Linear Regression Analysis on Free Fall Data

Sarah Nicole Straw

1st Febuary 2025 — Word count: 2482

Abstract

This report investigates the use of linear regression to predict the time of fall for spheres dropped from varying heights, The model was fitted to the dataset, with it's performance evaluated by calculating the Mean Squared Error (10.543) and analyzing the residuals. The regression coefficients from the regression model were found to be 22.220, -6.366, -7.490, 4.046, -0.352, 0.275 and 3.638.

A comparison of the predicted fall times of the model with theoretical times was also performed, the model was very accurate when fitted to specific materials of one radius, but predictions were squandered when several radii were included, as this has a heavy weight of the resulting time of fall. The residual histogram showed a roughly symmetric distribution. This report highlights the effectiveness of linear regression in predicting fall times while identifying areas for potential improvement.

1 Introduction

In this exercise the dynamics of free fall are explored by the application of linear regression. The model is used to predict the time of fall for spheres dropped from various heights. Other variables are included such as material, density, air pressure, ambient temperature as well as mass and radius of the sphere. A large data set is filtered, removing lines with missing or non-numerical values. A multiple linear regression model is fitted to the valid data, having 6 independent features fitted to associated the time of fall values. The weight of each variable on the fall time is displayed with a correlation matrix as well as the regression coefficients of the fitted model.

The model is compared to two other alternative loss functions by their respective coefficients. Three alternative methods of acquiring the regression coefficients are done, batch, stochastic and mini-batch gradient decent methods. The final value of the loss function and the coefficients are compared.

Finally, the model is tested in regions of drop height outside of the range of provided data, along with the theoretically calculated drop times in those regions. The performance of the model is evaluated with residuals, this being the difference between the model and theoretically predicted drop heights. The standard deviation and squared deviation are also calculated, these give insight into whether the model provides accurate predictions of free fall experiments. The patterns in the residuals are seen from histogram and scatter plots, which also show if the model is accurate to the data.

2 Part 1: Reading and verifying data

To read the data from a text file first the whitespace is removed and values are split up at each comma, then lines with missing data and 'nan' values are removed. Each line removed is counted, resulting in 78 removed lines.

The data is split into arrays for each material, variables are plotted against each other to display any patterns or dependency between them.

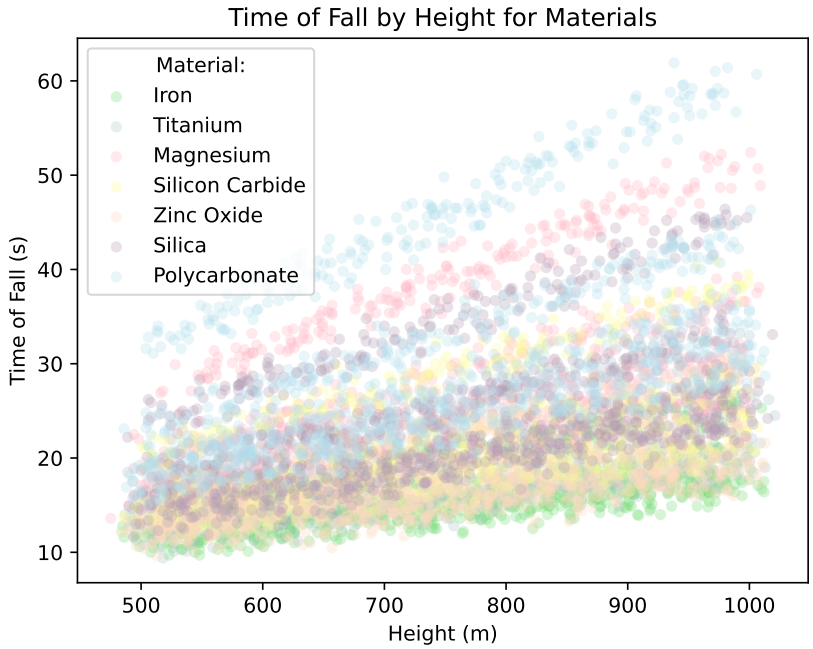

Figure 1: Time of drop plotted against height with each material colour coded, showing a general trend upwards in time with height. The highest time of fall values are from polycarbonate, due to it being the least dense while the most dense meterial, iron, has the lowest fall times.

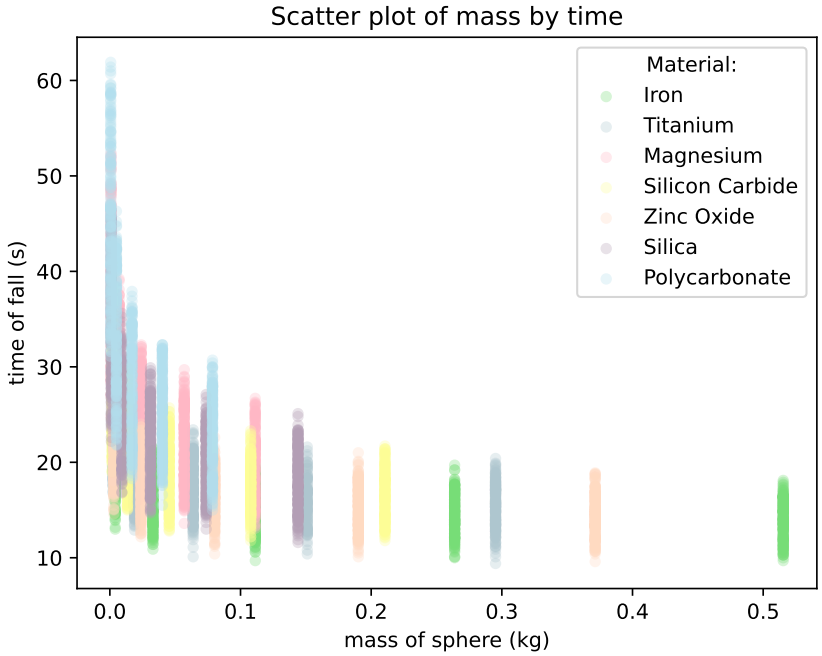

Figure 2: Mass of sphere plotted against time for every material. The overall trend appears to be exponential decay, this follows theory as shown in equation (7).

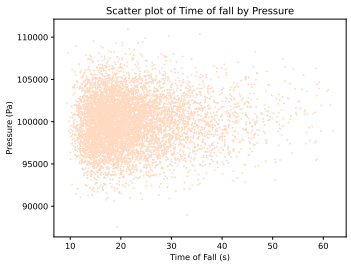

Figure 3: Scatter Plot of time of fall by air pressure for all data points. This shows no correlation between material type and pressure.

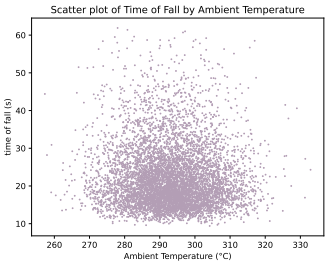

Figure 4: Scatter plot of time of fall by ambient temperature, also no correlation with material type.

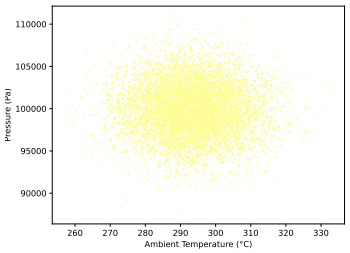

Figure 5: Scatter plot of ambient temperature by air pressure, resulting in a circular scatter plot around the average value of the variables.

3 Part 2: Correlation matrix

The correlation coefficient ranges from -1 to 1, for coefficients close to ± 1 signify a strong linear relationship where as one variable increases the other decreases (for -1) or increases (for +1). For values close to 0 there's no linear relationship however the variables could be related in a non-linear manner.

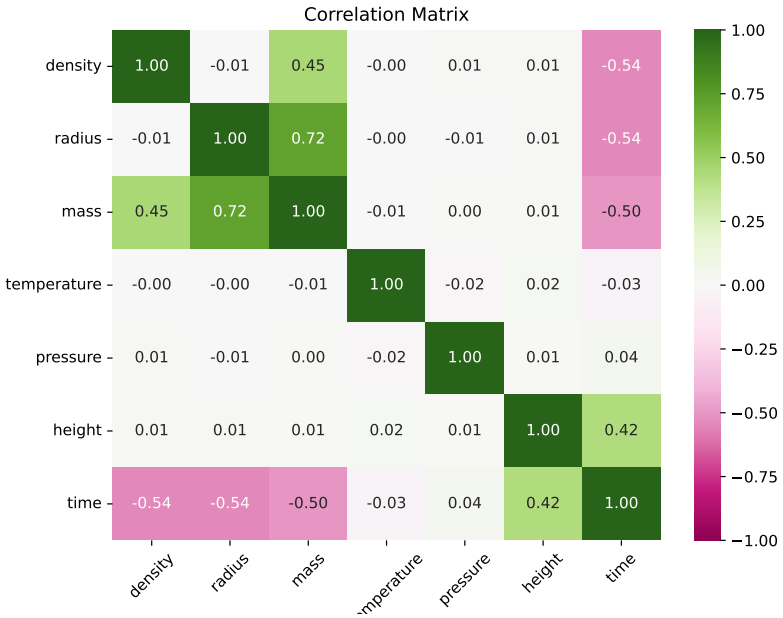

Figure 6: Correlation Matrix of all 7 variables, displaying the correlation coefficients between each combination of variables.

Some correlations are clearly physically explained, for example mass has strong linear relation to density and radius when these increase logically mass will also. However it has a negative relation to time, this makes sense as heavier spheres will accelerate faster and so have a smaller drop time. As mass has a negative relation to time but positive relation to radius and density it isn't surprising that these two variables also have a negative relation with time of fall. This also makes sense physically as denser, larger spheres will be heavier and so have a decreased drop time.

4 Part 3: Multiple linear regression

In this section a linear regression model is fitted to the data. The loss function of this is the standard least-squares analysis:

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

Where N is the number of data points, y_i is the actual time of drop and $\hat{y}_i$ is the model predicted time of drop from the other variables, for the i^{th} data point or row in the data provided.

The model predicted time values (y_i) can be calculated by the regression coefficients β in:

$$y_i = \beta_0 + \sum_{j=1}^{N}\beta_j X_{ij} + \epsilon_i \tag{2}$$

Where X_{ij} is the data matrix of i data points, or rows, and j features, or columns, in this case 6. In this case y_i is a time value, so in full this becomes:

$$t_i = \beta_0 + \beta_1\rho_i + \beta_2 r_i + \beta_3 m_i + \beta_4 T_i + \beta_5 p_i + \beta_6 h_i + \epsilon_i \tag{3}$$

The solution to this, to calculate the regression coefficients is:

$$\boldsymbol{\beta} = \left(X^T X\right)^{-1} X^T y_i \tag{4}$$

4.1 Part 3 (a): MSE loss function

The fitted linear regression model produced these regression coefficients, each aligning to a certain variable:

Coefficient	**Value**
β_0 (Intercept)	22.220
β_1 (Density)	-6.366
β_2 (Radius)	-7.490
β_3 (Mass)	4.046
β_4 (Temperature)	-0.352
β_5 (Pressure)	0.275
β_6 (Height)	3.638
Mean Squared Error	**10.543**

Table 1: Updated Regression Coefficients and Mean Squared Error

Variables with a strong positive correlation to time tend to have high regression coefficients, as seen with height, which aligns with theory as a higher drop height means a longer distance to fall and so a longer time to do so. This is seen in the model by the high regression coefficient of 3.638. The intercept β_0 is the predicted time when every variable value is zero, this is clear in equation (4). Although this is obviously not a real situation this intercept is more of a baseline for the entire regression model.

However, some variables have very different coefficients than their correlation matrix coefficient with time would suggest. This is likely due to multicollinearity, where highly correlated variables lead to unstable regression coefficients.

From the correlation matrix, it is clear that density, radius, and mass are strongly correlated, which makes sense physically since density is defined as mass per unit volume, which depends on the sphere's radius. As a result, the regression coefficients for density and radius have significant difference despite their exactly equal correlation with time, highlighting the effect of multicollinearity in this case. This

could have also effected the regression coefficient for mass, as it's positive which is counterintuitive to the theoretical physics behind free fall.

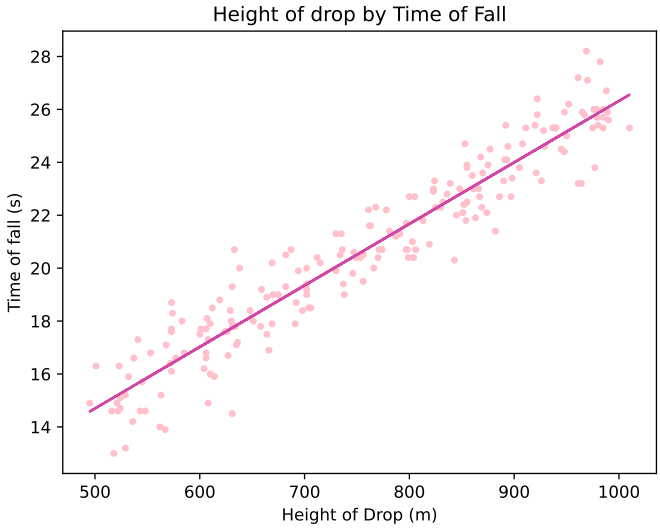

Figure 7: Linear Regression model on the iron subset of data with radius 0.05m. With each data point having the same radius, density and therefore mass the model is very accurate to the experimental data.

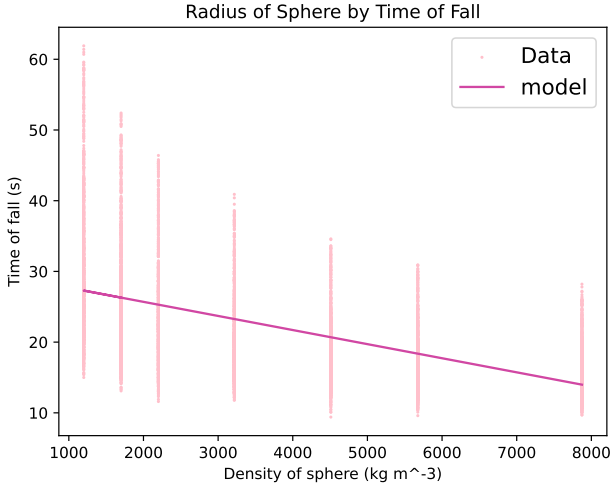

Figure 8: The Linear Regression model was fitted on all material data, with predicted time of fall calculated from the density dataset and plotted against it. Seven vertical lines can be seen from the seven materials densities.

4.2 Part 3 (b): Ridge and Lasso Regression Models

Ridge and Lasso regression models are designed to handle multicollinearity and overfitting, reducing the inflation or squishing of regression coefficients. Fitting these models to the data, the regression coefficients were extracted:

Coefficient	**Ridge**	**Lasso**
β_0 (Intercept)	22.220	22.220
β_1 (Density)	-6.362	-3.514
β_2 (Radius)	-7.484	-3.546
β_3 (Mass)	4.040	-0.000
β_4 (Temperature)	-0.352	-0.000
β_5 (Pressure)	0.275	0.000
β_6 (Height)	3.638	2.593
Mean Squared Error	10.5429	18.3170

Table 2: Ridge and Lasso Regression Coefficients and Mean Squared Error

In the Ridge regression the MSE is higher than the standard linear regression, this implies some bias is introduced to stabilize the coefficients. Most coefficient have barely been effected, showing the Ridge model only effects highly correlated coefficients. The intercept is kept, displaying the overall coefficients not being effected too much.

The Lasso regression has the largest MSE of the three models, this shows heavy bias and could mean underfitting of the data. The coefficients for temperature, mass and pressure been reduced to zero, implying the model considers these variables unnecessary, this goes directly against theory as mass has one of the largest weights in drop time and supports that the model has underfitted the data.

Overall the Ridge model is the best fit for the data and it accounts for the highly correlated variables and stabilizes their regression coefficients without underfitting them with too heavy bias.

4.3 Part 3 (c): Gradient Descent

In this section three gradient decent approaches are used to determine the regression coefficients. Here instead of solving equation (4) directly, coefficients are initially guessed and then adjusted in steps in the direction of the local gradient with respect to the β coefficients.

The local gradient function is calculated from the data matrix X, the number of data points and so number of rows in the data matrix N and the time array y:

$$\nabla_\beta \text{MSE}(\beta) = \frac{2}{N} X^T (X\beta - y) \tag{5}$$

The gradient decent means updating the beta values in the negative direction of $\nabla_\beta \text{MSE}(\beta)$. The 'length' of each step depends on the learning rate η. If the learning rate is too small rate the beta values never get close to the true value without a very high number of iterations or steps which takes longer to do and requires more computing power. However too large η means they'll step right over the true value leading to inaccuracy, so the size must be balanced and the most effective learning rate can be different for each method.

$$\beta_{\text{new}} = \beta_{\text{old}} - \eta \nabla_\beta \text{MSE}(\beta) \tag{6}$$

The resulting regression coefficients from the batch, stochastic and mini- batch gradient decent methods are found to be:

The batch gradient decent processes the entire data set in each iteration, this is computationally expensive since the free fall is such a large dataset. Processing it in it's entirety also requires a lot

Coefficient	Batch GD	Stochastic GD	Mini-Batch GD
β_0 (Intercept)	22.220	22.077	22.268
β_1 (Density)	22.220	22.077	22.268
β_2 (Radius)	-6.189	-6.299	-6.226
β_3 (Mass)	-7.207	-7.458	-7.274
β_4 (Temperature)	3.714	4.188	3.776
β_5 (Pressure)	-0.352	-0.330	-0.345
β_6 (Height)	0.277	0.132	0.260
Mean Squared Error	10.576	10.632	10.571

Table 3: Gradient Descent Methods: Regression Coefficients and Mean Squared Error

of processing power and storage space. However this leads to accurate regression coefficients, shown by having the second lowest MSE, very close to the lowest. The height coefficient is largest in this gradient method, this shows height has a stronger influence on the resulting time value than from the other methods.

The stochastic gradient decent method processes updates the beta coefficients at each data point, this means a more scattered path. This means a larger number of iterations was required for the coefficients to converge. Despite this, the stochastic method still had the largest MSE of the three methods. It also has the most negative coefficients for radius and mass, these must have a stronger negative relation to the final time value.

The final method, mini-batch, is a halfway house between the other two. Instead of processing the whole dataset or just one point for each iteration, a small randomly selected subset of the dataset is processed. This gives a smoother path of coefficient updates but also doesn't require the computing power and memory he batch method needs. In this case a batch size of 10 was used, achieving the lowest MSE of the three methods. This method has coefficients in between the values of the two other methods, this is unsurprising due to it's method of iteration.

Overall, the mini-batch approach seems the most efficient for this experiment due to its balance between the accuracy of the batch method with its high demand for resources and the shaky iterations of the stochastic method.

5 Part 4: Model verification

5.1 Part 4: (a)

In this section the linear regression model is tested by training it with a randomly selected 90% of the data, then the time of fall is predicted from the remaining 10% and these predictions are tested against the actual fall times of this 10%. The accuracy of the model to reality is analyzed by examining the residuals, this is the deviation of the model predicted time of drop from the actual time of fall.

The model is found to be especially accurate when fitted to height with time, this could be due to relation between height and time of fall can be approximately linear for small height values.

5.2 Part 4: Model verification (b)

Next the linear regression model is tested for predicting the time of fall for heights outside of the range given in the dataset. As there's no data for comparison, theoretical models of free fall will be used to

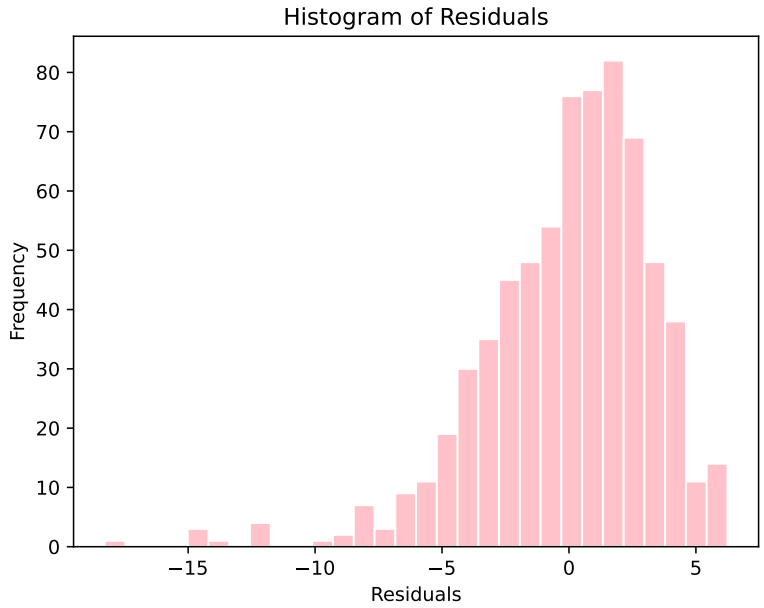

Figure 9: Histogram of residuals. It's centered around zero, showing the model isn't biased and is well fitted to the data. The trail off in the negative residual direction could mean some predictions are much higher than the actual time value, and are being overestimated.

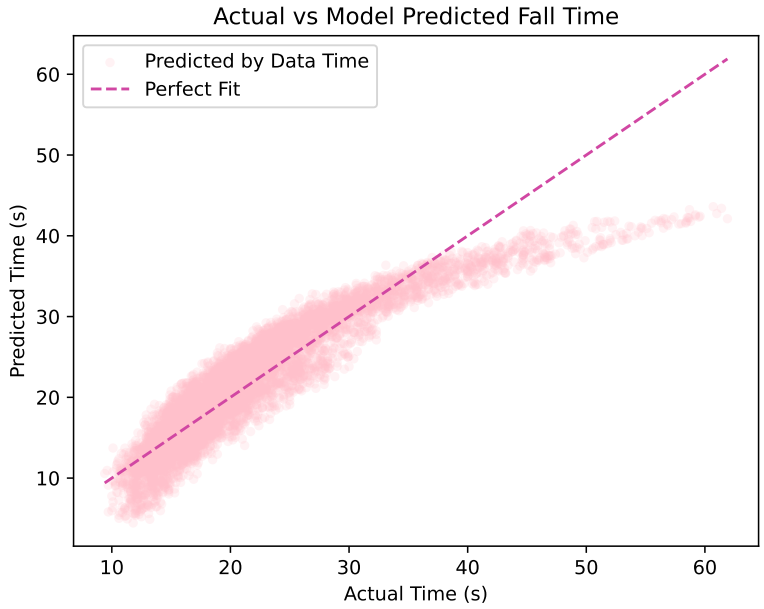

Figure 10: Scatter plot of model predicted time by theoretical time of fall, if the model were perfect each point would lie on the diagonal. It can be seen here that the model time becomes increasingly underestimated for longer drops, straying from the 'perfect fit' line.

calculate the true fall time in these regions. Theoretical time of free fall is calculated by:

$$t = \sqrt{\frac{m}{kg}} \cosh^{-1}\left(\exp\left(\frac{hk}{m}\right)\right) \tag{7}$$

Where k is the drag coefficient calculated by:

$$k = \frac{C_d \rho_0 A}{2} \tag{8}$$

In which C_d is the drag coefficient, approximately 0.47 for a sphere, A is the cross sectional area of a sphere and ρ_0 is air density calculated by:

$$\rho_0 = \frac{pM}{RT} \tag{9}$$

Here M is molar mass of dry air, R is the molar gas constant and P and T are pressure and temperature.

Now the theoretically calculated drop times from equations (7) to (9) is taken as the actual time for the regions outside of the dataset.

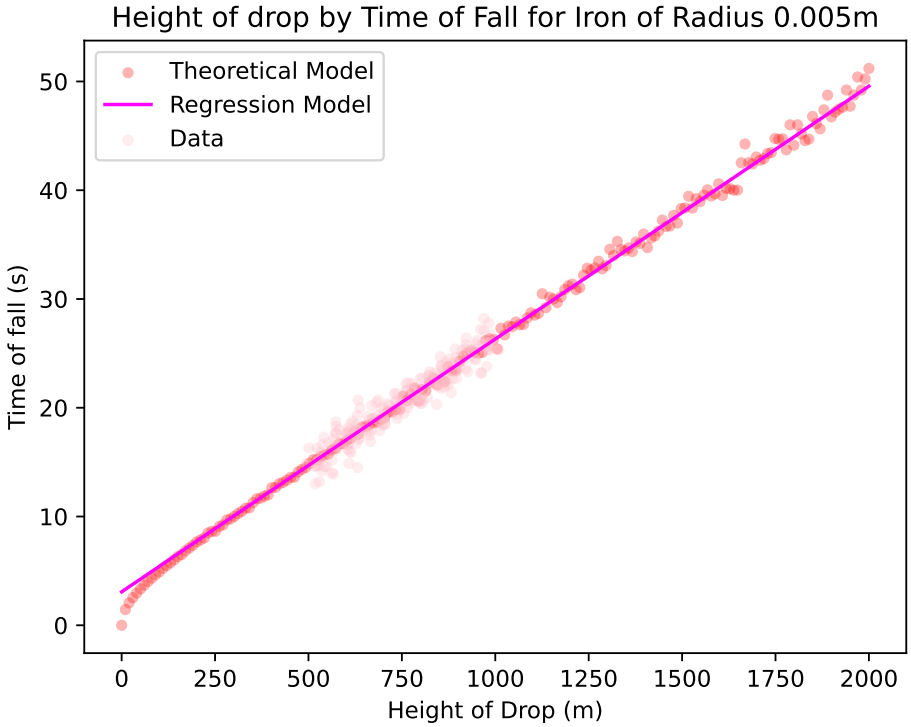

Figure 11: Filtering the data to just iron spheres of radius 0.05m as before to see a linear relation, the regression model is very accurate beyond the bounds of data, with the theoretical height and time data points surrounding the model line. In light pink the experimental data is shown, the height ranges from 500m to 1000m and is also surrounding the linear model. The model is trained with this data and so extends from it.

Overall, the linear regression model shows an accurate fit to the experimental data with little bias, displayed by the residuals centred around zero. However some points had heavily overestimated time predictions, causing a slight skew in residuals. This overestimation could be due to the neglection

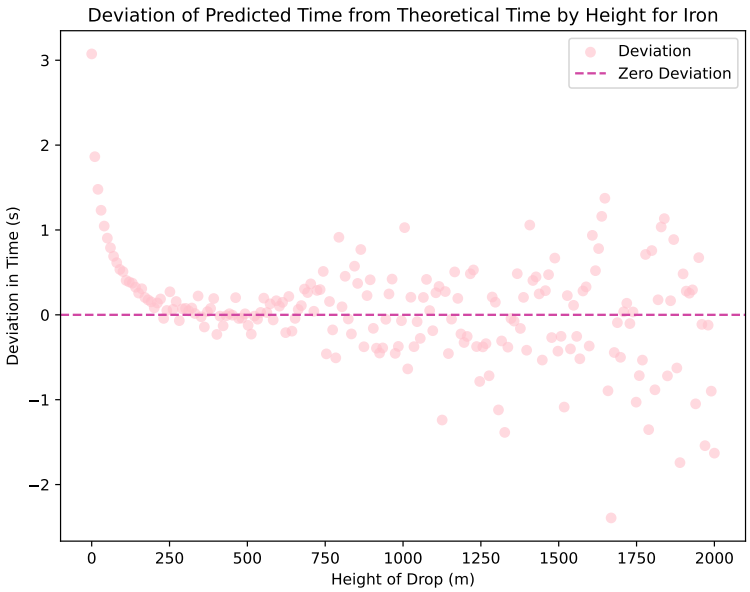

Figure 12: With only iron spheres of radius 0.05m, the deviation between the regression model predicted time of fall and the theoretically calculated time of fall are plotted against height. It's clear that as height increases the model becomes less accurate to the theoretical time, seen by the deviation in the data points becoming more spread out.

of other variables, especially non-linear ones such as air resistance, drag or consideration to terminal velocity.

While the model is accurate within certain heights, comparisons with theoretical time values for much larger or smaller height values show deviation from the true value with height. To improve this the non-linear variables also at play could be accounted for, as with the skewed residuals.