
Boone ULC (NLS)	9.805%	8.121%	2.333%	7.473%	12.138%

**Table 8: Mean Relative Errors Before/Never Merger or Acquisition (Flyaway WBS)**

Table 9 lists mean relative error descriptive statistics for each learning curve model and regression technique for production programs completed by a contractor after an acquisition or merger at the flyaway WBS level.

Learning Curve Theory & Regression Technique	Mean	Standard Deviation	Margin of Error	Lower Bound	Upper Bound
CUMAV (NLS)	2.618%	2.045%	0.588%	2.031%	3.206%
CUMAV (OLS)	2.483%	1.884%	0.541%	1.942%	3.024%
Boone CUMAV (NLS)	2.258%	1.944%	0.558%	1.700%	2.817%
ULC (NLS)	9.195%	5.796%	1.665%	7.530%	10.860%
ULC (OLS)	8.910%	5.556%	1.596%	7.314%	10.506%
Boone ULC (NLS)	8.138%	5.646%	1.622%	6.516%	9.759%

**Table 9: Mean Relative Errors After Merger or Acquisition (Flyaway WBS)**

Table 10 lists mean relative error descriptive statistics for each learning curve model and regression technique for production programs completed by a contractor before an acquisition or merger (or never had an acquisition or merger) at the hardware (PME) WBS level.

Learning Curve Theory & Regression Technique	Mean	Standard Deviation	Margin of Error	Lower Bound	Upper Bound
CUMAV (NLS)	4.110%	5.193%	1.492%	2.619%	5.602%
CUMAV (OLS)	3.605%	3.164%	0.909%	2.696%	4.513%
Boone CUMAV (NLS)	2.905%	2.768%	0.795%	2.110%	3.700%
ULC (NLS)	10.726%	7.679%	2.206%	8.520%	12.932%
ULC (OLS)	11.667%	9.422%	2.706%	8.960%	14.373%
Boone ULC (NLS)	9.626%	8.097%	2.326%	7.300%	11.951%

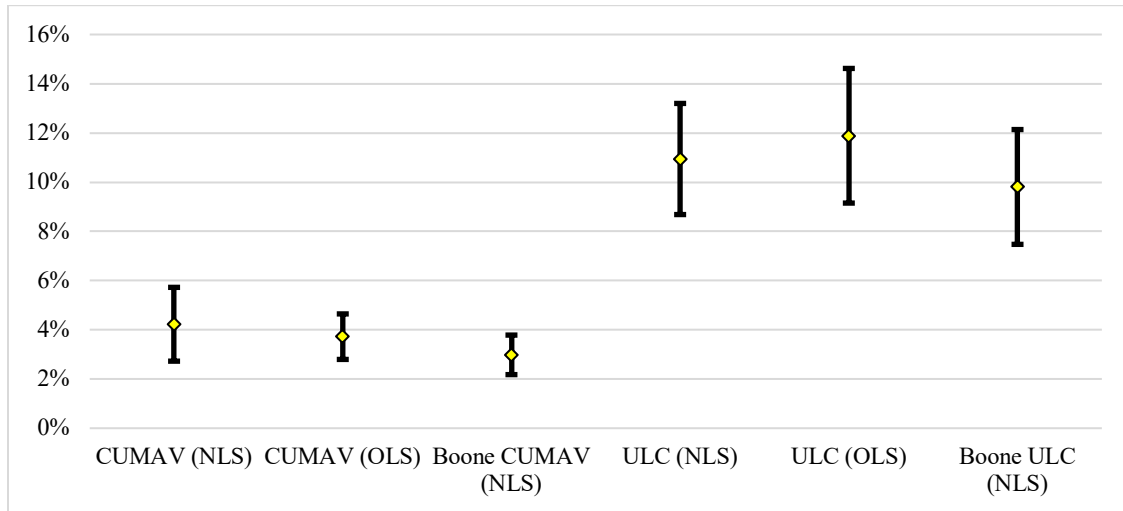
**Table 10: Mean Relative Errors Before/Never Merger or Acquisition  
(Hardware WBS)**

Table 11 lists mean relative error descriptive statistics for each learning curve model and regression technique for production programs completed by a contractor after an acquisition or merger at the hardware (PME) WBS level.

Learning Curve Theory & Regression Technique	Mean	Standard Deviation	Margin of Error	Lower Bound	Upper Bound
CUMAV (NLS)	2.505%	1.981%	0.569%	1.936%	3.074%
CUMAV (OLS)	2.389%	1.836%	0.527%	1.862%	2.917%
Boone CUMAV (NLS)	2.218%	1.898%	0.545%	1.673%	2.763%
ULC (NLS)	8.792%	5.792%	1.664%	7.128%	10.455%
ULC (OLS)	8.491%	5.482%	1.575%	6.916%	10.065%
Boone ULC (NLS)	7.849%	5.670%	1.629%	6.220%	9.477%

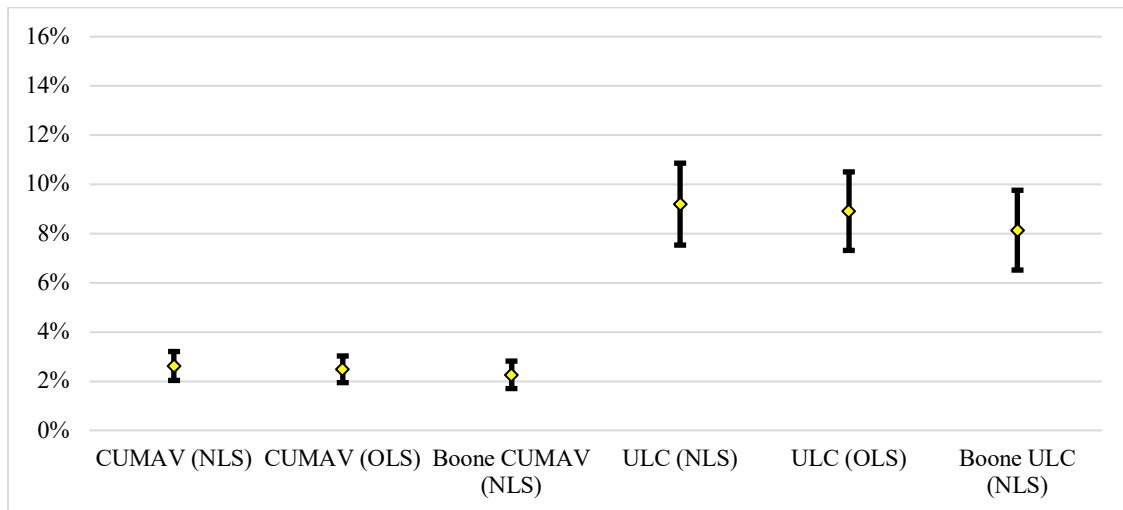
**Table 11: Mean Relative Errors After Merger or Acquisition (Hardware WBS)**

Tables 8-11 and Figures 10-13 indicate no statistically significant mean relative error differences before and after a merger or acquisition.



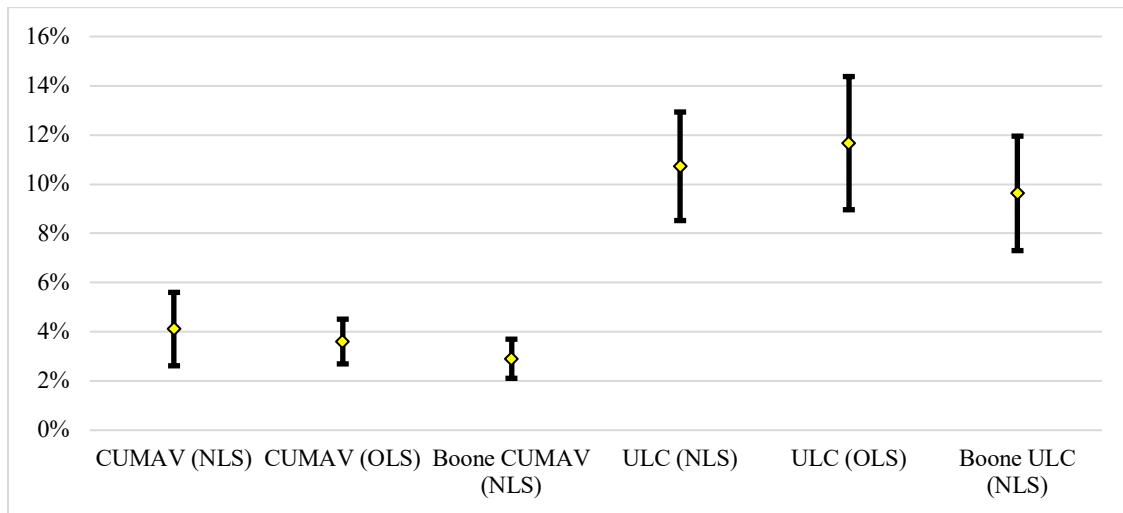
**Figure 10: Mean Relative Error Confidence Intervals Before/Never Merger or Acquisition (Flyaway WBS)**

Dispersing resources and labor should increase uncertainty. However, every confidence interval is narrower after a merger or acquisition, indicating less uncertainty.

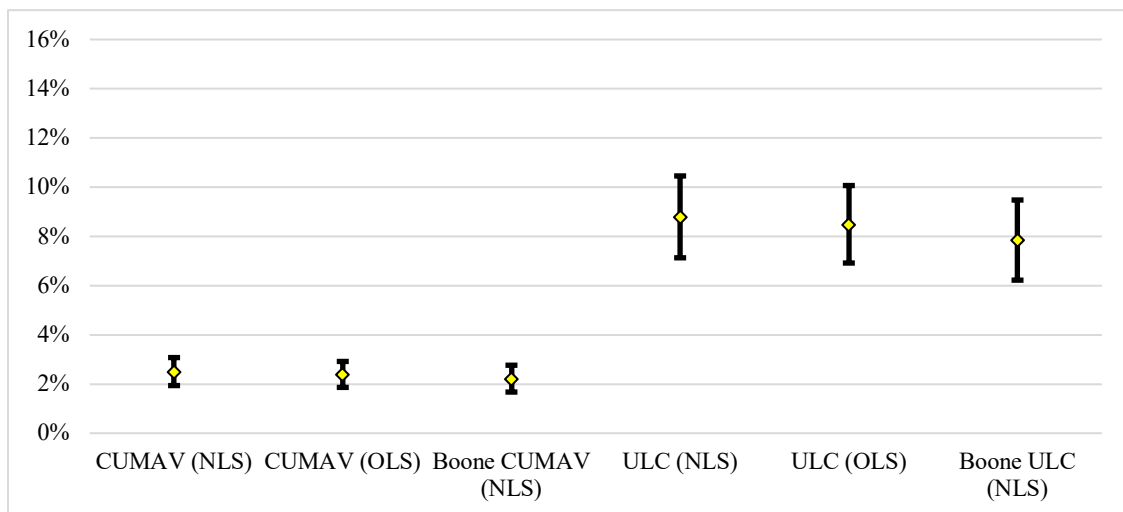


**Figure 11: Mean Relative Error Confidence Intervals After Merger or Acquisition (Flyaway WBS)**

The results for the hardware (PME) WBS level are almost identical to the flyaway WBS level.



**Figure 11: Mean Relative Error Confidence Intervals Before/Never Merger or Acquisition (Hardware WBS)**



**Figure 11: Mean Relative Error Confidence Intervals After Merger or Acquisition (Hardware WBS)**

## **Summary**

Chapter IV reviewed the results of the analysis. Mean relative error measured differences between learning curve model estimates and actual (i.e., observed) data. 95% confidence intervals determined whether the results were statistically significant.

Theories from research questions I and III have no statistically significant impact on learning curve models, but the analysis provided statistical evidence that research question II's theory affects learning curve models. Chapter V recapitulates the research goals and provides recommendations for future research.

## V. Conclusions and Recommendations

### Conclusions

The traditional learning curve models have approximately the same accuracy with either regression technique. Despite a regression technique specifically designed for nonlinear functions, NLS appears to propagate more uncertainty than OLS into the learning curve models. Cost analysts should remain using ordinary least squares (OLS) with the traditional learning curve models.

The actual first unit cost ( $A_1$ ) caused statistically significant decreases in every learning curve model's accuracy except Crawford's ULC with OLS. CUMAV theory models were no longer statistically significantly more accurate than ULC theory models when  $a$  was the actual first unit cost. The cost estimation field should look further into the impacts of using the actual first unit cost instead of the theoretical first unit cost to clear doubts about the CUMAV theory's unfair statistical advantage over the ULC theory (ICEAA Module 7, 2014).

A merger or acquisition did not significantly impact a learning curve model's accuracy. However, the results do not suggest accumulating and spreading resources and labor to various manufacturing facilities increases uncertainty in predicting labor requirements. Every learning curve model's mean relative error confidence interval was narrower for a contractor's production program after a merger or acquisition. In a time of unprecedented speed and innovation in the Air Force acquisition process, the results indicate improved certainty for cost estimates.

## **Limitations**

The traditional learning curve models may be more accurate than Boone's learning curve models for production programs that took less time. Boone's learning curve models might exaggerate cost inefficiencies in shorter production programs that did not occur because of its decay variable  $c$ . Without an industry-wide average for number of lots for aircraft production programs, the analysis did not investigate the impact of a production program's length on a learning curve model's accuracy. The literature did not mention an average number of lots in the aircraft industry. AFLCMC provided lot data of production programs that ranged from Lot 1 to Lot 19. Nearly half of the programs in the sample (25/49) had 7 or fewer lots, while the other half (24/49) had 9 or more lots.

## **Recommendations**

The mission dictates the complexity of systems integrated into an aircraft. Learning rates in the airframe production of a strategic bomber with stealth technology would likely be more difficult to estimate than in a cargo aircraft's airframe production. Future research could expand on Alchian's (1963) research on predicting labor requirements using a learning curve that incorporates an airframe weight factor.

The learning curve in Alchian's (1963) research aggregated onsite and offsite direct labor hours. Onsite direct labor hours are the labor hours recorded from work at the main assembly plant, such as Boeing's primary plant in Puget Sound, WA. Offsite direct labor hours are an estimation of work outside of the plant or "plants of the reporting facility" (Alchian, 1963). Future research could compare the learning curve models using onsite versus offsite direct labor hours.



## Appendix

### Python Analysis

- 1) Import the following packages into your Python coding environment (i.e., Jupyter, Spyder, etc.)

```
import pandas as pd
import seaborn as sns
import numpy
import numpy as np
import sympy as sym
import math
import matplotlib.pyplot as plt
from matplotlib.pyplot import figure, show
from scipy.stats import shapiro, bartlett, kruskal, friedmanchisquare, mannwhitneyu,
boxcox
import statsmodels.api as sm
import statsmodels.formula.api as smf
import statsmodels.stats.api as sms
from statsmodels.formula.api import ols
from statsmodels.graphics.factorplots import interaction_plot
from statsmodels.stats.diagnostic import het_breuschpagan
from statsmodels.stats.stattools import durbin_watson
from statsmodels.sandbox.stats.runs import runtest_1samp
from statsmodels.stats.diagnostic import kstest_normal
from statsmodels.stats.outliers_influence import summary_table
from statsmodels.sandbox.regression.predstd import wls_prediction_std
import scipy.optimize as optimize
from scipy.optimize import curve_fit
%matplotlib inline
```

- 2) Create a data frame (df) that reads the .csv files containing flyaway and hardware (PME) WBS level costs

```
rate_df = pd.read_csv("Flyaway.csv")
```

- 3) Either create an iterative algorithm to automatically select subsets of the .csv file for each production program or use the code below and manually update the data frame each time with two letters. The first letter was arbitrarily chosen to designate the mission type.

```
Aa_df=rate_df.query('Program=="A-10A Aircraft Production"')
Ab_df=rate_df.query('Program=="A-4C Aircraft Production"')
Ac_df=rate_df.query('Program=="A-4E Aircraft Production"')
```

```

Ad_df=rate_df.query('Program=="A-6E Aircraft Production"')
Ae_df=rate_df.query('Program=="A-7D Aircraft Production"')
Af_df=rate_df.query('Program=="AH-64E Apache (Formerly AB3) Production"')
Ag_df=rate_df.query('Program=="AV-8B Aircraft Production"')
Ba_df=rate_df.query('Program=="B-1B Aircraft Production"')
BRa_df=rate_df.query('Program=="B-52A/B/C/D/E/F Aircraft Production"')
Bb_df=rate_df.query('Program=="B-52D/E/G/H Aircraft Production (Wichita)"')
Ca_df=rate_df.query('Program=="C-130J Aircraft System w/Engine Production"')
Cb_df=rate_df.query('Program=="C-17A Aircraft Production"')
Cc_df=rate_df.query('Program=="C-5 Reliability Enhancement & Reengineering (RERP) Production"')
Cd_df=rate_df.query('Program=="C-5 Wing Modification Production"')
Ce_df=rate_df.query('Program=="C-5B Aircraft Production"')
EWa_df=rate_df.query('Program=="AEA - E/A-18G - Electronic Variant of the F/A-18 Aircraft Production"')
EWb_df=rate_df.query('Program=="AEA - Prime - E/A-18G - Electronic Variant of the F/A-18 Aircraft Production"')
EWc_df=rate_df.query('Program=="AWACS Blk40/45 Upgrade Production"')
EWd_df=rate_df.query('Program=="E-3A Aircraft Production"')
EWe_df=rate_df.query('Program=="EA-18G Aircraft Production"')
EWf_df=rate_df.query('Program=="EA-6B Aircraft Production"')
EWg_df=rate_df.query('Program=="JSTARS Radar Subsystem (E-8C) Production"')
Fa_df=rate_df.query('Program=="AN/APG-79 Production"')
Fb_df=rate_df.query('Program=="F/A-18A/B Aircraft Production"')
Fc_df=rate_df.query('Program=="F/A-18C/D Aircraft Production"')
Fd_df=rate_df.query('Program=="F/A-18E/F Aircraft Production"')
Fe_df=rate_df.query('Program=="F-117A Aircraft Production"')
Ff_df=rate_df.query('Program=="F-14A Aircraft Production"')
Fg_df=rate_df.query('Program=="F-15A/B Aircraft Production"')
Fh_df=rate_df.query('Program=="F-15C/D Aircraft Production"')
Fi_df=rate_df.query('Program=="F-16A/B Aircraft Production"')
Fj_df=rate_df.query('Program=="F-22A Aircraft Production"')
Fk_df=rate_df.query('Program=="F-35A - Joint Strike Fighter (JSF) Aircraft Production Program"')
Fl_df=rate_df.query('Program=="F-35B - Joint Strike Fighter (JSF) Aircraft Production Program"')
Fm_df=rate_df.query('Program=="F-4B Aircraft Production"')
Fn_df=rate_df.query('Program=="F4D-1 Aircraft Production"')
Fo_df=rate_df.query('Program=="F-4E Aircraft Production"')
MPa_df=rate_df.query('Program=="P-3C Aircraft Production"')
MPb_df=rate_df.query('Program=="P-8A Multi-Mission Maritime Aircraft (MMA) Aircraft Production"')
Ra_df=rate_df.query('Program=="RF-4C Aircraft Production"')
Rb_df=rate_df.query('Program=="RQ-4A/B Aircraft Production"')
SPa_df=rate_df.query('Program=="HD/TLA MTS-B (AN/DAS-1) Production"')
SPb_df=rate_df.query('Program=="MH-60R Avionics Production"')
SPc_df=rate_df.query('Program=="MH-60R Naval Hawk Helicopter Production"')
Ka_df=rate_df.query('Program=="KC-135A Aircraft Production"')
Ta_df=rate_df.query('Program=="T-1A Aircraft Production"')
Tb_df=rate_df.query('Program=="T-38A Aircraft Production"')
Tc_df=rate_df.query('Program=="TA-4J Aircraft Production"')
Ua_df=rate_df.query('Program=="H1 UPGRADE Production Program - UH-1Y"')

```

4) Manually update the program by changing ?? to Aa, Ab, etc.

mastertest = ??\_df

## Wright's CUMAV (Nonlinear Solver)

```
test = mastertest

#references the named columns on the .csv
x = test['CUP']
y = test['CAUC']

#changes the data in the referenced columns to arrays
x = np.array(x, dtype=float)
y = np.array(y, dtype=float)

#plots the actual cost data points
plt.plot(x, y, 'ko', label="Actual Data")
plt.xlabel('Cumulative Units Produced')
plt.ylabel('Cumulative Average Unit Cost')
plt.legend(loc='upper right')

#prints the title of the learning curve model
print("\nCUMAV\n")

#defines the traditional learning curve model function
def func(fx,a,b):
    return a*fx**b

#obtains the parameters
popt, pcov = curve_fit(func, x, y)
print("Parameters a and b:")
print(*popt, sep='\n')

#plots a graph of the function
plt.plot(x, func(x, *popt), 'r--')

#calculates the relative errors and mean relative error
relerr = abs(1 - y / func(x, *popt))
np.set_printoptions(suppress=True)
print("Relative Errors:", relerr * 100)
avgrel = sum(relerr) / len(y) * 100
print("Mean Relative Error:", avgrel)
```

## Crawford's ULC (Nonlinear Solver)

```
test = mastertest

#references the named columns on the .csv
x2 = test['LMP']
y2 = test['AUC']

#changes the data in the referenced columns to arrays
x2 = np.array(x2, dtype=float)
y2 = np.array(y2, dtype=float)

#plots the actual cost data points
plt.plot(x2, y2, 'ko', label="Actual Data")
plt.xlabel('Lot Midpoint')
plt.ylabel('Average Unit Cost')
plt.legend(loc='upper right')

#prints the title of the learning curve model
print("\nULC\n")

#defines the traditional learning curve model function
def func(fx,a,b):
    return a*fx**b

#obtains the parameters
popt, pcov = curve_fit(func, x2, y2)
print("Parameters a and b:")
print(*popt, sep='\n')

#plots a graph of the function
plt.plot(x2, func(x2, *popt), 'b--')

#calculates the relative errors and mean relative error
relerr = abs(1 - y2 / func(x2, *popt))
np.set_printoptions(suppress=True)
print("Relative Errors:", relerr * 100)
avgrel = sum(relerr) / len(y2) * 100
print("Mean Relative Error:", avgrel)
```

## Boone's CUMAV (Nonlinear Solver)

```
test = mastertest

#references the named columns on the .csv
x3 = test['CUP']
y3 = test['CAUC']

#changes the data in the referenced columns to arrays
x3 = np.array(x3, dtype=float)
y3 = np.array(y3, dtype=float)

#plots the actual cost data points
plt.plot(x3, y3, 'ko', label="Actual Data")
plt.xlabel('Cumulative Units Produced')
plt.ylabel('Cumulative Average Unit Cost')
plt.legend(loc='upper right')

#prints the title of the learning curve model
print("\nBOONE CUMAV\n")

#defines Boone's learning curve model function
def func(fx,a,b,c):
    return a*fx**(b/(1+(fx/c)))

#obtains the parameters
#the maxfev allows the nonlinear solver to try 50000 times to obtain the optimal
parameters while the bounds were necessary to guide the nonlinear solver
#often b will vary from -6.644 to -0.0144, corresponding to a 1% to 99% decrease in cost
or labor hours every time X doubles, respectively
#Boone's decay variable ranges from 0 to 5000
popt, pcov = curve_fit(func, x3, y3, maxfev=50000, bounds=([0,-6.644,0],[2000000,-
0.0144,5000]))
print("Parameters a, b, and c:")
print(*popt, sep='\n')

#plots a graph of the function
plt.plot(x3, func(x3, *popt), 'm--')

#calculates the relative errors and mean relative error
relerr = abs(1 - y3 / func(x3, *popt))
np.set_printoptions(suppress=True)
print("Relative Errors:", relerr * 100)
avgrel = sum(relerr) / len(y3) * 100
print("Mean Relative Error:", avgrel)
```

## Boone's ULC (Nonlinear Solver)

```
test = mastertest

#references the named columns on the .csv
x4 = test['LMP']
y4 = test['AUC']

#changes the data in the referenced columns to arrays
x4 = np.array(x4, dtype=float)
y4 = np.array(y4, dtype=float)

#plots the actual cost data points
plt.plot(x4, y4, 'ko', label="Actual Data")
plt.xlabel('Lot Midpoint')
plt.ylabel('Average Unit Cost')
plt.legend(loc='upper right')

#prints the title of the learning curve model
print("\nBOONE ULC\n")

#defines Boone's learning curve model function
def func(fx,a,b,c):
    return a*fx**(b/(1+(fx/c)))

#obtains the parameters
#the maxfev allows the nonlinear solver to try 50000 times to obtain the optimal
parameters while the bounds were necessary to guide the nonlinear solver
#often b will vary from -6.644 to -0.0144, corresponding to a 1% to 99% decrease in cost
or labor hours every time X doubles, respectively
#Boone's decay variable ranges from 0 to 5000
popt, pcov = curve_fit(func, x4, y4, maxfev=50000, bounds=([0,-6.644,0],[2000000,-
0.0144,5000]))
print("Parameters a, b, and c:")
print(*popt, sep='\n')

#plots a graph of the function
plt.plot(x4, func(x4, *popt), 'c--')

#calculates the relative errors and mean relative error
relerr = abs(1 - y4 / func(x4, *popt))
np.set_printoptions(suppress=True)
print("Relative Errors:", relerr * 100)
avgrel = sum(relerr) / len(y4) * 100
print("Mean Relative Error:", avgrel)
```

## Wright's CUMAV (Ordinary Least Squares)

```
test = mastertest

#references the named columns on the .csv
x = test['CUP']
y = test['CAUC']

#changes the data in the referenced columns to arrays
x = np.array(x, dtype=float)
y = np.array(y, dtype=float)

#plots the actual cost data points
plt.plot(x, y, 'ko', label="Actual Data")
plt.xlabel('Cumulative Units Produced')
plt.ylabel('Cumulative Average Unit Cost')
plt.legend(loc='upper right')

#prints the title of the learning curve model
print("\nCUMAV\n")

#defines the traditional learning curve model function
def func(fx,a,b):
    return a*fx**b
#defines the log of the traditional learning curve model function
def logfunc(fx,a,b):
    return np.log(func(fx,a,b))

#obtains the parameters
popt, pcov = curve_fit(logfunc, x, np.log(y))
print("Parameters a and b:")
print(*popt, sep='\n')

#plots a graph of the function
plt.plot(x, func(x, *popt), 'r--')

#calculates the relative errors and mean relative error
relerr = abs(1 - y / func(x, *popt))
np.set_printoptions(suppress=True)
print("Relative Errors:", relerr * 100)
avgrel = sum(relerr) / len(y) * 100
print("Mean Relative Error:", avgrel)

#defines a data frame to use OLS
df = pd.DataFrame({'x':func(x, *popt), 'y':y})

#performs OLS
wrightfit_lm = smf.ols('y~x', data=df).fit()
```

## Crawford's ULC (Ordinary Least Squares)

```
test = mastertest

#references the named columns on the .csv
x2 = test['LMP']
y2 = test['AUC']

#changes the data in the referenced columns to arrays
x2 = np.array(x2, dtype=float)
y2 = np.array(y2, dtype=float)

#plots the actual cost data points
plt.plot(x2, y2, 'ko', label="Actual Data")
plt.xlabel('Lot Midpoint')
plt.ylabel('Average Unit Cost')
plt.legend(loc='upper right')

#prints the title of the learning curve model
print("\nULC\n")

#defines the traditional learning curve model function
def func(fx,a,b):
    return a*fx**b
#defines the log of the traditional learning curve model function
def logfunc(fx,a,b):
    return np.log(func(fx,a,b))

#obtains the parameters
popt, pcov = curve_fit(func, x2, y2)
print("Parameters a and b:")
print(*popt, sep='\n')

#plots a graph of the function
plt.plot(x2, func(x2, *popt), 'b--')

#calculates the relative errors and mean relative error
relerr = abs(1 - y2 / func(x2, *popt))
np.set_printoptions(suppress=True)
print("Relative Errors:", relerr * 100)
avgrel = sum(relerr) / len(y2) * 100
print("Mean Relative Error:", avgrel)

#defines a data frame to use OLS
df = pd.DataFrame({'x':func(x2, *popt), 'y':y2})

#performs OLS
crawfordfit_lm = smf.ols('y~x', data=df).fit()
```



## Bibliography

- Alchian, A. (1963). Reliability of Progress Curves in Airframe Production. *Econometrica*, 31(4), 679-693. <https://doi.org/10.2307/1909166>
- Amir, A. & Weiss, S. (2020, November 5). *Boeing Company*. *Encyclopedia Britannica*. <https://www.britannica.com/topic/Boeing-Company>
- Asher, H. (1956). Cost-Quantity Relationships in the Airframe Industry. *RAND Corporation*, R-291. <https://www.rand.org/pubs/reports/R291.html>
- Boeing (2020). *Boeing International*. <https://www.boeing.com/company/key-orgs/boeing-international/>
- Boone, E. (2018). An Analysis of Learning Curve Theory and the Flattening Effect at the End of the Production Cycle. Air Force Institute of Technology, AFIT-ENV-MS-18-M-181. <https://scholar.afit.edu/etd/1877/>
- Britannica, T. Editors of Encyclopaedia (2020, June 12). *McDonnell Douglas Corporation*. *Encyclopedia Britannica*. <https://www.britannica.com/topic/McDonnell-Douglas-Corporation>
- Crawford, J. (1944). Learning Curve, Ship Curve, Ratios, Related Data. *Lockheed Aircraft Corporation*.
- DAU, T. Editors of Glossary (2021). *Defense Acquisition University*.
- Hogan, D. (2020). An Analysis of Learning Curve Theory and Diminishing Rates of Learning. Air Force Institute of Technology, AFIT-ENV-MS-20-M-212. <https://scholar.afit.edu/etd/3607/>
- ICEAA CEBOK, T. Editors of Glossary (2013). *International Cost Estimating and Analysis Association*.
- ICEAA Modules, T. Editors of Modules (2014). *International Cost Estimating and Analysis Association*.
- Mislick, G., & Nussbaum, D. (2015). *Cost Estimation: Methods and Tools*. Wiley.
- Wright, T. (1936). Factors Affecting the Cost of Airplanes. *Journal of Aeronautical Sciences*, 3(4): 122–28. <http://arc.aiaa.org/doi/abs/10.2514/8.155>.

REPORT DOCUMENTATION PAGE					Form Approved OMB No. 0704-0188	
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p><b>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</b></p>						
1. REPORT DATE (DD-MM-YYYY) 25-03-2021		2. REPORT TYPE Graduate Research Paper		3. DATES COVERED (From - To) October 2019 - March 2021		
4. TITLE AND SUBTITLE An Analysis of Uncertainty in Learning Curve Models				5a. CONTRACT NUMBER		
				5b. GRANT NUMBER		
				5c. PROGRAM ELEMENT NUMBER		
				5d. PROJECT NUMBER		
6. AUTHOR(S) Pollack, Alexander, D., 1st Lt, USAF				5e. TASK NUMBER		
				5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way Wright-Patterson AFB OH 45433-7765				8. PERFORMING ORGANIZATION REPORT NUMBER AFIT-ENV-MS-21-M-257		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Intentionally Left Blank				10. SPONSOR/MONITOR'S ACRONYM(S)		
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION/AVAILABILITY STATEMENT Distribution Statement A. Approved for Public Release; Distribution Unlimited.						
13. SUPPLEMENTARY NOTES This work is declared a work of the U.S. Government and is not subject to copyright protection in the United States.						
14. ABSTRACT The dilemma with learning curves and actual costs is "fitting square pegs into round holes." Alternative regression techniques or further modifications to $Y=aX^b$ may address discrepancies in modeling observed data with learning curves. Adequate data permits further analysis of the constants, variables, and factors that make up a learning curve model. The analysis involves trying a nonlinear solver on the traditional learning curve models (i.e., Wright's cumulative average theory and Crawford's unit learning curve theory) and changing the theoretical first unit cost (T1) to the actual first unit cost (A1). Mergers and acquisitions in the aircraft industry may have reduced certainty in learning curve models since human resources were dispersed to manufacturing facilities worldwide. The results indicated no significant differences between using a nonlinear solver over ordinary least squares. However, the nonlinear solver propagated more uncertainty into the learning curve models. The actual first unit cost (A1) almost rids cumulative average theory models' statistical advantage over unit learning curve theory models. Conversely, mergers and acquisitions reduced uncertainty in learning curve models. Research on program attributes should continue when the appropriate data is available. Cost						
15. SUBJECT TERMS learning curve models; regression techniques; actual first unit cost; mergers and acquisitions						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON	
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. John J. Elshaw, AFIT/ENV	
U	U	U	UU	53	19b. TELEPHONE NUMBER (Include area code) (937) 255-3636, ext 3363 john.elshaw@afit.edu	