Problem Chosen	2023	Team Control Number
C	MCM/ICM Summary Sheet	2320476

This paper applies machine learning models, specifically linear regression, ridge regression, and neural networks, to make concise predictions about Wordle results. The inputs of these models include word attributes and trends observed from particular dates. This paper then explains the mathematical foundations of each algorithm, how they were applied to the data set, and what resulting predictions were made. When testing the word EERIE on March 1, 2023, the reported results lie between 11,851 and 14,451, with an attempt distribution of 1%, 6%, 21%, 33%, 23%, 12%, 4%, and a difficulty skewness between 0.0594 and 0.7557. An analysis of the regression coefficients revealed that a word's commonality index has the highest impact on difficulty, while the proportion of vowels and number of repeats have the greatest impact on the percentage in hard mode. Similarly, the proportion in hard mode and number of days passed have the greatest effect on how many results are generated daily. As Wordle's popularity continues to steadily decline, these trends can be interpreted to maximize user engagement for future puzzles. This paper illustrates the practicality of machine learning methods applied to data sets.

An Application of Linear Regression and Neural Network Machine Learning: Predicting Wordle Results

Februrary 20, 2023

Contents

1 Abstract

Wordle is a five-letter game where players must find the word in 6 tries or less given information each try. For each guess, the game highlights which tiles representing individual letters were close to the day's word. Wordle became a public sensation, prompting users to create novel guessing strategies and post their results on social media.

A data set supplied by @WordleStats on Twitter includes attributes such as the number of reported daily results, the number of players in hard mode, and the attempt distribution for each word. This paper models and predicts the number of reported results, attempt distribution, percentage of results in hard mode, and difficulty of a given word on a particular date.

These predictions are derived from linear regression models and neural networks, specifically sequential neural networks, ridge regression models, and least squares linear regression methods.

Key predictions include reported results between 11,851 and 14,451, an attempt distribution of 1%, 6%, 21%, 33%, 23%, 12%, 4%, and a difficulty skewness between .0594 and .7557 on March 1, 2023 with the word EERIE.

This paper shows that linear regression and neural networks apply to this data set. The scope of these models can be extended to other real-world problems given more data points and attributes to sample from, propelling the growth and improvement of machine learning.

2 Introduction

Wordle is an online word game where players are given six guesses to figure out the five-letter word of the day [1]. Each guess must be a proper five-letter word. The color of each letter's tile changes based on how close the guess was. The tiles of letters not in the word stay grey. Tiles in the word but in the wrong place turn yellow. Tiles in the word and the right place turn green. All green tiles indicate that the word was guessed correctly. The player has six tries to guess the word and turn all the tiles green.

Josh Wardle created the game for his partner, Ms. Palak Shah, who loves word games [2]. After much satisfaction from their family and friends, he released it to the rest of the world in October 2021. It was a straightforward game, accessible to all on the internet, with no ads, barriers, or pop-ups. Soon after Wordle was released for public enjoyment, the word game became a sensation. On November 1, 2021, there were 90 reported players. Two months later, more than 300,000. The once-a-day nature of the game promotes scarcity and stimulating demand. In December, a feature was added where one could share results with other players, increasing the game's popularity.

Since the game was designed with his partner in mind, Ms. Palak went through a list of 12,000 five-letter words and narrowed down the universe of five-letter words to 2,500. In January 2022, New York Times acquired the game following a seven-figure agreement with Wardle. Upon acquisition, the words are no longer chosen from the creator's database but from New York Times's own [3]. Other changes included excluding plurals ending with "es" or "s."

After gameplay, users can share their results on Twitter. An emoji trend complements the craze–representing rows as green, yellow, and grey emojis to illustrate the scorecard [4]. Sharing results via Twitter prompted gathering self-reported data like guess distributions and success rates.

3 Data

3.1 The File

Twitter user @WordleStats collected user data on Twitter; MCM compiled a subset of these findings. The dataset comprises all daily Wordle results from January 7, 2022, to December 31, 2022. Each entry consists of:

- **Date of the puzzle**, in mm-dd-yy format
- The **contest number**, a reference to all Wordle puzzles since the conception of the game
- The **word**
- The **number of reported results** represented as the total number of scores from Twitter that day
- The **number in hard mode** represented by the number of scores that were recorded in hard mode on Twitter that day
- The **distribution of how many players solved the puzzle in how many guesses**, categories being one try, two tries, three tries, four tries, five tries, six tries, and seven or more tries

3.2 Data Cleaning

On 04-29-2022, the word listed was "tash." However, "trash" is a four-letter word and thus could not be Wordle's word of the day. Research into Wordle history revealed that the word for that day was "trash." [5] The correction was made.

On 10-05-2022, the word listed was "maxsh," a five-letter word but with an incorrect character. Research revealed it should be "marsh," and the correction was made.

On 11-16-2022, the word listed was "clen," another four-letter word. It was corrected to "clean."

On 12-11-2022, the word listed, "naïve," was written with a special character. It was corrected to be "naive." for word processing efficacy.

On 12-16-2022, the word listed was "rprobe," a six-letter word, which violates Wordle's 5-letter format. It was correct to "probe."

On 11-30-2022, an abnormality was noted. The number of reported players and the number of players in hard mode was considerably and unusually low, 2500. 93.62% of players were in hard mode that particular day. The data point was left in the set but treated as an outlier.

Furthermore, the dates were organized in descending order, from December 31, 2022, to January 07, 2022. The entries were reversed for analysis when considering the number of days that passed.

3.3 Additional Computation

Additional variables were computed using the data.

- **Day of Week** Each day was assigned a value of 1 through 7 to indicate the day of the week, 1 being Sunday and 7 being Saturday
- **Weekend** A value of 0 if it is not a weekend, and a value of 1 if it is a weekend
- **Month** The month of each entry
- **Proportion in Hard Mode** The proportion in hard mode was calculated by dividing the number in hard mode by the number of reported results
- **Number of Vowels** The number of vowels per word was extracted, vowels being considered "a," "e," "i," "o," and "u."
- **Number of Consonants** The number of consonants was found by subtracting the number of vowels from 5
- The **Proportion of Vowels** The proportion of vowels was found by dividing the number of vowels by the number of consonants
- **Skew** The perceived difficulty of a word is determined by the skew of the guess distribution. If the guess distribution is right-skewed (tending towards 1/2/3 tries), the word is considered easy to guess; if the guess distribution is left-skewed (tending towards 5/6/7 or more tries), the word is considered hard to guess. The following ranges are defined: $<$-0.5 for difficult, -0.5 to 0.5 for moderate, and $>$0.5 for easy [6].

3.4 Analysis

As seen in Figure 1, the histograms were constructed to create distributions for the number of guesses to understand existing trends.

Solving the Wordle in 1 try is rare. Results range from 0 to 6% of the reported results. The median is 0%. Values most frequently around 1-2%.

Solving the Wordle in 2 tries is more common. Distribution ranges from 0 to 26% but is greatly right-skewed. The median is 5%. Values most frequently around 2-6%.

For solving the Wordle in 3 tries, distribution ranges from 4% to 47%, looking somewhat normal. Values most frequently around 22.7%, plus or minus 6.73%.

For solving the Wordle in 4 tries, distribution ranges from 11% to 49%, looking somewhat normal, with slight left skewness. Values most frequently around 32.9%, plus or minus 4.07%.

For solving the Wordle in 5 tries, distribution ranges from 9% to 44%, looking somewhat normal. Values most frequently around 23.6%, plus or minus 5.36%.

For solving the Wordle in 6 tries, distribution ranges from 2% to 37%, with most values at the low end. The median is 10%.

For solving the Wordle in 7 tries or more, the distribution is highly left skewed, ranging from 0% to %. The median is %. Outliers include 20%, 23%, 26%, and 48%.

Most curves follow a relatively normal path when each word's guess distribution is plotted against each other. However, some curves follow a right or left skew, but not many.

Throughout time, reported daily results grew rapidly, peaking at 361,908 in February of 2022, then dropping gradually. Quantities range from 2,569 to 361,908. The median is 44,400.

The proportion of players in hard mode ranges from 1.17% to 93.62%, with that 93.62% being seen as an outlier. Most values are concentrated on the lower end. The median is 8.34%, and the mean is 7.76%.

30.36% of words have one vowel. 60.45% of words have two vowels. 9.19% have three vowels.

Of the words, 25.1% contain one or more of the same letter.

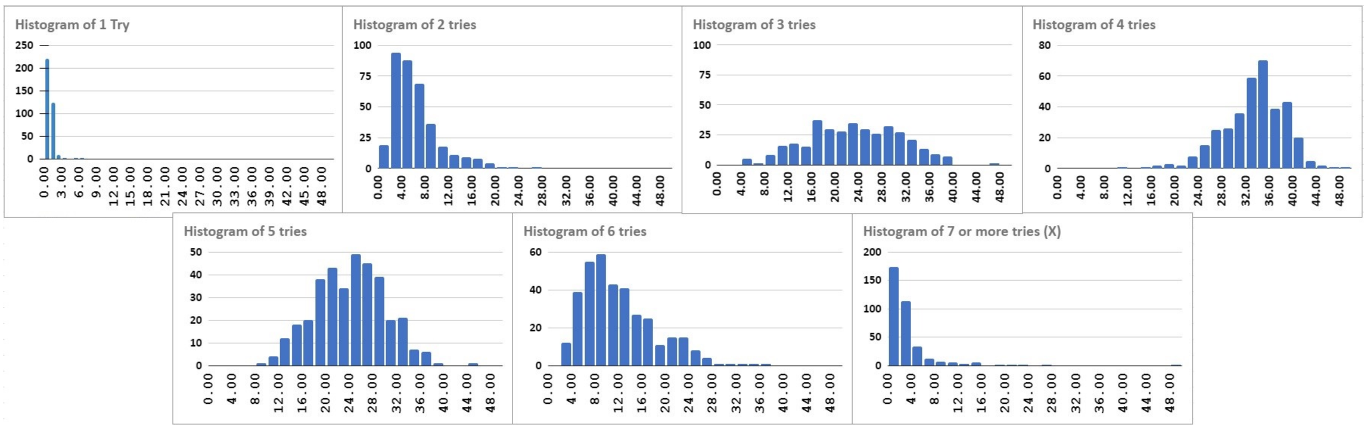

Figure 1: Histograms of frequency of number of attempts

28.69% of the Wordles occurred on the weekend.

4 Methods

4.1 Linear Regression

Linear regression is principally based on the equation:

$$Y = Xw + b \tag{1}$$

Where Y is the output, X is the input, w is a weight value, and b is a bias the function must overcome before outputting Y. Least squares regression is a method for fitting a linear model to a data set by minimizing the sum of the squared differences between the predicted values and the actual values of the response variable [7]. The objective is to find the values of the model coefficients that minimize the sum of the squared residuals, which is given by the cost function [8]:

$$J(\beta) = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_{i1} - \cdots - \beta_p x_{ip})^2 \tag{2}$$

Where n is the number of observations, p is the number of predictors, y_i is the response variable for observation i, x_{ij} is the value of predictor j for observation i, and $\beta_0, \beta_1, \ldots, \beta_p$ are the model coefficients.

The objective is to find the values of the model coefficients that minimize the cost function. This is done by taking the partial derivatives of the cost function with respect to each of the coefficients, setting them equal to zero, and solving for the coefficients. The resulting equations are known as the normal equations [9]:

$$\frac{\partial J(\beta)}{\partial \beta_j} = -2 \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_{i1} - \cdots - \beta_p x_{ip}) x_{ij} = 0, \quad j = 0, 1, \ldots, p \tag{3}$$

These equations can be rewritten in matrix notation as:

$$\mathbf{X}^\top \mathbf{X} \boldsymbol{\beta} = \mathbf{X}^\top \mathbf{y} \tag{4}$$

Where $\mathbf{X}$ is the design matrix, which contains the values of the predictors for each observation, and has dimensions $n \times (p+1)$, $\mathbf{y}$ is the vector of response values, with dimensions $n \times 1$, and $\boldsymbol{\beta}$ is the vector of coefficients, with dimensions $(p+1) \times 1$ [10].

The solution to the normal equations is:

$$\boldsymbol{\beta} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y} \tag{5}$$

This equation gives the values of the coefficients that minimize the cost function. Once the coefficients are known, the model can be used to predict the response variable for new observations using the equation:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p \tag{6}$$

where $\hat{y}$ is the predicted response, and $x_1, \ldots, x_p$ are the predictor values for the new observation.

The objective of ridge regression is to find the coefficients, denoted by β, that minimizes the following cost function [11]:

$$J(\beta) = \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} x_{ij} \beta_j)^2 + \alpha \sum_{j=1}^{p} \beta_j^2 \tag{7}$$

Here, n is the number of observations, p is the number of predictors, y_i is the response variable for observation i, x_{ij} is the value of predictor j for observation i, and α is a non-negative hyperparameter that controls the strength of the penalty term. The penalty term is the sum of the squared coefficients multiplied by the hyperparameter α.

The first term in the cost function is the sum of squared errors, which is the same as in the least squares regression. The second term is the penalty term, which is used to shrink the coefficients toward zero, thus reducing the impact of multicollinearity [12].

4.2 Neural Network

A neural network is a machine learning model inspired by the brain's structure and function. It consists of multiple layers of interconnected nodes, or neurons, that perform a series of computations on the input data to produce an output[13]. As seen in Figure 2, each node is connected to each subsequent node in the layer following. The connecting of these nodes is represented by the standard linear regression equation 1.

Let X be the input data as a vector, and let $f(X;\theta)$ be a neural network with two hidden layers using ReLU activation. The first hidden layer, also known as the feature extraction layer, takes the input data and produces a set of features:

$$\vec{H_1} = ReLU(W_1\vec{X} + \vec{b_1}) \tag{8}$$

where W_1 is a matrix of weights, $\vec{b_1}$ is a vector of biases, and $ReLU(x) = max(0,x)$ is the ReLU activation function. The ReLU function is a piece-wise defined linear function that is defined as zero for negative inputs and zero, and linear for positive inputs. It has the advantage of being computationally efficient, avoiding the vanishing gradient problem, and introducing non-linearity into the neural network.

The second hidden layer, also known as the classification layer, takes the features from the first hidden layer and produces a set of output scores:

$$\vec{H_2} = ReLU(W_2\vec{H_1} + \vec{b_2}) \tag{9}$$

where W_2 is another matrix of weights, $\vec{b_2}$ is another vector of biases, and ReLU(x) is the ReLU activation function.

Finally, the output layer takes the scores from the second hidden layer and produces a probability distribution over the possible classes:

$$\vec{Y} = softmax(W_3\vec{H_2} + \vec{b_3}) \tag{10}$$

where W_3 is a matrix of weights, $\vec{b_3}$ is a vector of biases, and

$$softmax(x_i) = \exp x_i / \sum_j \exp x_j \tag{11}$$

is the softmax function, which ensures that the outputs form a valid probability distribution.

To train the network, a loss function that measures the discrepancy between the predicted output distribution and the true label distribution needs to be defined. A popular choice is the Means Squared Error or the Log Mean Squared Error ($-log$), which are defined as:

$$L = MSE(y) = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2 \tag{12}$$

$$L = LMSE(y) = \frac{1}{m}\sum_{i=1}^{m}(log(y_i{+}1){-}log(\hat{y}_i{+}1))^2 \tag{13}$$

where y_i is the i-th element of the true label distribution, and $\hat{y}_i$ is the i-th element of the predicted output distribution[14].

To train the network, backpropagation is used, which is an algorithm that computes the gradients of the loss with respect to the parameters of the network [15]. The gradients are then used to update the weights and biases using an optimization algorithm such as stochastic gradient descent (SGD) or Adam. The following equation defines this:

$$\vec{S_{t+1}} = \vec{S_t} - r\nabla L \tag{14}$$

Where S is the vector of weights and biases, r is the Learning Rate or Coefficient of Propagation and L is the loss function.

Over multiple iterations, this SGD approaches a local minimum for the loss function, which in turn optimizes the accuracy of the predicted output (y_i) against the actual output $(\hat{y}_i)$. All local minimums will converge to a relatively similar minimum due to the topology created by this neural network[16].

The optimal number of nodes per layer in a two-hidden-layer neural network is a function dependent on the sample size of inputs and the number of output nodes[17]. The number of nodes for layer one is

$$N_1 = \sqrt{(m+2)N} + 2\sqrt{\frac{N}{m+2}} \tag{15}$$

The number of nodes for layer two is

$$N_2 = m\sqrt{\frac{N}{m+2}} \tag{16}$$

Where N is the sample size of input data and m is the number of output nodes.

4.3 Limitations

Linear regression is a very common and straightforward way to model a dependent variable versus one or more independent variables. Despite their widespread use and ease of use, linear regression models have some drawbacks that can affect how well they function when used with smaller training sets.

The assumption of linearity between the dependent and independent variables is one of the fundamental drawbacks of linear regression models. This indicates that a linear relationship between the variables is presumed,

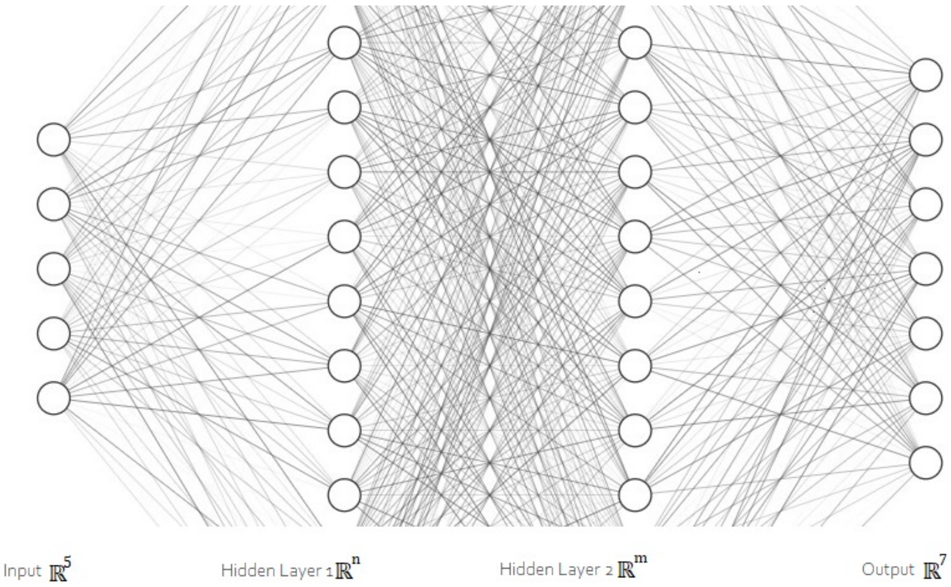

Figure 2: Graphic showing the architecture of arbitrary neural network with 5 input nodes, 7 output nodes, and two hidden layers with arbitrary n and m nodes

even though this may not always be true in real-world situations. For instance, a linear regression model might not be able to fully represent the underlying patterns in the data if the real connection between the variables is non-linear.

Another limitation of linear regression models is their susceptibility to overfitting or underfitting the data. Overfitting occurs when the model is too complex and captures noise or irrelevant patterns in the training data, leading to poor generalization performance on new data over multiple iterations [18]. Underfitting occurs when the model is too simple and cannot capture the true patterns in the data, leading to poor performance on both the training and test data.

When applied to small training sets, these limitations of linear regression models can become more pronounced. The small amount of training data may not provide enough information to accurately capture the true relationship between the variables. This can lead to overfitting or underfitting, as the model may not have enough information to learn about the underlying patterns in the data.

Small training sets may also be more susceptible to bias and noise, which can impact the accuracy of the model[19]. For example, if the training set is biased towards a particular subset of the population, the resulting model may not generalize well to other populations. Similarly, if the training data contains a high degree of noise or measurement error, this can impact the model's accuracy and lead to poor performance on new data.

Since the more complex neural network model is, at its heart, based on the mapping of the linear regression between the nodes, the same limitations are prevalent in the neural network. This is especially true when discussing small sample sizes and training data. On top of this, neural network models are infamous for their inability to learn properly from smaller training sets. Although the neural network is learning at an exponentially faster rate than just memorizing the data, this can be limited by a smaller sample size. This can lead to overfitting; over many epochs, the neural network will memorize the data rather than learning attributes that have been discovered through backpropagation.

5 Modeling

The following model uses elements of a standard linear regression model and a sequential two-layer neural network to analyze and make predictions on the data at hand. These methods in tandem, allow for a more holistic view of the data and more accurate predictions for a given desired output. This model is able to predict the number of reported results for a future date, predict the attempt distribution for a given word on a future date, explain and give attributes of a word that makes it more difficult, quantify the difficulty of a given word, and explain the relationship between word attributes and percentage of people that play in hard mode.

5.1 Model Formulation

5.1.1 Predicting Number of Reported Results

Ridge Regression is used to predict the number of reported results given an arbitrary day.

This model relies on six predictors - days passed since 1/7/2022, months passed since 1/7/2022, weekday (1-7), weekend as an indicator (0/1), the proportion of results in hard mode, and difficulty of the word based on skew. Because these variables are coded on

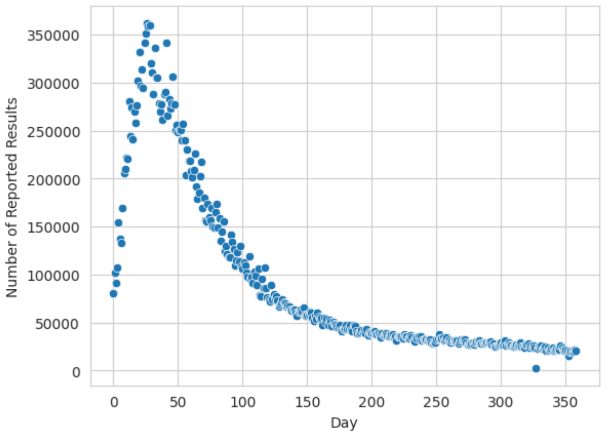

Figure 3: Plots Number of Days Passed Since 1/7/2022 against Number of Reported Results

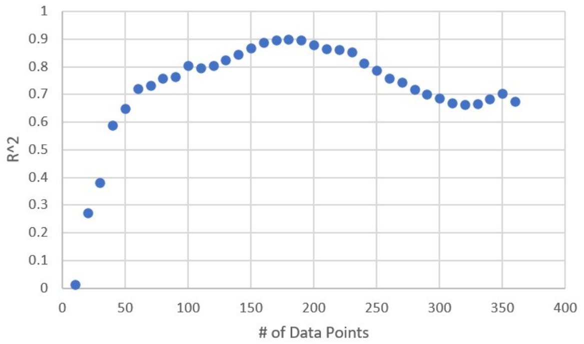

Figure 4: Plots the R^2 of a linear fit to the data against Number of Data Points Since 12/31/22

different scales, each predictor variable is standardized using min-max scaling so that all inputs to the model are assigned a value between 0 and 1.

The ridge regression model is then fit with alpha = 1.0, to increase the impact of the loss function and keep all parameters within a boundary so that the model is not biased towards a particular parameter.

Before splitting the data, the number of days that passed since January 7, 2022, was compared to the number of reported results, shown in Figure 3. Because this relationship is exponential for the first few weeks, the data points are linearly fit in groups of 10 and found their corresponding R^2 values. As shown in Figure 4, R^2 is maximized at around Day 180, meaning that the model is best suited to be linearly fit only on the last 179 days of the data set. Thus, only attributes from these days were included in this model.

The data is split into testing and training data, but there are several ways to make this division. Therefore, this model starts by looping through several test sizes, with the size representing the proportion of the data being used as test data rather than training data. Test sizes from 0.10 to 0.90 in increments of 0.05 are tested on the model, along with their corresponding errors represented by the coefficient of determination R^2. After plotting each test size versus R^2, the test size that maximizes R^2 to increase the accuracy of our model is 0.25.

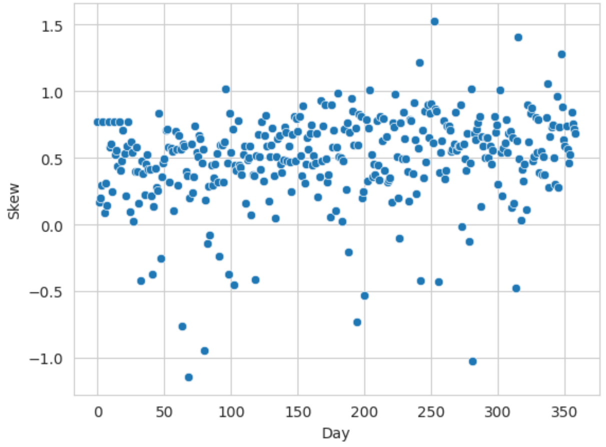

Figure 5: Plots the Skew of Attempt Distribution against the day in order.

Once the optimal test size is determined, the model is run with 1000 repetitions. The six regression coefficients and intercepts are stored after every run; the final regression coefficients are the average over these 1000 runs.

The predicted number of results for March 1, 2023, is determined by passing the attributes of that day (days passed, months passed, weekend, weekday) through the linear equation constructed from the coefficients and y-intercept obtained. The same process can be followed to obtain predictions for any arbitrary day after 1/7/2022, the first date provided in the data set.

5.1.2 Predicting Attempt Distribution

A key assumption made for the model predicting attempt distribution was that the day the word occurred has no proportional impact on the attempt distribution. As seen in Figure 5, there is no common trend or relationship between the day of the week or date that it occurred, and the attempt distribution skew. This leaves only the contents of the word having an affected relationship with the attempt distribution.

A neural network is used to create a prediction for the attempt distribution. This is done using 5 inputs, the letters in sequential order, and 7 outputs, the distribution as a percentage (1 try, 2 tries. . . etc). Since the neural network using the ReLU activation intakes only values between 0 and 1, this means inputs and outputs have to be scaled to achieve efficient results. The letters of the word are min-maxed by setting $a = 0$ and $z = 26$ and dividing through by 26. This allows the alphabet to linearly increase and be normalized. The given percentages for the output also had to be scaled by 0.01 to have a value that would be between 0 and 1.

This model uses 2 hidden layers as the

algorithm is relatively simple while also encompassing the complexity of the problem. The first layer has 69 nodes, this is calculated using equation 15. The second layer has 44 nodes, calculated using equation 16. Both hidden layers use ReLU activation. The output layer is calculated using softmax activation, seen in equation 11, to produce a probability distribution.

This model is compiled using a loss function of MSE as the difference in accuracy between MSE, and LMSE is negligible, and MSE is more efficient. The backpropagation used is adam, a form of SGD. This form of the model is iterated on itself for 3 epochs.

Due to variations between each individual learning run of the neural network, a prediction is made 1000 times and averaged across the iterations.

The data supplied was partitioned into 83.5% training data and 16.5% testing data to calculate the evaluation of the model. After evaluation of the against test data, the accuracy reached 67.8%.

5.1.3 Predicting Difficulty from Word Attributes

Least Squares Linear Regression is used to predict the perceived difficulty of guessing a word given various attributes of that word.

This paper analyzes the effects of 5 different word attributes in this model: proportion of vowels, the number of repeat letters [20], if the word starts with one of five most common letters (0/1), if the word ends with one of five most common letters (0/1), and the commonality index, a weighted average of character frequencies for each word [21].

The skew of the guess distribution for the word represents the difficulty of the word. As the skew becomes more negative (smaller than -0.5), the guess distribution becomes more left-skewed, indicating that the word is harder to guess. As the skew becomes more positive (greater than 0.5), the guess distribution becomes more right-skewed, indicating that the word is easier to guess.

As with the ridge regression model, the data is split into testing and training data. Test sizes from 0.10 to 0.90 in increments of 0.05 are tested on the model, along with their corresponding errors represented by the root mean squared error, or RMSE. After plotting the test sizes versus RMSE, the data is partitioned with the test size that minimizes error, which is 0.3.

Once the optimal test size is determined, the model is run with 1000 repetitions. The five regression coefficients and intercepts are stored after every run; the final regression coefficients are the average over these 1000 runs.

The difficulty for a word can be found by entering its attributes as inputs to the linear regression equation produced by the model. For example, the word EERIE is encoded by the array [0.8, 3, 0, 1, 0.4291], with each item representing the proportion of vowels, number of repeats, starting with a common letter, ending with a common letter, and commonality index respectively. The output estimates the skew of the word's guess distribution, representing difficulty.

5.1.4 Predicting Percent in Hard Mode from Word Attributes

Least Squares Linear Regression is also used to predict the percentage of results in hard mode given various word attributes.

The same five word attributes explored previously as inputs to this model - the proportion of vowels, number of repeat letters if the word starts with one of five most common letters (0/1), if the word ends with one of five most common letters (0/1), and the commonality index, a weighted average of character frequencies for each word - are implemented.

This model first loops through several test sizes, following the same process outlined in the previous two models. Test sizes from 0.10 to 0.90 in increments of 0.05 are tested on the model, along with their corresponding errors represented by the RMSE. After plotting the test sizes versus RMSE, the data is partitioned with the test size that minimizes error, which is 0.25.

Once the optimal test size is determined, the model is run with 1000 repetitions. The five regression coefficients and intercepts are stored after every run; the final regression coefficients are the average over these 1000 runs.

The sign of each regression coefficient determines the impact of the word attribute on the percentage of results in hard mode. A negative coefficient indicates that when the word attribute is more prevalent, the percentage in hard mode tends to decrease; a positive coefficient indicates that when the word attribute is more prevalent, the percentage in hard mode tends to increase.

6 Results

6.1 Number of Reported Results

This paper predicts that the average number of reported results is 13,151 for March 1, 2023, rounded to the nearest whole number.

With a mean of 13,151 reported results and a standard deviation of 650 results, this model is 95% confident that the true number of reported results for March 1st, 2023, will lie between 11,851 and 14,451 results.

Table 1: Coefficients for Reported Results

Coefficient Name	**Value**
Day	-117.9168
Proportion in Hard Mode	-38101.8918
Weekday	59.1378
Month	72.6281
Workday	-915.0335
Difficulty	113.1080

6.2 Attempt Distribution

The guess distribution this papers predicts for the word EERIE in the form (1 try, 2 tries, 3 tries, 4 tries, 5 tries, 6 tries, and 7 or more tries) is 0.010909, 0.060588, 0.208171, 0.328472, 0.232481, 0.120825, 0.038551 respectively. The guess distribution in percentages resemble 1%, 6%, 21%, 33%, 23%, 12%, 4%.

The accuracy of this model when compared to the test data set is 67.8%. This implies the proportion of variance in the data explained by the model is 67.8%. This leaves relative confidence in the model to be able to predict the attempt distribution with 1-2 percentage points for each try.

6.3 Difficulty From Word Attrs

Table 2: Coefficients to Predict Difficulty

Coefficient Name	**Value**
Proportion of Vowels	0.0176
Number of Repeat Letters	-0.0094
If Starts With Top 5 Letter	0.0134
If Ends With Top 5 Letter	-0.0249
Commonality Index of Letters	-0.1306

When entered into the model, the word EERIE has a predicted guess distribution with a skew of 0.6370 $>$ 0.50, classifying EERIE as an easy word to guess. With a mean skew of 0.6370 and a standard deviation of 0.0593, this model is 95% confident that the perceived difficulty of the word EERIE falls between 0.5182 and 0.7557, which still categorizes EERIE as an easy word to guess.

6.4 Percent in Hard Mode

Table 3: Coefficients for Hard Mode

Coefficient Name	**Value**
Proportion of Vowels	-1.4098
Number of Repeat Letter	-1.3215
If Starts With Top 5 Letters	-0.5038
If Ends With Top 5 Letters	-0.1429
Commonality Index of Letters	0.4252

From Table 6.3, as the proportion of vowels increases, the percent in hard mode decreases by about 1.4% for every unit change.

As the number of repeats increases by 1 letter, the percentage in hard mode decreases by about 1.3% for every unit change.

If the word starts with or ends with a common letter, the percentage in hard mode decreases slightly, by less than 1% per unit change.

If the commonality index increases, the percentage in hard mode increases slightly by 0.4% per unit change.

Based on these outcomes, it is concluded that words with a higher commonality index increase the percentage of players that choose hard mode. Other word attributes like having a higher proportion of vowels, number of repeats, and ending or starting with a common letter are associated with a lower percentage of players in hard mode.

7 Conclusions

This paper applies linear regression, ridge regression, and neural network machine learning to predict Wordle results for a given word on a given day. On March 1st, 2023, the game is expected to have between 11,851 and 14,451 reported results, computed with 95% confidence. Given the word EERIE on the same day, the guess distribution is predicted to be [1%, 6%, 21%, 33%, 23%, 12%, 4%] for (1 try, 2 tries, 3 tries, . . . , 7 or more tries) respectively, with 67% accuracy. The predicted difficulty for EERIE, represented by the skewness of this attempt distribution, is between 0.5940 and 0.7557 with 95% confidence, categorizing this word as easy to guess.

Apart from making predictions with specific words and dates, this paper also identifies predictors for metrics like the number of reported results, difficulty, and percentage in hard mode. The effects of these predictors are determined from the regression coefficients of three separate models.

The simple machine learning algorithms from this research can be applied to other real-world data sets by extracting tangible predictions and attributes of a given dependent variable. The choice of which model to use depends on the nature of the data and the necessary prediction at hand. Linear regression is a powerful tool to analyze linear relationships between variables, and neural networks can model more complex relationships in predicting target variables; the choice of which model to use depends on the nature of the data and the necessary prediction at hand. This is a step forward for machine learning models and yet another application of linear regression and neural networks.

Overall, this study contributes to the growing body of literature on machine learning applications in scientific research and has potential implications for fields such as healthcare, social sciences, and data analytics.

To the New York Times Puzzle Editor:

Our consultation was requested in analyzing Wordle Data collected by @WordleStats from Twitter. First, we would like to thank you for requesting our services. We take our duties in analytics and modeling very seriously, and our clients are of utmost importance. Working with The New York Times has been an absolute pleasure, and we hope our findings bring value to your work.

We began by developing a model that explains variation in the number of reported scores daily, and we tested this on March 1, 2023. Our model relies on ridge regression with six predictors - days passed since 1/7/2022, months passed since 1/7/2022, weekday, weekend, the proportion of results in hard mode, and word difficulty. After passing the attributes for March 1, 2023, through our model, we would expect Wordle to have between 11,851 and 14,451 reported results, with 95% confidence.

We observed a steady decrease in the number of reported results as each day passes from the beginning of 2022. This trend is predicted to continue throughout 2023, so you may expect less user engagement for Wordle as time progresses. While this pattern may seem alarming, we hope that our further analyses of word attributes may give you more insight into how to tackle this issue.

To explore how a word's attributes may affect the difficulty and, therefore, the success rate, we identified five predictors - the proportion of vowels, the number of repeated letters, starting letters, ending letters, and the commonality index. We created our own commonality index by finding each word's weighted average of character frequencies. These attributes were then used to train our Least Squares Linear Regression Model to predict difficulty. We also quantified difficulty based on the skewness of the guess distribution.

We were surprised that one of our test words, EERIE, was considered easy to guess with a positive skew between 0.0594 and 0.7557, computed with 95% confidence.

An analysis of the regression coefficients reveals that a word's commonality index has the highest impact on difficulty. In contrast, the proportion of vowels and the number of repeats substantially affect the percentage of players who complete hard mode.

Apart from word attributes, we also attempted to find a pattern between the date and guess distribution, which tied into difficulty. After conducting some initial analysis, we realized that playing on a particular day does not significantly impact the guess distribution enough to be considered a relevant factor. Therefore, the distribution of guesses should look similar on any given day.

Since we couldn't find particular factors influencing how people guess each word, we used a neural network to predict the guess distribution. The model takes in the letters of a word in sequential order and produces a guess distribution with 67% accuracy. Two hidden layers within the network help increase accuracy and find factors that affect the guess distribution. We found that the guess distribution for EERIE is expected to be 1%, 6%, 21%, 33%, 23%, 12%, and 4% for (1 try, 2 tries, 3 tries, . . . , 7 or more tries) respectively.

Thus, on March 1st, 2023, we expect between 11,851 and 14,451 people to play until completion with approximately 1% guessing in one try, 6% guessing in two tries, 21% guessing in three tries, 33% guessing in four tries, 23% guessing in five tries, 12% guessing in six tries, and 4% in seven tries.

We hope these findings were to your liking. We would like to bring attention to decreasing user engagement rates as you consider all these factors. Consider these observations when crafting a new word game to join Wordle in the ranks of the New York Times's legendary puzzles!

Best regards,

Team 2320476

References

[1] *Wordle - a daily word game.* URL: `https://www.nytimes.com/games/wordle/index.html`.

[2] Daniel Victor. *Wordle Is a Love Story: The word game has gone from dozens of players to hundreds of thousands in a few months. It was created by a software engineer in Brooklyn for his partner.* Jan. 2022. (accessed: 02.16.2023).

[3] *New York Times announces changes to Wordle.* Jan. 2022. (accessed: 02.16.2023).

[4] Naomi Tomky. *History of Wordle: The humble story of a cultural phenomenon.* Jan. 2022. (accessed: 02.16.2023).

[5] Caitlin Welsh. *Wordle today.* 2022. (accessed: 02.16.2023).

[6] Suddhendu Biswas. *Topics in Statistical Methodology.* John Wiley Sons, 1991. ISBN: 9780470211533.

[7] Mark Lunt. "Introduction to statistical modelling: linear regression". In: *Rheumatology* 54.7 (Apr. 2013), pp. 1137–1140.

[8] Xin Yan and Xiaogang Su. "Linear Regression Analysis: Theory and Computing". In: 2009.

[9] T. Daniya, Dr Kumar, and Cristin R. "Least Square Estimation of Parameters for Linear Regression". In: *International Journal of Control and Automation* 13 (Apr. 2020), pp. 447–452.

[10] G.S. Watson. "Liner Least Squares Regression". In: *The Annals of Mathematical Statistics* 38 (Dec. 1967), pp. 1679–1699.

[11] Gary McDonald. "Ridge regression". In: *Wiley Interdisciplinary Reviews: Computational Statistics* 1 (July 2009), pp. 93–100. DOI: `10.1002/wics.14`.

[12] Hanan Duzan and Nurul Sima Mohamad Shariff. "Ridge Regression for Solving the Multicollinearity Problem: Review of Methods and Models". In: *Journal of Applied Sciences* 15 (Mar. 2015), pp. 392–404.

[13] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: June 2016, pp. 770–778.

[14] Bo Pang, Erik Nijkamp, and Ying Nian Wu. "Deep Learning With TensorFlow:A Review". In: *Journal of Educational and Behavioral Statistics* 45.2 (2020), pp. 227–248.

[15] Shun-ichi Amari. "Backpropagation and stochastic gradient descent method". In: *Neurocomputing* 5 (1993), pp. 185–196.

[16] Anna Choromańska et al. "The Loss Surfaces of Multilayer Networks". In: *International Conference on Artificial Intelligence and Statistics*. 2014.

[17] Guangbin Huang. "Learning capability and storage capacity of two-hidden-layer feedforward networks". In: *IEEE transactions on neural networks* 142 (2003), pp. 274–81.

[18] Michael A. Babyak. "What You See May Not Be What You Get: A Brief, Nontechnical Introduction to Overfitting in Regression-Type Models". In: *Psychosomatic Medicine* 66 (2004), pp. 411–421.

[19] Olivier Chapelle, Vladimir Vapnik, and Y. Bengio. "Model Selection for Small Sample Regression". In: *Machine Learning* 48 (Dec. 2000).

[20] Matiss Rikters and Sanita Reinsone. "How Masterly Are People at Playing with Their Vocabulary?" In: *Baltic J. Modern Computing,* 10.3 (2022), pp. 382–391.

[21] Nisansa De Silva. "Selecting Optimum Seed Words for Wordle using Character Statistics". In: *2022 Moratuwa Engineering Research Conference (MERCon)*. 2022, pp. 1–6.