2011

Privacy Preserving Regression Residual Analysis

John Ross Wallrabenstein
Purdue University, jwallrab@purdue.edu

Chris Clifton
Purdue University, clifton@cs.purdue.edu

Report Number:
11-017

Privacy Preserving Regression Residual Analysis

John Ross Wallrabenstein* Chris Clifton†

Abstract

Regression analysis is one of the most basic statistical tools for generating predictive models that describe the relationship between variables. Once a model has been generated, numerous goodness-of-fit measures are used to evaluate the degree to which the model characterizes the relationship between the variables under consideration. The analysis of regression residuals is one such measure, where residuals may be subjectively examined for the presence of structure. However, the residual plots reveal substantial information about each participant's private data. This issue is most pronounced in the two party case, where the violation of privacy is complete. In this work, we describe an algorithmic approach drawn from random graph theory to evaluate the degree of deviation of the regression residuals from an ideal model. We demonstrate that our approach is effective at characterizing accurate and poor models where previously proposed measures remain neutral or are not applicable. Finally, we provide an efficient privacy preserving protocol for computing our proposed goodness-of-fit measure.

1 Introduction

Regression analysis on data aggregated from multiple autonomous sources will likely result in the discovery of more general and useful relationships. Indeed, regression analysis is a widely used and powerful statistical modeling tool. Such a protocol presents a problem, as determining the goodness-of-fit for for the model may result in a violation of privacy. Specifically, the residual plots reveal substantial information about each participant's private data. This issue is most pronounced in the two party case, where the violation of privacy is complete [24]. That is, examining the residual plot for regression involving a single independent and dependent variable reveals the private input of the other party in its entirety.

As an example, suppose that Alice manufactures catalytic converters, and Bob is an auto manufacturer working on a new engine design. Alice would like to know the relationship between varying levels of platinum and emissions, to optimize the cost of materials vs. size of the converter. However, revealing this to Bob would give Bob an advantage in price negotiations. Bob does not want to reveal the total emissions, as difficulties with engine management software are leading to high variance in these amounts. Disclosing this would instill fear that the new engine will be delayed, and lead consumers towards competing products. To solve this dilemma, they agree to use privacy preserving regression approaches [1, 7, 21, 9] so Alice can learn the parameters for potential models relating platinum content to emissions. But which model is best? Analysis of the regression residuals is a common method for determining a model's goodness-of-fit. Numerous statistics exist to evaluate the residuals for desirable properties under specific assumptions about the type of regression model or the distribution of the residuals. However, these measures require that the residuals be known to both parties. This is undesirable, as in the two-party case, the input of the other player is disclosed given the regression equation and the residuals.

We seek a privacy preserving solution to the problem of determining the residual's distance from an ideal residual model. To accomplish this, we build on previous results from the study of entropy in arbitrary graphs [8, 11, 22]. We derive and evaluate an area of influence measure for residuals, and build an undirected graph from the residual points based on the induced adjacency matrix. With a graph constructed from the residual plot, we evaluate the entropy content of the graph against a modified version of the Erdős-Rényi random graph model to decide whether or not the residuals imply a proper regression model.

We present an algorithm that, for sufficiently large sets of points, distinguishes accurately between proper and poor residual sets. The accuracy of our algorithm quickly approaches 1 for graphs with more than 2^6 vertices. To the best of the authors' knowledge, this work is a novel first step towards applying random graph theory to the problem of analyzing regression residual plots as a measure of goodness-of-fit.

We develop a privacy preserving protocol that evaluates the distance of arbitrary residual plots from an

*Computer Science Department, Purdue University, USA. jwallrab@cs.purdue.edu

†Computer Science Department, Purdue University, USA. clifton@cs.purdue.edu

ideal model. We consider the general two party scenario where Alice has private input $X = \{x_1, \cdots, x_m\}$ and Bob has private input $Y = \{y_1, \cdots, y_m\}$. We provide the result of the function to both parties, however it may be distributed arbitrarily. We show that our model correctly evaluates our algorithm for determining a residual plot's deviation from the ideal case, and that it is secure in the semi-honest model. While privacy preserving protocols for regression have been proposed in the literature [1, 7, 21, 9], even the most promising approach has only provided a solution for privately computing the correlation coefficient of the regression model. We demonstrate that this measure alone is insufficient for proper model selection, and compare our proposed measure to currently available statistics.

2 Problem Definition

We consider the case of *classical regression*:

$$y = \beta x + \epsilon \tag{2.1}$$

The dependent variable y is a column vector of T elements, x is the $T \times \Lambda$ matrix of values taken by the Λ independent variables, β is the column vector of Λ parameters to be estimated by the model, and ϵ a column vector of T model *disturbances*. Classical regression is concerned with finding the best parameter estimates for β that minimize the model disturbances ϵ. Although we review only classical regression, our algorithms apply to more advanced models as well.

Once such a model has been found, a common method for verifying the goodness-of-fit is to examine the residual plot visually for patterns. If any patterns or structure are present in the residual plot, the model validity is called into question. Specifically, the residual plot shows the relationship between the dependent variable y and the *residuals* $r_{i,0 \leq i \leq T} = y_i - \hat{y}_i$, the distance between the observed value of the dependent variable, y_i, and the value predicted by the regression model, $\hat{y}_i$. If the model is a good fit, the residual plot should appear random. However, when simple linear regression is applied to variables exhibiting heteroscedasticity or a non-linear association, the residual plot will exhibit structure. Unless otherwise specified, we deal exclusively with the set of *standardized* residuals.

More formally, it is desirable that the residuals exhibit the following three properties:

1. Normality: The magnitude of the residuals essentially follows a Gaussian distribution.

2. Homoskedasticity: The residuals have similar variance.

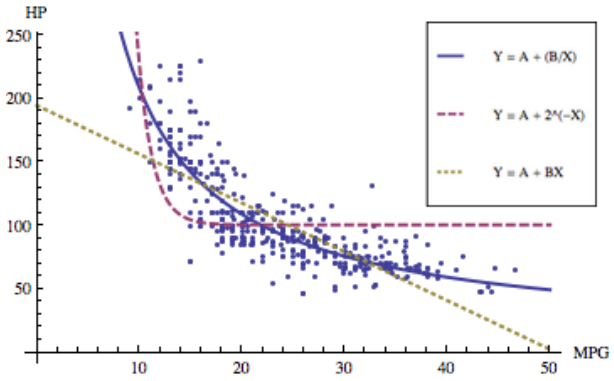
3. Serial Independence: The residuals are not cross-correlated with themselves.

Numerous test statistics have been proposed to validate whether or not regression residuals satisfy one or more of these properties [2, 3, 4, 5, 10, 13, 15, 16, 20, 23]. We propose a novel method based on random graph theory to evaluate the goodness-of-fit of regression models based on the residuals. We show that our measure is capable of proper model selection where other measures remain neutral or are not applicable.

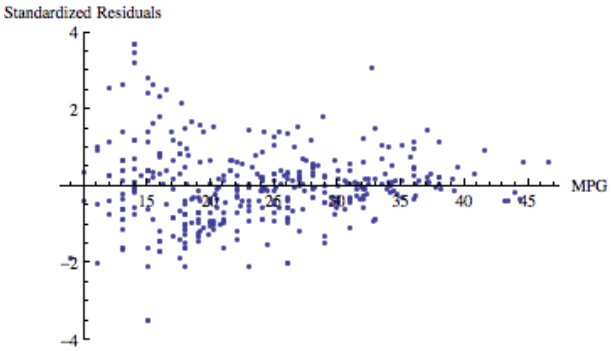
Figure 1 illustrates the 1993 Auto MPG data set from the UCI Machine Learning Repository [12] as described by three candidate regression models. The model $a + \frac{b}{x}$ fits the data well, and the corresponding standardized residual plot is depicted in Figure 1b. Although the residual plot is not truly uniform, the residuals are roughly partitioned by the x-axis and display minimal structure. Note that this model is the best fit to the data, yet the residuals exhibit heteroscedasticity. The model $a + \frac{b}{2^x}$ captures the general trend, but is a poor fit to the data. Note that the residual plot, illustrated in Figure 1d, exhibits a high degree of clustering and a clear downward trend. The model $a + bx$ does not capture the fact that at high horsepower, increased horsepower does not have nearly the impact on mileage that it does at low horsepower. The model's residuals in Figure 1f show signs of underlying structure, although to a lesser extent than Figure 1d.

The goal of our algorithm is to distinguish these residual plots algorithmically, rather than visually, to characterize the regression model's goodness-of-fit. Our algorithm operates on the residual points of the form $\{i, r_i\}_{0 \leq i \leq n}$, where n is the number of data points. That is, we consider the residuals plotted in sequence against the x-axis. These plots are illustrated in Figures 1c, 1e and 1g. Note that the structure that was clearly apparent when the residuals were plotted against the independent variable (x_i = MPG) is obfuscated in these plots. However, our measure is able to correctly differentiate between the regression models using plots of the form $\{i, r_i\}_{0 \leq i \leq n}$, in addition to plots of the residuals versus the independent variable. All scores for our measure are computed over plots of the form $\{i, r_i\}_{0 \leq i \leq n}$, rather than plots of residuals against the independent variable.

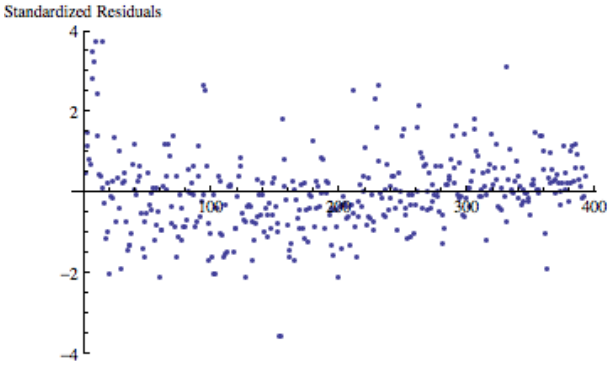
We begin by reviewing previously proposed measures for the analysis of regression residuals in Section 3. We describe our measure in Section 4, and provide an empirical evaluation in Section 5. In Section 6, we discuss its relation to existing residual measures. We describe the privacy preserving protocol for computing our measure in Section 7, and conclude with Section 8.



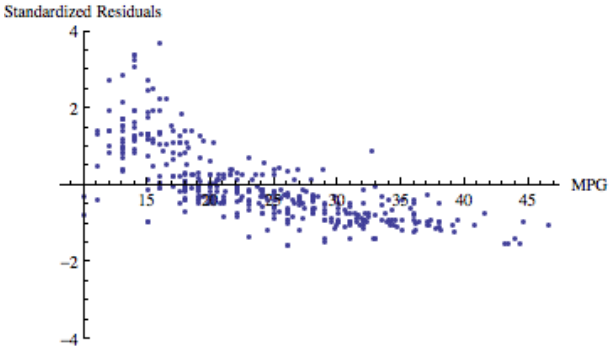
(a) UCI Automotive Dataset



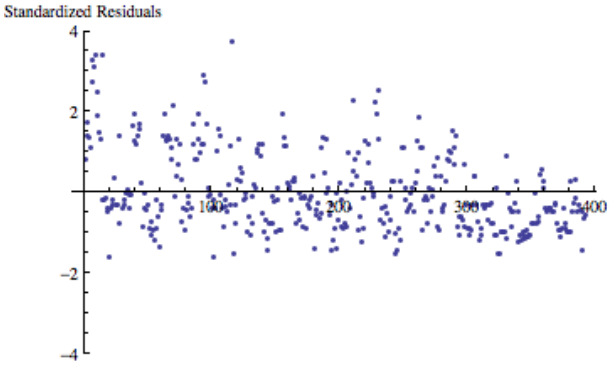
(b) Residuals for $Y = A + (B/X)$ vs. MPG



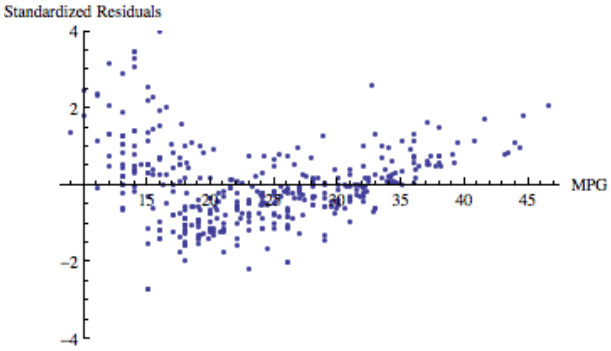
(c) Residuals for $Y = A + (B/X)$ vs. X



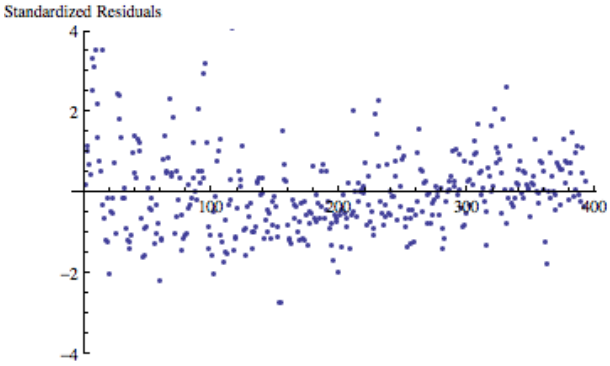
(d) Residuals for $Y = A + 2^{-X}$ vs. MPG



(e) Residuals for $Y = A + 2^{-X}$ vs. X



(f) Residuals for $Y = A + BX$ vs. MPG



(g) Residuals for $Y = A + BX$ vs. X

Figure 1: Figures 1b, 1d and 1f are of the form $\{x_i, r_i\}$, where x_i is the independent variable (x_i = MPG). Similarly, Figures 1c, 1e and 1g are of the form $\{i, r_i\}$, where r_i is the standardized residual.

3 Related Work

We review previously proposed measures for determining whether or not regression residuals satisfy *normality*, *homoskedasticity* and *serial independence*.

Of these characteristics, most attention has been focused on the assumption that the residuals can be accurately modeled under the normal distribution. The well-known Jarque-Bera (JB) Lagrange Multiplier method [16] is perhaps the most common test for residual normality. Although the JB statistic is asymptotically optimal, the true distribution of the residuals is assumed to be a member of the Pearson family. Additionally, the χ^2-approximation holds only for large sample sizes, and frequently rejects the null hypothesis of normality when it is indeed true.

Another Lagrange Multiplier test, the Breusch-Pagan (BP) statistic [5], is commonly used to test residuals for heteroscedasticity. The statistic is robust [17], however the test only addresses the issue of heteroscedasticity, and is limited in scope to linear regression.

The Breusch-Godfrey (BG) statistic [4, 13] is used to test for the presence of autocorrelation; the lack of serial independence. The statistic is more generally applicable than the similar Durbin-Watson test for serial correlation [10]. However, the test is again limited in scope to addressing autocorrelation in linear regression models.

Jarque and Bera originally proposed a more general statistic for testing normality, homoscedasticity and serial independence of regression residuals simultaneously [15]. However, as with the JB test for normality, the statistic assumes that the true distribution of the residuals are drawn from the Pearson family.

To date, little work exists in the literature with respect to privacy preserving regression in the two party case. The issue was examined briefly by Sanil et. al. [21]. However, the authors argue that the residuals should be made publicly available to perform standard goodness-of-fit tests. While some notion of privacy is preserved in this manner when more than two parties are involved, no such notion exists in the two-party case. Chaudhuri and Monteleoni approach the issue of disclosure through the residual plot from a different approach [7]. The authors introduce random noise into the private values input by the parties. However, this approach results in a suboptimal regression model. Further, an adversary learns substantial information concerning the bounds of the other party's input. Recent work by Amirbekyan and Estivill-Castro has made progress toward a true privacy preserving model diagnosis for linear regression in the two-party case [1]. The authors show how to privately compute the coefficient of determination, which is commonly used as a standalone model diagnostic. However, our algorithm goes beyond these approaches by providing an additional goodness-of-fit measure, and does not require commodity servers. Further, our algorithm is capable of distinguishing between models when the coefficient of determination is neutral.

4 Residual Analysis

We begin by reviewing Erdős and Rényi's work on random graph theory. Based on this, we give a measure of graph connectedness that measures how far a residual graph differs from an ideal residual graph. We then show how we induce a graph from the residual plot, allowing us to use the graph connectedness measure to characterize the regression model's goodness-of-fit to the data.

4.1 Erdős-Rényi Random Graphs The measure we consider to differentiate between arbitrary residual graphs and ideal residual graphs is given by Erdős and Rényi [11]. In their model, a random graph of m vertices are connected by edges with independently chosen probability $0 < p < 1$, denoted by $\mathcal{G}(m, p)$. The probability of a given graph $\mathcal{G}$ with k edges is given by:

$$P(\mathcal{G}) = p^k(1-p)^{\binom{m}{2}-k} \tag{4.2}$$

Thus, each edge is viewed as a single Bernoulli trial occurring with probability p, while the entire edge set is represented as a Binomial distribution.

Rather than evaluate the probability of a graph directly, we modify Erdős and Rényi's model to evaluate the ratio of *connected* edges to the number of points in the regression model. Our goal is to provide an equally meaningful measure while avoiding the following observation, which is inevitable when $0 < p < 1$:

$$\lim_{k\to\infty} p^k(1-p)^{\binom{m}{2}-k} = 0 \tag{4.3}$$

Clearly a straightforward application of Erdős and Rényi's $\mathcal{G}(m, p)$ measure would be ineffective for large data sets. Thus, we introduce the notion of *graph connectedness* as:

$$\mathcal{C}(\mathcal{G}) = \frac{|\mathcal{E}_{\mathcal{G}}|}{|\mathcal{V}|} \tag{4.4}$$

We have that $\mathcal{C}(\mathcal{G}) = 0$ when no structure is present in the induced graph, and $\mathcal{C}(\mathcal{G}) > 0$ indicates the degree of structure present. We discuss the derivation of this measure and its relation to ideal regression models in Section 4.2.

4.2 Inducing A Graph We aim to map a set of points $\mathcal{P} \mapsto \mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. The mapping from $\mathcal{P} \mapsto \mathcal{V}$ is trivial: the vertex set $\mathcal{V}$ is simply the set of points $\mathcal{P}$. In order to complete our induced graph $\mathcal{G}$ over the set of residual points $\mathcal{P}$, we must define the criteria that governs whether or not an edge exists between $p_i, p_j \in \mathcal{P}, i \neq j$. We first introduce the necessary building blocks before discussing the mapping from $\mathcal{P} \mapsto \mathcal{E}$ in Section 4.2.3.

4.2.1 Ideal Regression Models We consider *ideal* regression models to be those where all data points lie on the regression line. Thus, given an arbitrary point $p_i = (x_i, r_i), p_i \in \mathcal{P}$, we have that $\forall r_i, r_i = 0$ as $y_i = \hat{y}_i \implies y_i - \hat{y}_i = 0 = r_i$. We require that any regression model that is *ideal* satisfy:

$$\mathcal{C}(\mathcal{G}_{Ideal}) = 0 \tag{4.5}$$

Further, any deviation of a real regression model $\mathcal{G}_{Real}$ from the ideal model $\mathcal{G}_{Ideal}$ should be penalized in proportion to the theoretical distance between the two models. Thus, $\forall \mathcal{G} \neq \mathcal{G}_{Ideal}, \mathcal{C}(\mathcal{G}) \geq 0$. That is, $\mathcal{C}(\mathcal{G})$ represents the extent to which the residual graph $\mathcal{G}$ lacks the ideal properties of normality, homoscedasticity and serial independence.

4.2.2 Radius of Influence We define the *radius of influence* of a point $p_i \in \mathcal{P}$ as $|r_i|$, or the non-negative distance of the residual from the x-axis. Clearly, residuals for data points farther from the regression line have a greater radius of influence than those closer to the predicted value $\hat{y}_i$. Thus, the area of influence of a given residual r_i is:

$$\textit{Influence} = \pi \cdot (|r_i|)^2 \tag{4.6}$$

We now show how to map $\mathcal{P} \mapsto \mathcal{E}$ using our definition of a residual's radius of influence.

4.2.3 Edge Connections We say that an edge exists between two points $p_i, p_j \in \mathcal{P}$ iff $dist(p_i, p_j) < |r_i| + |r_j|$. That is:

$$\sqrt{(x_i - x_j)^2 + (r_i - r_j)^2} < |r_i| + |r_j| \implies e_{ij} \in \mathcal{E}$$

THEOREM 4.1. $\forall \mathcal{G}, \mathcal{G} \in \mathcal{G}_{Ideal} \implies \mathcal{C}(\mathcal{G}) = 0$

Proof. Recall that $\forall r_i, r_j \in \mathcal{G}_{Ideal}, r_i = r_j = 0$. Thus, for any two *consecutive* points p_i, p_j where $x_j = x_i + 1$, we have that:

$$\begin{aligned} dist(p_i, p_j) &= \sqrt{(x_i - (x_i + 1))^2 + (0 - 0)^2} \\ &= \sqrt{(1)^2 + 0} \\ &= 1 > |r_i| + |r_j| = 0 \\ &\implies \mathcal{C}(\mathcal{G}) = 0 \end{aligned}$$

Thus, none of the residuals in an ideal regression model are connected.

For the privacy preserving computation of our measure, we square the Euclidean distance between p_i, p_j, and the sum $|r_i| + |r_j|$. This eliminates computing the root for the Euclidean distance, which is difficult under Z_n, while preserving the relationship between the two values.

4.3 Residual Entropy Algorithm To analyze a residual plot algorithmically, we first construct a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ over the residual points, as described in Section 4.2. The calculation of $\mathcal{C}(\mathcal{G})$ is given in Algorithm 4.3.1.

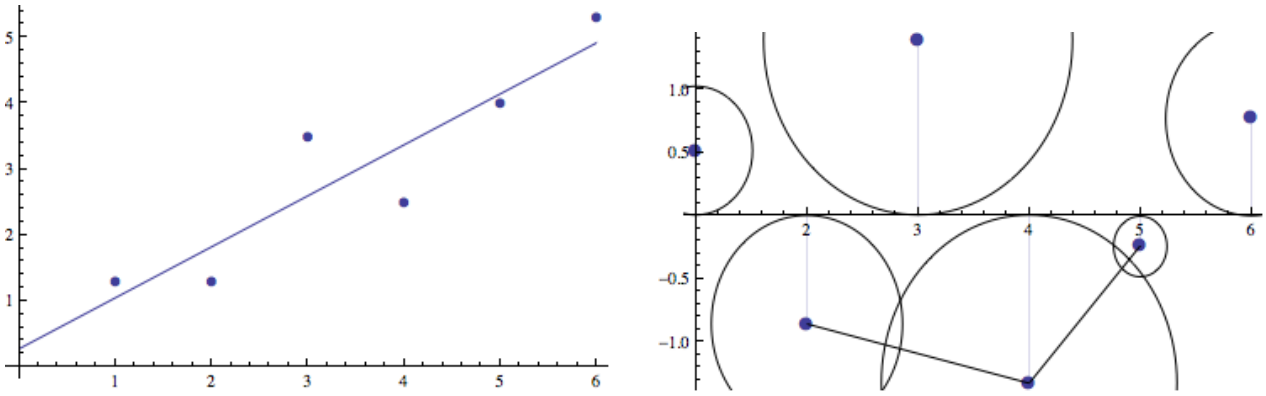
Algorithm 4.3.1 Residual Entropy Algorithm

```
Given: A set of points p_i ∈ P
Number of adjacent edges N ← 0
for all p_i ∈ P do
  for all p_j ∈ P, i ≠ j, j > i do
    if dist(p_i, p_j) < |r_i| + |r_j| then
      N ← N + 1
    end if
  end for
end for
return C(G) = N / |V|
```

4.4 Algorithm Example Consider the following regression example, where the data are related by the regression equation $y = 0.2733 + 0.7743x$:

x	1	2	3	4	5	6
y	1.3	1.3	3.5	2.5	4	5.3
$\frac{r_i - \mu_r}{\sigma_r}$	0.51	-0.86	1.39	-1.33	-0.24	0.77

The regression line accurately models the data and represents a good fit, displayed in Figure 2a. The residuals are distributed randomly, and are divided in half by the x-axis, displayed in Figure 2b. The algorithm proceeds by finding the distance between each pair of residuals by computing the standard Euclidean distance metric. In this example, there are two sets of residuals with overlapping areas of influence, so $|\mathcal{E}| = 2$.



(a) Example Regression Model

(b) Example Residuals

Figure 2: Plot 2a illustrates the example data fit to the regression equation $y = 0.6762 + 0.8629x$. Plot 2b illustrates the corresponding standardized residuals. The circles denote the area of influence for each residual, where overlapping influence areas imply an edge between the residuals.

Given that there are six data points, we compute $\mathcal{C}(\mathcal{G}_{ex})$ by Equation 4.4:

$$\begin{aligned} \mathcal{C}(\mathcal{G}_{ex}) &= \frac{|\mathcal{E}_{\mathcal{G}}|}{|\mathcal{V}|} \\ &= \frac{2}{6} = \frac{1}{3} \end{aligned}$$

Figure 2b illustrates the radius of influence for the residuals, and shows the edges induced when the influence area of residuals overlap.

5 Empirical Evaluation

We now give several examples, including a comparison with other measures on the real data example of Figure 1, and synthetic data examples that demonstrate the ability of the measure to detect failures to meet normality, homoscedasticity, and serial independence.

5.0.1 Ideal Examples We present two *ideal* regression models for data following the trend $y = x^2$. In the ideal model, all data points lie on the regression line, and have corresponding residual values $r_i = 0$. Two examples of data satisfying the ideal regression model are illustrated in Figure 3.

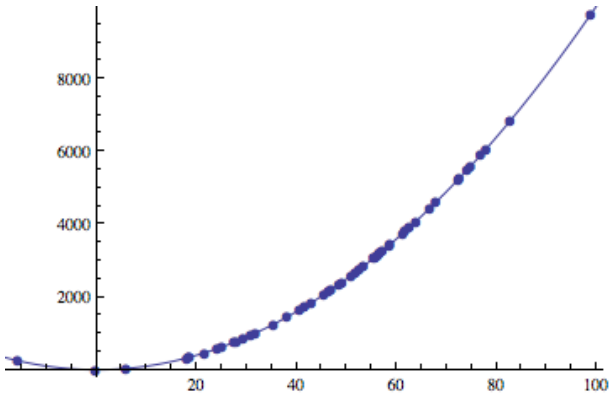
Recall that our measure is evaluated over data of the form $\{i, r_i\}_{0 \le i \le n}$, rather than the residuals versus the independent variable as $\{x_i, r_i\}_{0 \le i \le n}$. The residuals of both models score $\mathcal{C}(\mathcal{G}) = 0$, as predicted by Theorem 4.1.

5.1 UCI Automotive Data We evaluate three regression models that describe the relationship between horsepower and miles per gallon in automobiles. The first model has the general form $y = a + (b/x)$, the second follows $y = a + (b/(2^x))$, and the third follows $y = a + bx$. Our rationale for selecting these forms is to demonstrate that our measure $\mathcal{C}(\mathcal{G})$ captures a different characteristic than currently available statistics. Additionally, the goodness-of-fit for the models can be easily determined through visualization. Thus, we demonstrate how our algorithm facilitates this classification algorithmically. The three models are illustrated in Figure 1. In each case, the parameters a and b are chosen to give a best fit given the nature of the model, and the linear and geometric models would both seem plausible. Based on the residual plot, the first model visually provides an accurate description of the data, while the other two models stand out as a poor representation of the data.

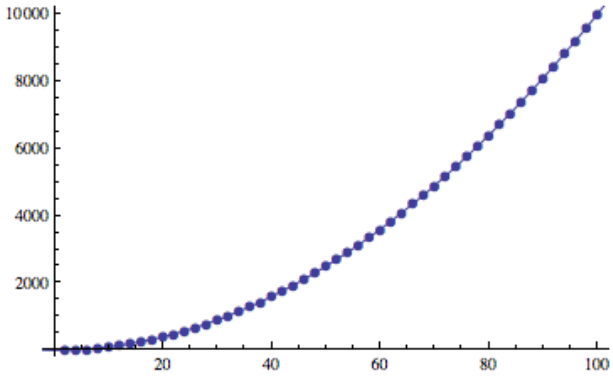
Several analytic measures are given in the following table:

	$y = a + (b/x)$	$y = a + b/(2^x)$	$y = a + bx$
RMS	1.00	1.03	1.00
R^2	0.97	0.90	0.61
JB	180, $p = 0$	331,$p = 0$	74,p=0
BP	n/a	n/a	24.5, p=0
BG	n/a	n/a	85.8, p=0
$\mathcal{C}(\mathcal{G})$	0.518	0.753	0.640

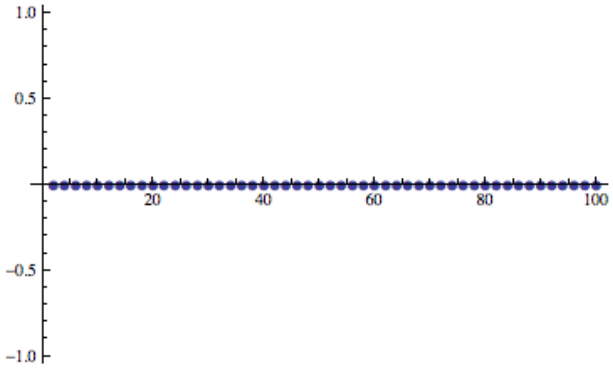
We see that our algorithm correctly picks the best model, as does R^2 analysis of the data itself. However, the R^2 analysis suggests that the exponential model is better than the linear model, giving the implausible implication that at high horsepowers, additional horsepower can be gained at almost no cost in mileage. Note that our algorithm ranks models based on the distance of the residuals from an ideal residual set, which differs



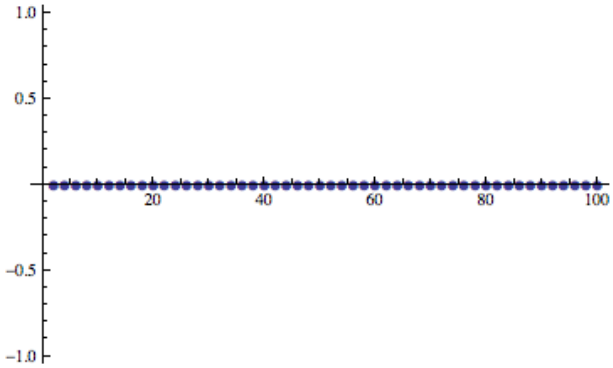
(a) Ideal Model with Data $\in N(50, 30)$



(b) Ideal Model with Uniform Data



(c) Residuals for Figure 3a



(d) Residuals for Figure 3b

Figure 3: Two example *ideal* regression models following $y = x^2$.

from the characteristics addressed by existing statistics.

We now give a somewhat deeper analysis of how the three goals of the analysis are met, and how they compare with measures specific to those goals.

5.1.1 Normality We use the Jarque-Bera (JB) statistic [16] to test the assumption that the residuals are normally distributed. For all models, the JB statistic indicates that the residuals are *not* normally distributed. That is, each model returned a JB statistic with a p-value of 0, which rejects the null hypothesis that the data are normally distributed. The JB statistic often rejects the null hypothesis for small samples. The distribution of residuals for each model are given in Figure 4.

5.1.2 Homoscedasticity We use the Breusch-Pagan (BP) statistic [5] to test the assumption that the residuals of the *linear regression* model are homoscedastic. Recall that the BP statistic is only defined for linear regression models. The BP statistic correctly indicates that the residuals for the linear regression model in Figure 1f are heteroscedastic. That is, the BP statistic gives a p-value of 0, which rejects the null hypothesis of homoscedasticity.

5.1.3 Serial Independence We use the Breusch-Godfrey (BG) statistic [4, 13] to test the assumption that the residuals of the *linear regression* model are serially independent. Recall that the BG statistic is only defined for linear regression models. The BG statistic incorrectly indicates that the residuals for the linear regression model in Figure 1f are autocorrelated. This is clearly no the case for the relationship between automobile horsepower and MPG. That is, the BG statistic gives a p-value of 0, which rejects the null hypothesis of serial independence.

5.1.4 Coefficient of Determination A primary benefit of our measure is that it can distinguish between models with the same R^2 value, favoring the residual plots of models with lower structural content. Our measure is capable of determining the best fit model even when the coefficient of determination values provide no guidance. We explore this aspect in greater detail in Section 5.2.

5.2 Synthetic Data We evaluate the accuracy of our algorithm on the residuals from synthetic data generated using Mathematica. The distributions were chosen to illustrate the ability of our measure to calculate the residuals' distance from an ideal model, despite the R^2 values for both distributions being identical.

Figure 5 illustrates a linear regression model applied to datasets that roughly follow $y = x$ and $y = x+sin(x)$, respectively. The errors added to the data are normally distributed, as $N(\mu = 0, \sigma = 1)$. Clearly the linear model represents a good fit to the data in Figure 5a, while the model poorly characterizes the non-linear association exhibited by the data in Figure 5b.

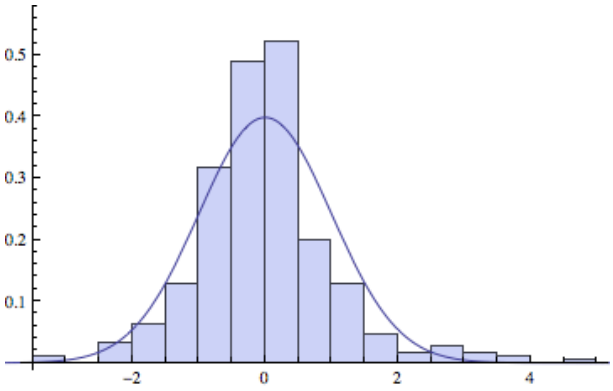
	Figure 5a	Figure 5b
RMS	1.00	1.00
R^2	0.8005	0.8007
JB	4.52, $p = 0.1$	10.4,$p = 0.01$
BP	0.0016, $p = 0.97$	0.0799, p=0.78
BG	0.5867, $p = 0.44$	289, p=0
$\mathcal{C}(\mathcal{G})$	0.432	0.848

5.2.1 Normality We use the Jarque-Bera (JB) statistic [16] to test the assumption that the residuals are normally distributed. Assuming a significance level of $\alpha = 0.01$, the JB statistic indicates that the residuals for both models are normally distributed, although the result for the data of Figure 5b is a borderline case. The distribution of residuals for Figure 5a and Figure 5b are given in Figure 5e and Figure 5f, respectively.

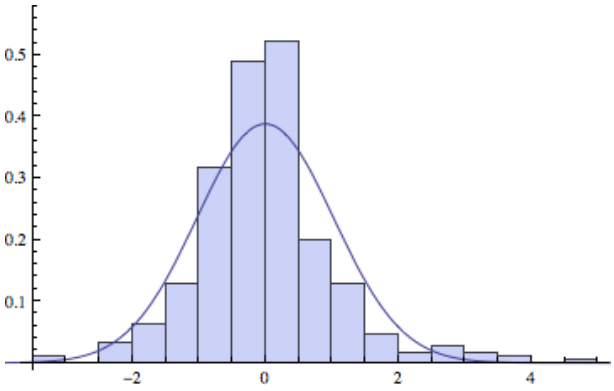
5.2.2 Homoscedasticity We use the Breusch-Pagan (BP) statistic [5] to test the assumption that the residuals of the two synthetic models are homoscedastic. The BP statistic correctly indicates that the residuals for both models are homoscedastic. That is, the BP statistic gives p-values of 0.97 and 0.78 for Figure 5c and Figure 5d, respectively, which are both greater than $\alpha = 0.01$.

5.2.3 Serial Independence We use the Breusch-Godfrey (BG) statistic [4, 13] to test the assumption that the residuals of the two synthetic models are serially independent. The BG statistic correctly indicates that the residuals in Figure 5c are serially independent, while the residuals of Figure 5d are *not* serially independent. That is, the BG statistic gives a p-value of $0.44 > \alpha = 0.01$ for the residuals in Figure 5c, so we accept the null hypothesis of serial independence. Similarly, the BG statistic gives a p-value of $0 < \alpha = 0.01$ for the residuals in Figure 5d, so we reject the null hypothesis of serial independence and conclude that the values are autocorrelated.

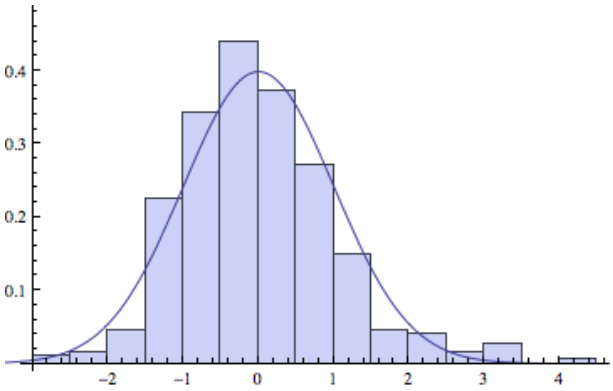
5.2.4 Coefficient of Determination Examining the residual plots, we see that both Figure 5c and Figure 5d yield identical R^2 values of approximately $\frac{8}{10} \pm 0.0002$. However, it is visually apparent that Figure 5d exhibits structure, while Figure 5c is essentially random. Given that the R^2 measure cannot differentiate



(a) Residual Distribution for $y = 9.9 + (1977.4/x)$

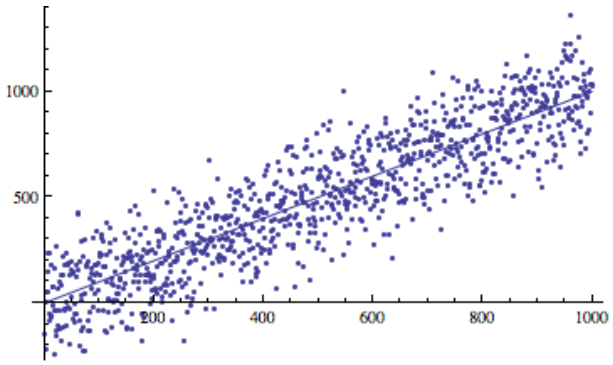


(b) Residual Distribution for $y = 100.6 + (126833 * 2^{-x})$

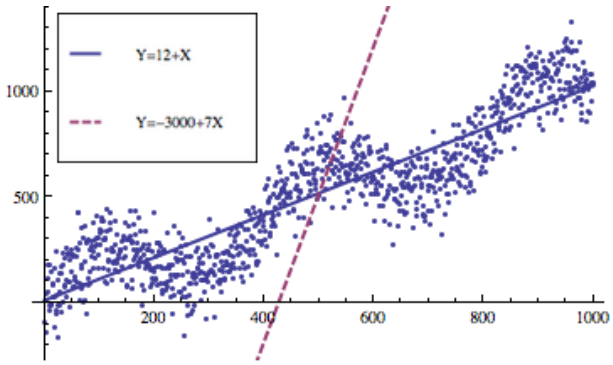


(c) Residual Distribution for $y = 194.5 - 3.8x$

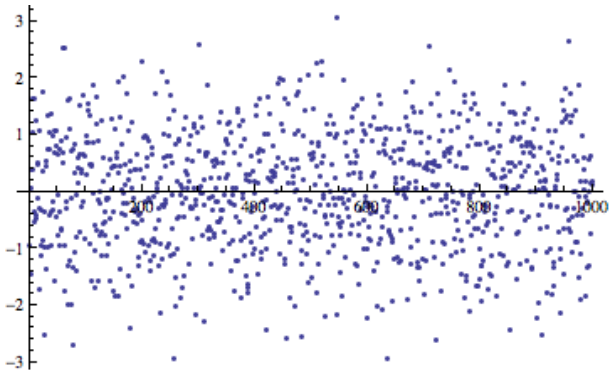
Figure 4: The residual distribution plots for the three candidate regression models of Figure 1.



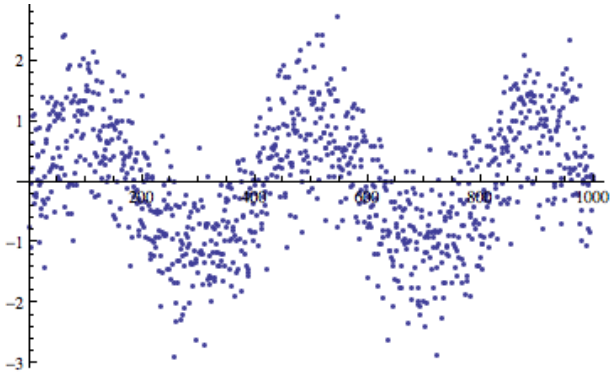
(a) $Y = X + N(\mu = 0, \sigma = 1)$



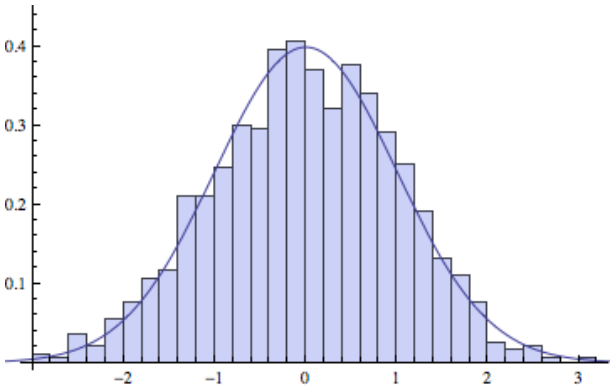
(b) $Y = X + N(\mu = 0, \sigma = 1) + 200 \cdot sin(\frac{x}{2^6})$



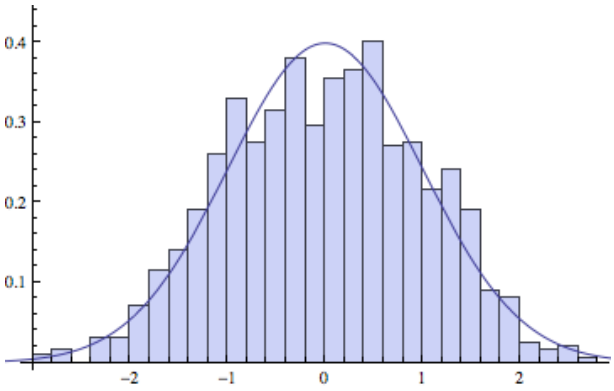
(c) Standardized Residuals for Plot 5a



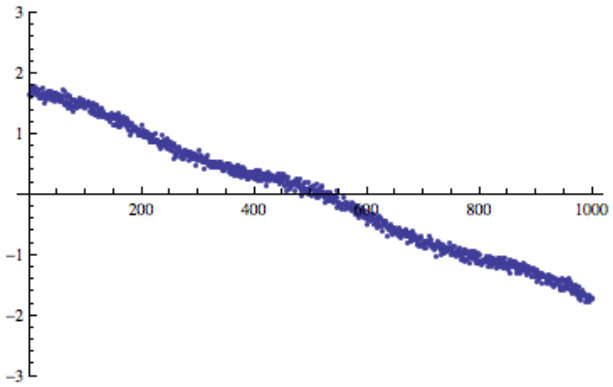
(d) Standardized Residuals for Plot 5b, $Y = 12 + X$



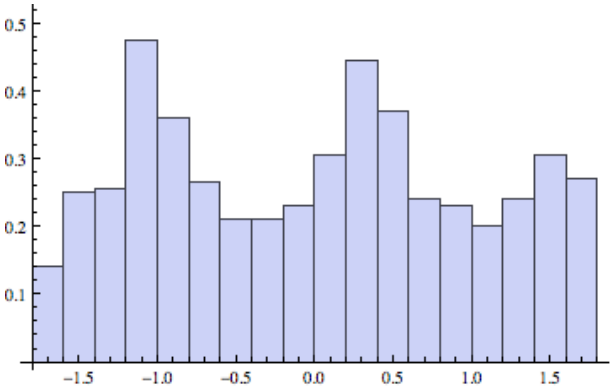
(e) Standardized Residual Distribution for Plot 5c



(f) Standardized Residual Distribution for Plot 5d



(g) Standardized Residuals for for Plot 5b, $Y = -3000 + 7X$



(h) Residual Distribution for Plot 5g

Figure 5: Synthetic Regression Data

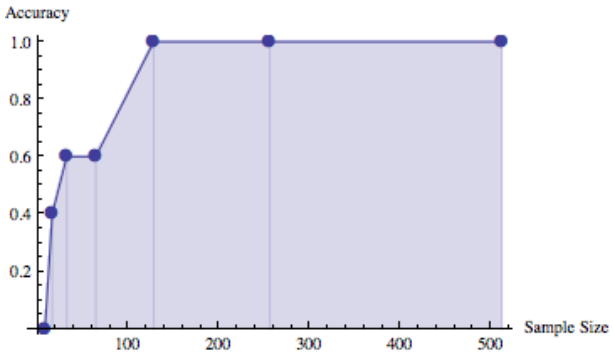


Figure 6: Algorithm Accuracy

between these two distributions, we observe that $\mathcal{C}(\mathcal{G})$ accurately characterizes Figure 5d as more structured than Figure 5c.

5.2.5 Worse Case Scores As a counterpart to the demonstration on an ideal model in Section 3, we now give an example of an extremely poor model. We again use the data of Figure 5b, but with the poorly fit linear model illustrated by the dashed line. For residual plots of the form $\{i, r_i\}_{0 \leq i \leq n}$, where n is the number of data points, $\mathcal{C}(\mathcal{G}) > 1$ for very poor models. Symmetrically, assuming the same form, $\mathcal{C}(\mathcal{G}) \approx 0$ for accurate models. As an example of the worst case scenario, consider the second model of Figure 5b. This model clearly fails to properly characterize the trend in the data, and this is easily observed by examining the corresponding residual plot in Figure 5g. The second model of Figure 5b evaluates to $\mathcal{C}(\mathcal{G}) = 1.244$, which is expected given the model's poor fit to the original data.

5.2.6 Sampling Accuracy We generate five independent samples from each distribution of size $2^i, 3 \leq i \leq 9$ and evaluate the ability of our algorithm to distinguish between the residuals of Figure 5c and Figure 5d. That is, when $\mathcal{C}(5c) < \mathcal{C}(5d)$, we claim that our algorithm accurately distinguished between the data from Figure 5c and Figure 5d. Our accuracy results are given in Figure 6. Even with small sample sizes, our algorithm scores samples from Figure 5d higher than Figure 5c, implying that the model is a better fit to Figure 5a than to Figure 5b. Thus, for large graphs it is sufficient to examine a small sample chosen uniformly at random to serve as a characterization of the data.

6 Discussion

We have demonstrated that our algorithm is capable of distinguishing poor models from those that more accurately characterize the trend of the dataset. In this section, we present an analysis of the strengths and weaknesses of our approach, having illustrated these aspects for existing measures.

6.1 Normality Our measure is biased towards residuals drawn from a normal distribution. As our measure penalizes residuals proportional to their distance from $r_i = 0$, $\mathcal{C}(\mathcal{G})$ yields a worse score for uniformly distributed residuals than for normally distributed residuals, all else being equal. That is, residuals drawn from a normal distribution $N(\mu, \sigma)$ are preferred over a uniform distribution when $\mu = 0$, as they have a smaller area of influence.

6.2 Homoscedasticity Our measure is biased against heteroscedasticity, which follows again from the observation that any deviation in the residuals from $r_i = 0$ is penalized by a greater area of influence. Clearly, this results in residuals further from zero begin connected to more neighboring residuals, which in turn increases $\mathcal{C}(\mathcal{G})$. As we deal with the standardized residuals, the presence of heteroscedasticity necessarily implies that the residuals are more dispersed in certain areas, and thus further from zero, than a similar homoscedastic plot.

6.3 Serial Independence Our measure does not directly test for the presence of serial independence. However, the synthetic data of Figure 5b gives evidence that autocorrelated residuals are scored higher than those that are serially independent; for example, the synthetic data of Figure 5a.

6.4 Related Measures Although our measure does not explicitly test for the desirable properties of residuals, it is able to differentiate between models when existing measures are neutral (e.g. R^2 of Figures 5a and 5b) or are not applicable due to restrictions on the residual distribution or regression model under consideration. Further, our measure addresses an inherently different characteristic than existing measures; the distance of the regression residuals from an ideal residual model.

7 Privacy Preserving Residual Analysis

The problem becomes more interesting from a privacy perspective. The goal of Secure Multi-Party Computation is to allow n parties to privately compute the result of some public function $f(x_1, \cdots, x_n)$. At the conclusion of the protocol, the result is made available to some subset of the parties, while their inputs $x_1, \cdots, x_n$ remain private. While it has been shown that any polynomial time function can be securely computed in polyno-

mial time [25], the general scrambled circuit evaluation method is computationally expensive for complex circuits to the point of being prohibitive. Thus, domain specific protocols are developed in response.

The goal of our privacy preserving protocol is to allow Alice and Bob to determine whether or not the residual plot of a regression model exhibits structure. If the residual plot lacks entropy, the regression model is not a good fit for the data. Without loss of generality, assume that Alice and Bob possess m values for the independent (respect dependent) variable. Assume that Alice has a private input set $X = \{x_1, \cdots, x_m\}$, and Bob has a private input set $Y = \{y_1, \cdots, y_m\}$. Let $f(X, Y) \rightarrow \mathcal{N}$ denote our algorithm for evaluating the deviation of a given graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ from an ideal residual graph, where $\mathcal{N}$ denotes the size of the induced edge set $|\mathcal{E}_\mathcal{G}|$. Our goal is an algorithm that evaluates $f(X, Y)$ and is secure under the semi-honest adversary model.

7.1 Additively Homomorphic Cryptosystem Our algorithm requires an additively homomorphic cryptosystem. Let $E_P(\cdot)$ represent encryption under party P's public key, and let $D_P(\cdot)$ represent decryption under party P's private key. We require the homomorphic cryptosystem to satisfy the following properties:

$$\begin{aligned} E(x) \cdot E(y) &= E(x + y) \\ E(x)^c &= E(x \cdot c) \end{aligned}$$

That is, the product of two ciphertexts yields the ciphertext of their sum. Similarly, exponentiation by a plaintext constant c yields the product of the encrypted value and the constant. For our implementation, we use Paillier's cryptosystem [19]. Plaintext elements under this cryptosystem are represented in $\mathbb{Z}_n$, and the encryption function maps plaintext values to ciphertext values in $\mathbb{Z}^*_{n^2}$.

7.2 Product Protocol Π It is also necessary in our protocol to obtain the product of two ciphertexts without revealing the inputs to either party. We require that the result be encrypted and blinded from both parties. The product protocol is a dialogue between Alice, the sender A, and Bob, the receiver B, where A has input $E_B(x), E_B(y)$ and wishes to obtain $E_B(x \cdot y)$ under an additively homomorphic encryption system using the receiver's public key E_B. We denote the product protocol functionality as $\Pi(E_B(x), E_B(y)) \rightarrow E_B(x \cdot y)$. Alice additively blinds x, y using independently generated random numbers $R_1, R_2 \in \mathbb{Z}_n$ and sends $\langle E_B(x+R_1), E_B(y+R_2)\rangle$ to B, the receiver. Bob decrypts and computes $P = (x + R_1) \cdot (y + R_2)$, returning $E_B((x + R_1) \cdot (y + R_2)) = E_B(P)$ to Alice who unblinds the result in encrypted form. Recall the additive homomorphic properties when evaluating operations on encrypted data.

Algorithm 7.2.1 Product Protocol Π

Alice:
$R_1 \leftarrow$ random $e \in \mathbb{Z}_n$
$R_2 \leftarrow$ random $e \in \mathbb{Z}_n$
$E_B(x') \leftarrow E_B(x) \cdot E_B(R_1)$
$E_B(y') \leftarrow E_B(y) \cdot E_B(R_2)$
Bob $\leftarrow \langle E_B(x'), E_B(y')\rangle$

Bob:
$x' \leftarrow D_B(E_B(x'))$
$y' \leftarrow D_B(E_B(y'))$
Alice $\leftarrow E_B(x' \cdot y')$

Alice:
$E_B(x \cdot y) \leftarrow E_B(x' \cdot y') \cdot E_B(x')^{-R_2} \cdot E_B(y')^{-R_1} \cdot E_B(-(R_1 \cdot R_2))$

We omit the trivial proof that the final step correctly produces the desired result, $E_B(x \cdot y)$.

7.3 Comparison Protocol λ Our protocol requires that a comparison operation can be evaluated over two ciphertexts without revealing the inputs to either party. In the protocol, Alice has two ciphertexts $E_B(x), E_B(y)$ encrypted with Bob's public key for an additively homomorphic cryptosystem. The outcome of the protocol gives Alice $E_B(1)$ when $x < y$, and $E_B(0)$ otherwise. To accomplish this, we use the FairPlay [18] implementation of Yao's scrambled circuit evaluation protocol [25].

7.4 Adapter Protocol α As our protocol will be used in addition to existing algorithms for computing regression models, we provide an adapter functionality to translate the output of a regression algorithm split between two parties to the input for our residual protocol. Our algorithm is designed as the functionality $f(x, r) \rightarrow \mathcal{N}$, where Alice's private input x are the independent variable's values, and Bob's private input r is the set of residuals $r_i = y_i - \hat{y}_i$. If the residuals are not computed during the regression algorithm, our adapter protocol provides the necessary transition functionality $\alpha((x, f(x)), y) \rightarrow \{r, \perp\}$. Alice's private input $(x, f(x))$ are the independent variable's values and the regression equation, and Bob's private input y is the set of dependent variable's values. We assume that both players know $f(x)$ from the output of the regression algorithm, although it is only necessary that Alice knows

$f(x)$. The adapter protocol returns the residual set r encrypted with Bob's key to Alice, and $\perp$ to Bob, where $\perp$ denotes no output.

Algorithm 7.4.1 Adapter Protocol α

Bob:
Publicize E_B, n
for all $y_i \in Y$ **do**
  $y'_i \leftarrow E_B(y_i)$
end for
Alice $\leftarrow Y'$

Alice:
Computed residual set Z
for all $y'_i \in Y'$ **do**
  $\bar{y}_i \leftarrow E_B(-1 \cdot f(x_i))$
  $z_i \leftarrow y'_i \cdot \bar{y}_i$
end for

7.5 Privacy Preserving Residual Algorithm ρ

Our privacy preserving algorithm is a dialogue between Alice, in possession of the m independent values $x_i \in X$, and Bob, in possession of the m dependent values $y_i \in Y$. The encryption function of Paillier's cryptosystem [19] is only defined over inputs in $\mathbb{Z}_n$, so all plaintext values are mapped from $\mathbb{R} \mapsto \mathbb{Z}_n$ through the ceiling function to facilitate encryption.

Our algorithm is given as requiring the adapter protocol α. That is, we assume that neither party has the plaintext residuals, and that Alice must compute them given encryptions of Bob's input, Y. After running the adapter protocol, Alice needs to compute the distance between every point in the residual graph. This is computed in pieces; first by computing the distance between the x-components i and j, and then computing the distance between the y-components $E_B(r_i)$ and $E_B(r_j)$. Recall that we compute our measure on graphs of the form $\{i, r_i\}$, so Alice simply increments i for each residual rather than using her private value x_i. The x-component distances are stored in D^x, and the y-component distances are stored in D^r. These are combined into $D \leftarrow D^x \cdot D^r$ to finish the computation of the squared Euclidean distance. Recall that we square this computation, and the area of influence, to avoid computing roots. The influence of each residual pair is computed by summing the encrypted residuals as $I \leftarrow r_i \cdot r_j$, as the encryption scheme is additively homomorphic.

The most expensive portion of the protocol is computing $\lambda(d_k, I_k)$ for all the distances between vertices d_k. If $d_k \leq (|r_i| + |r_j|)^2, (r_i, r_j \in I_k)$, the output of $\lambda(d_k, I_k) = 1$, which is then used to increment the number of edges $\mathcal{N}$. Note that the output of λ is not know to either party; the encrypted result is multiplied with the encryption of $\mathcal{N}$, which yields an encryption of their sum. Otherwise, Alice would deduce which edges are connected. The algorithm outputs the number of connected edges in the graph $\mathcal{N}$ as the measure of deviation from an ideal residual model.

Algorithm 7.5.1 Residual Analysis ρ

Bob:
Publicize E_B, n

Alice:
Residuals $R, r_i \in R$
$R \leftarrow \alpha(x, f(x)), y)$
Residual Distances $D^r, d^r_k \in D^r$
X-axis Distances $D^x, d^x_k \in D^x$
Combined Distances $D, d_k \in D$
Influence $I, I_k \in I$
for $r_i \in R$ **do**
  for $r_j \in R, i < j$ **do**
    $d^r_k \leftarrow \pi(r_i \cdot r_j^{-1}, r_i \cdot r_j^{-1})$
    $d^x_k \leftarrow E_B(\lceil (i-j)^2 \rceil)$
    $d_k \leftarrow d^x_k \cdot d^r_k$
    $I_k \leftarrow r_i \cdot r_j$
  end for
end for
Number of edges $\mathcal{N} \leftarrow E_B(0)$
for all $d_k \in D$ **do**
  $\mathcal{N} \leftarrow \mathcal{N} \cdot \lambda(d_k, I_k)$
end for
result $\leftarrow \mathcal{N}$

7.6 Security under the Semi-Honest Model

In our analysis, we assume that the players are semi-honest. That is, they follow the protocol specification but attempt to learn additional information from the protocol transcript. We assume all players are bound to probabilistic polynomial time (PPT), and our notation for the underlying cryptosystem follows Paillier's construction [19]. Our proof follows the definition of two-party security for semi-honest PPT adversaries under the simulation paradigm described by Goldreich [14].

LEMMA 7.1. Π *is secure under the semi-honest model.*

Proof. Recall that we assume the existence of a semantically secure additively homomorphic public cryptosystem.

Alice: The view for Alice consists of a series of ciphertexts encrypted under Bob's public key E_B, which can be efficiently simulated with random elements in $\mathbb{Z}^*_{n^2}$.

Bob: Bob's view of the protocol consists of two ciphertexts c_1, c_2 that decrypt to the plaintext values $x' = x + R_1 \mod n, y' = y + R_2 \mod n$. As R_1, R_2 are chosen uniformly at random from $\mathbb{Z}_n$, both c_1 and c_2 can be simulated with random elements from $\mathbb{Z}^*_{n^2}$, which decrypt to random elements in $\mathbb{Z}_n$.

LEMMA 7.2. *λ is secure under the semi-honest model.*

For the security proof, see [25, 18].

LEMMA 7.3. *α is secure under the semi-honest model.*

Proof. **Bob**: Bob does not receive any messages during the protocol, so the simulator runs his portion of the protocol with Alice.

Alice: We assume that the modulus n is known by both parties, as is $|Y|$. As E_B is produced by Bob independent of the data, the simulator can generate E_B in the same manner. Alice also receives Y; given the semantic security of the cryptosystem, the simulator can choose values uniformly from $\mathbb{Z}^*_{n^2}$ to simulate $E_B(y_i)$. The remainder of the simulation proceeds using Alice's protocol; given the semantic security of the cryptosystem, the resulting encrypted distances will be computationally indistinguishable from a uniform distribution over the range of $\mathbb{Z}^*_{n^2}$ regardless of the input.

THEOREM 7.1. *ρ is secure under the semi-honest model.*

Proof. We first show that Alice can simulate her view. At the final step, the result is part of the view from the ideal model; $\mathcal{N}$ can be simulated to yield the proper connectedness measure of the graph. Given the security of α, Π and λ, by the composition theorem [6] we show that ρ is secure under the semi-honest model.

Alice receives no messages, except from the output of α, which has already been shown to be secure. As with protocol α, n is assumed public and E_B is easily simulated. Due to the semantic security of the cryptosystem, the encryption of the residual values $E_B(r_i)$ can be simulated with values drawn uniformly from $\mathbb{Z}^*_{n^2}$. This simulator proceeds using Alice's protocol with the simulated values for $E_B(r_i)$. The composition theorem [6] and semantic security of the output of Π and λ enable simulation of the remaining operations performed over the simulated values $E_B(r_i)$. The exception is the final encryption of $\mathcal{N}$, which must decrypt to the correct result. This is simulated by computing $E_B(result)$, as the simulator knows the final outcome.

Bob receives no messages except through the product and comparison protocols, Π and λ respectively. This simulator can thus proceed to run Bob's algorithm, with simulations of the output of Π and λ generated as above. The composition theorem completes the proof.

8 Conclusion

We have introduced a new goodness-of-fit measure for regression models, and demonstrated its effectiveness at proper model selection. We have shown that the measure is broadly applicable to any residual plot, regardless of the distribution the original data follows. Our measure is able to select the best regression model where other measures are limited by the type of regression performed, the distribution of the original data, or are neutral with respect to the models under consideration. Further, we give algorithms for computing our measure in a secure manner, which has important implications for privacy preserving data mining.

References

[1] Artak Amirbekyan and Vladimir Estivill-Castro. Privacy-preserving regression algorithms. In *Proceedings of the 7th WSEAS International Conference on Simulation, Modelling and Optimization*, pages 37–45, Stevens Point, Wisconsin, USA, 2007. World Scientific and Engineering Academy and Society (WSEAS).

[2] D. F. Andrews. Significance tests based on residuals. *Biometrika*, 58(1):pp. 139–148, 1971.

[3] Anil K. Bera and Carlos M. Jarque. Efficient tests for normality, homoscedasticity and serial independence of regression residuals: Monte carlo evidence. *Economics Letters*, 7(4):313 – 318, 1981.

[4] T S Breusch. Testing for autocorrelation in dynamic linear models. *Australian Economic Papers*, 17(31):334–55, December 1978.

[5] T. S. Breusch and A. R. Pagan. A simple test for heteroscedasticity and random coefficient variation. *Econometrica*, 47(5):pp. 1287–1294, 1979.

[6] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *FOCS*, pages 136–145, 2001.

[7] Kamalika Chaudhuri and Claire Monteleoni. Privacy-preserving logistic regression. In *NIPS*, pages 289–296, 2008.

[8] Yongwook Choi and Wojciech Szpankowski. Compression of graphical structures. In *Proceedings of the 2009 IEEE international conference on Symposium on Information Theory - Volume 1*, ISIT'09, pages 364–368, Piscataway, NJ, USA, 2009. IEEE Press.

[9] Wenliang Du, Yunghsiang S. Han, and Shigang Chen. Privacy-preserving multivariate statistical analysis: Linear regression and classification. In *2004 SIAM International Conference on Data Mining*, Lake Buena Vista, Florida, April 22-24 2004.

[10] J. Durbin and G. S. Watson. Testing for serial correlation in least squares regression. i. *Biometrika*, 37(3-4):409–428, 1950.

[11] P. Erdős and A. Rényi. Asymmetric graphs. *Acta Math. Acad. Sci. Hungary*, 14:295–315, 1963.

[12] A. Frank and A. Asuncion. UCI machine learning repository, 2010.

[13] L. G. Godfrey. Testing against general autoregressive and moving average error models when the regressors include lagged dependent variables. *Econometrica*, 46(6):pp. 1293–1301, 1978.

[14] O. Goldreich. *Foundations of Cryptography*, volume 2. Cambridge University Press, 2004.

[15] Carlos M. Jarque and Anil K. Bera. Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economics Letters*, 6(3):255 – 259, 1980.

[16] M. Jarque, Anil K. Bera, Carlos M. Jarque, and Anil K. Bera. A test for normality of observations and regression residuals. *Internat. Statist. Rev*, pages 163–172, 1987.

[17] John D. Lyon and Chih-Ling Tsai. A comparison of tests for heteroscedasticity. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 45(3):pp. 337–349, 1996.

[18] Dahlia Malkhi, Noam Nisan, Benny Pinkas, and Yaron Sella. Fairplay; a secure two-party computation system. In *Proceedings of the 13th conference on USENIX Security Symposium - Volume 13*, SSYM'04, pages 20–20, Berkeley, CA, USA, 2004. USENIX Association.

[19] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *EUROCRYPT'99: Proceedings of the 17th international conference on Theory and application of cryptographic techniques*, pages 223–238, Berlin, Heidelberg, 1999. Springer-Verlag.

[20] Donald A. Pierce and Kenneth J. Kopecky. Testing goodness of fit for the distribution of errors in regression models. *Biometrika*, 66(1):pp. 1–5, 1979.

[21] Ashish P. Sanil, Alan F. Karr, Xiaodong Lin, and Jerome P. Reiter. Privacy preserving regression modelling via distributed computation. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 677–682, New York, NY, USA, 2004. ACM.

[22] G. Simonyi. Graph entropy: A survey. In *Combinatorial Optimization*, volume 20, pages 399–441. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 1995.

[23] H. Theil. The analysis of disturbances in regression analysis. *Journal of the American Statistical Association*, 60(312):pp. 1067–1079, 1965.

[24] J. Vaidya, C. Clifton, and Y.M. Zhu. *Privacy Preserving Data Mining*. Springer, first edition, 2006.

[25] Andrew C. Yao. How to generate and exchange secrets. In *SFCS '86: Proceedings of the 27th Annual Symposium on Foundations of Computer Science*, pages 162–167, Washington, DC, USA, 1986. IEEE Computer Society.