THE UNIVERSITY OF WINNIPEG | Professional, Applied and Continuing Education

INTRODUCTION TO MACHINE LEARNING

DIT 45100

THE UNIVERSITY OF WINNIPEG | Professional, Applied and Continuing Education

Module 2

Linear Regression

Linear Regression

- What is regression?
  - Continuous target feature (aka dependent variable or label)
  - Linear regression
  - Non-linear or polynomial regression
- Error-Based Learning

Big Idea

- Error-Based Learning
  - Ever tried learning a new game?
  - Or a new skill?

Big Idea

- A **parameterised** prediction model is initialized with a set of random parameters and an error function is used to judge how well this initial model performs when making predictions for instances in a training dataset.
- Based on the value of the error function the parameters are iteratively adjusted to create a more and more accurate model.

Linear Regression

Types of Linear Regression

- Simple Linear Regression
- Multiple Linear Regression

Simple Linear Regression

Table: The **office rentals dataset**: a dataset that includes office rental prices and a number of descriptive features for 10 Dublin city-centre offices.

ID	Size	Floor	Broadband Rate	Energy Rating	Rental Price
1	500	4	8	C	320
2	550	7	50	A	380
3	620	9	7	A	400
4	630	5	24	B	390
5	665	8	100	C	385
6	700	4	8	B	410
7	770	10	7	B	480
8	880	12	50	A	600
9	920	14	8	C	570
10	1,000	9	24	B	620

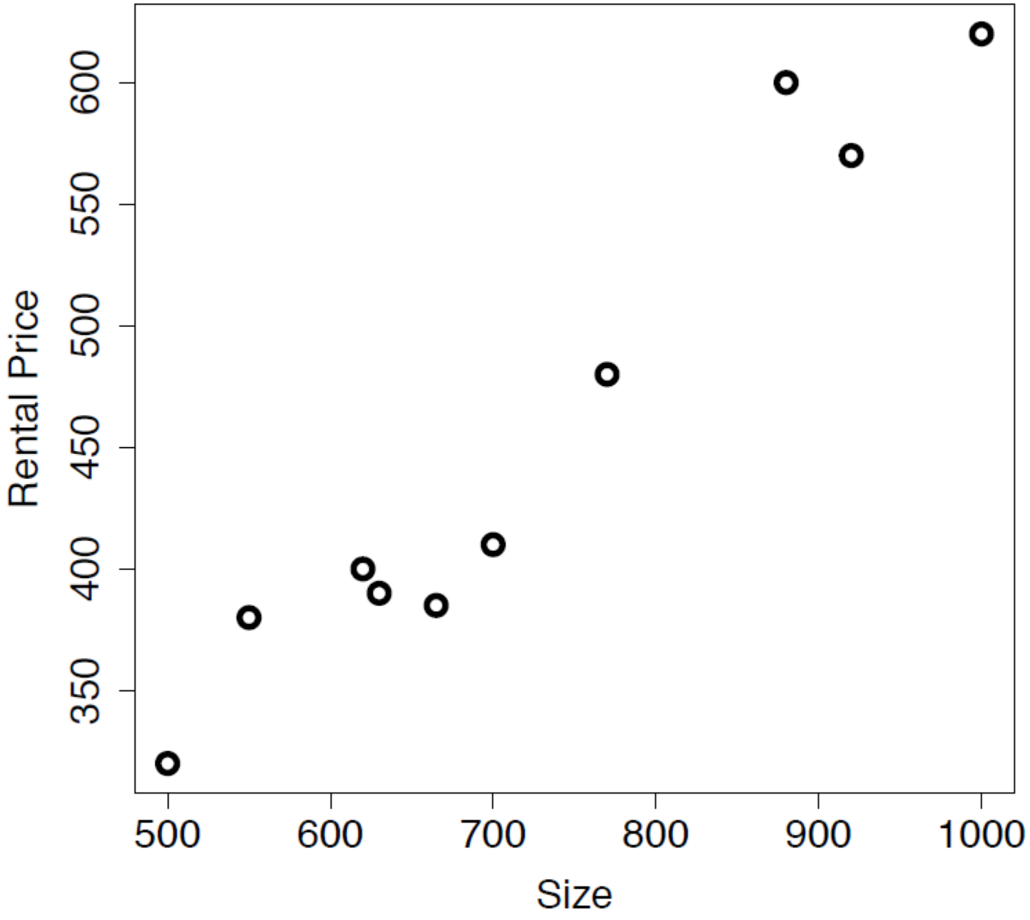
Simple Linear Regression

Table: The **office rentals dataset**: a dataset that includes office rental prices and a number of descriptive features for 10 Dublin city-centre offices.

ID	SIZE	RENTAL PRICE
1	500	320
2	550	380
3	620	400
4	630	390
5	665	385
6	700	410
7	770	480
8	880	600
9	920	570
10	1,000	620

Simple Linear Regression

Figure: A scatter plot of the SIZE and RENTAL PRICE features from the office rentals dataset.

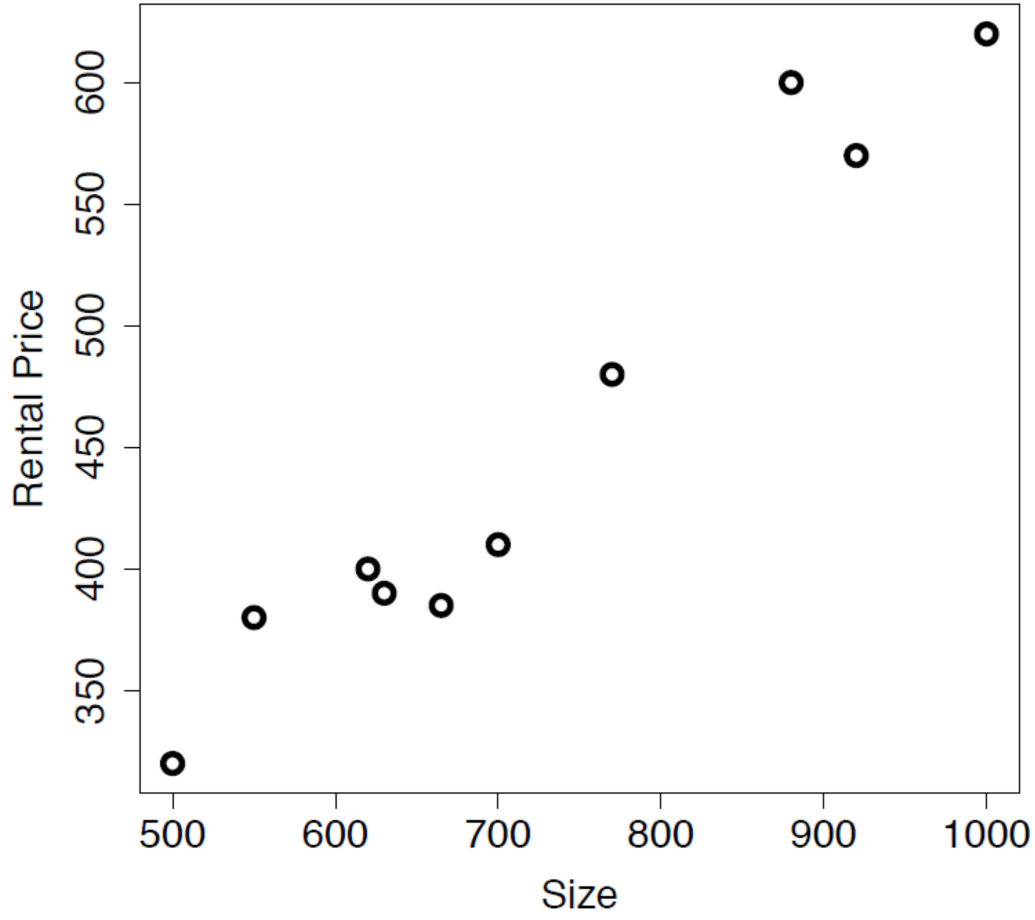


Simple Linear Regression

- From the scatter plot it appears that there is a linear relationship between the SIZE and RENTAL PRICE.

- The equation of a line can be written as:

$$y = mx + b \qquad (1)$$

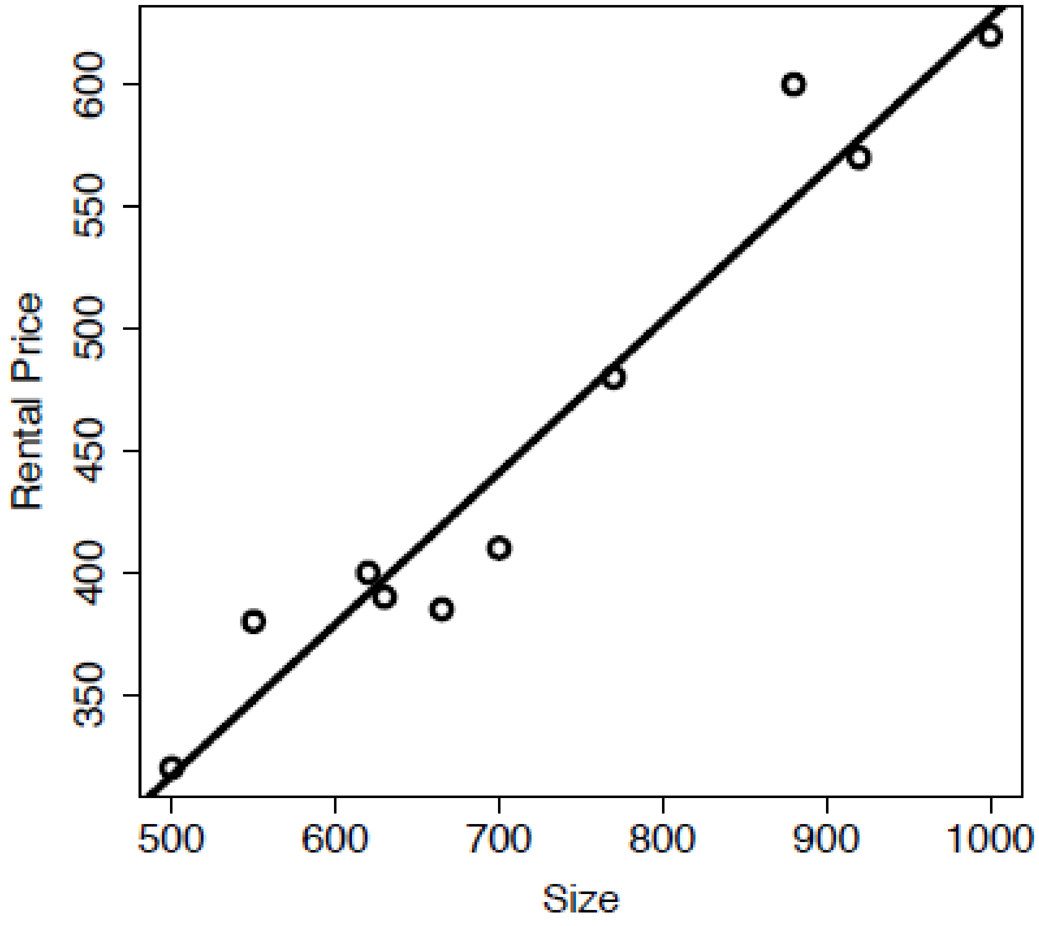


Simple Linear Regression

- This shows the same scatter plot as shown in the previous slide with a simple linear model added to capture the relationship between office sizes and office rental prices.

- This model is:

RENTAL PRICE = 6.47 + 0.62 x SIZE



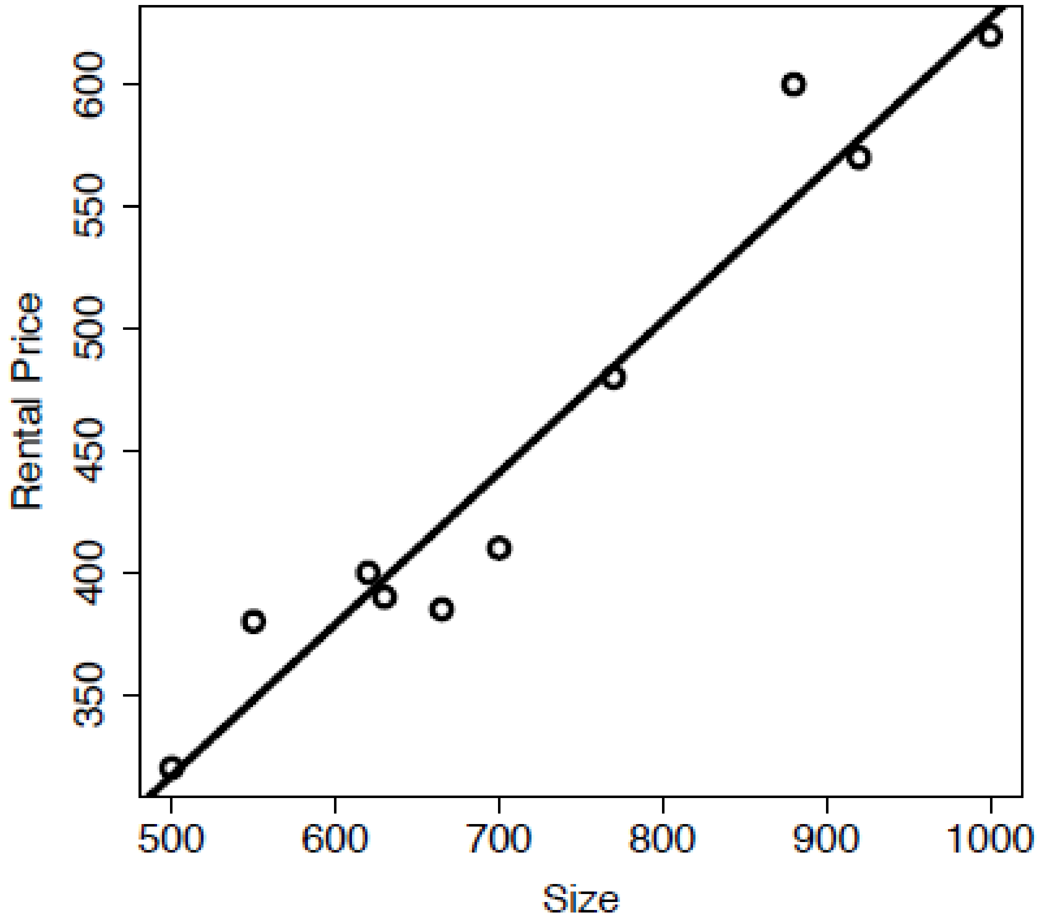
Simple Linear Regression

- This shows the same scatter plot as shown in the previous slide with a simple linear model added to capture the relationship between office sizes and office rental prices.

- This model is:

RENTAL PRICE = 6.47 + 0.62 x SIZE

- Using this model determine the expected rental price of the 730 square foot office:



Simple Linear Regression

- This shows the same scatter plot as shown in the previous slide with a simple linear model added to capture the relationship between office sizes and office rental prices.

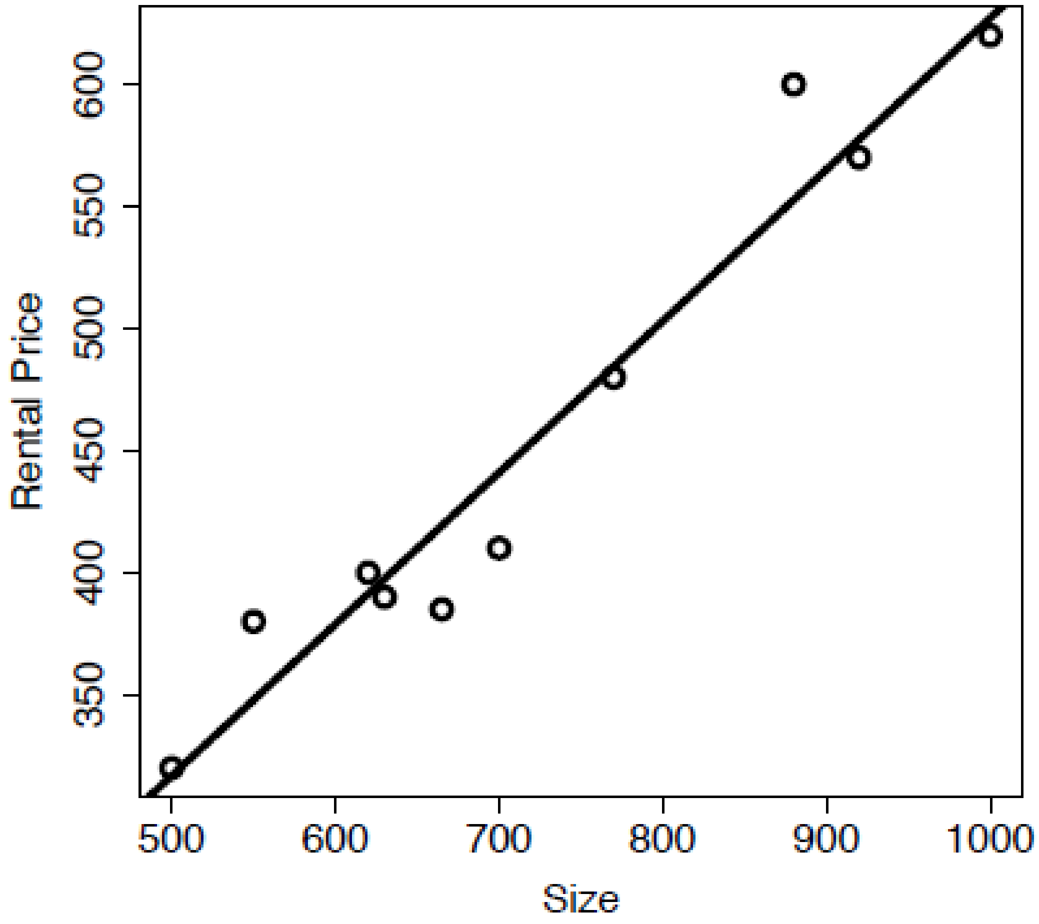
- This model is:

RENTAL PRICE = 6.47 + 0.62 x SIZE

- Using this model determine the expected rental price of the 730 square foot office:

RENTAL PRICE = 6.47 + 0.62 x 730

= 459.07



Simple Linear Regression

How do we find the best fit model?

- Let’s rewrite Equation (1) in matrix form:

$$M_{\mathbf{w}}(d) = \mathbf{w}[0] + \mathbf{w}[1] \times \mathbf{d}[1] \qquad (2)$$

Simple Linear Regression

How do we find the best fit model?

- Let's rewrite Equation (1) in matrix form:

$$M_{\mathbf{w}}(d) = \mathbf{w}[0] + \mathbf{w}[1] \times \mathbf{d}[1] \qquad (2)$$

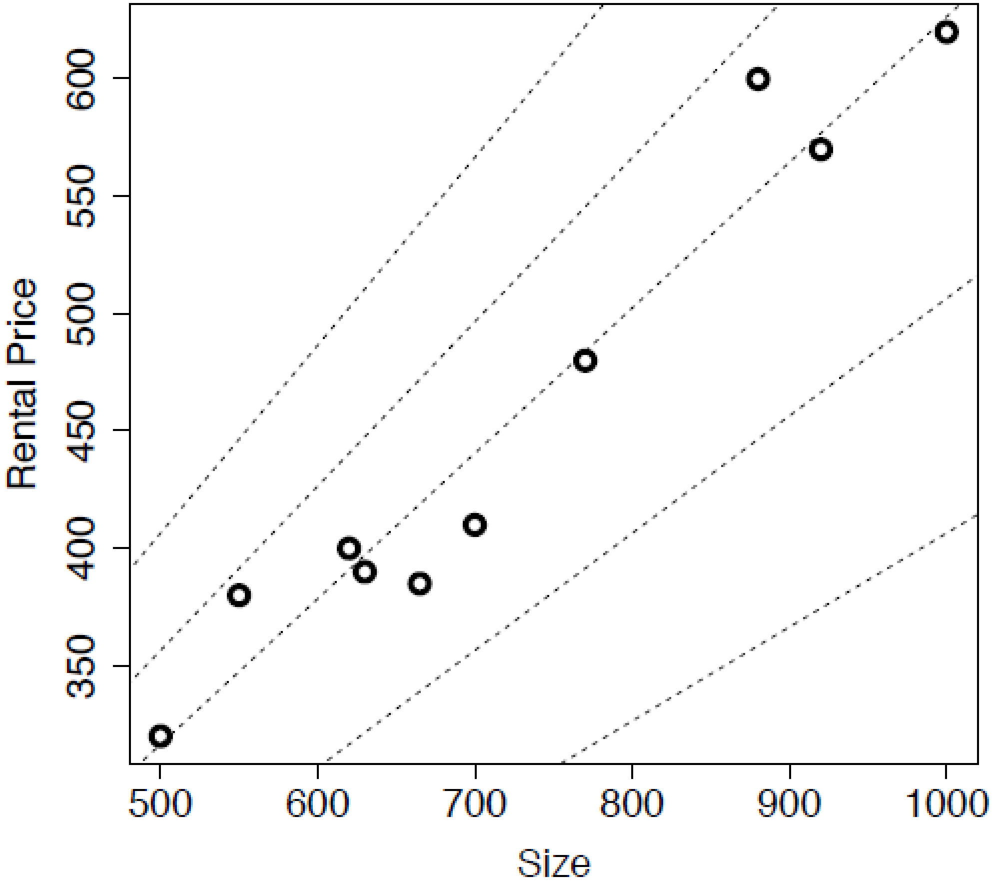


Figure: A scatter plot of the SIZE and RENTAL PRICE features from the office rentals dataset. A collection of possible simple linear regression models capturing the relationship between these two features are also shown.

- For all models **w**[0] is set to 6.47.
- From top to bottom the models use 0.8, 0.7, 0.62, 0.5 and 0.4 respectively for **w**[1].

Simple Linear Regression

Measuring Error

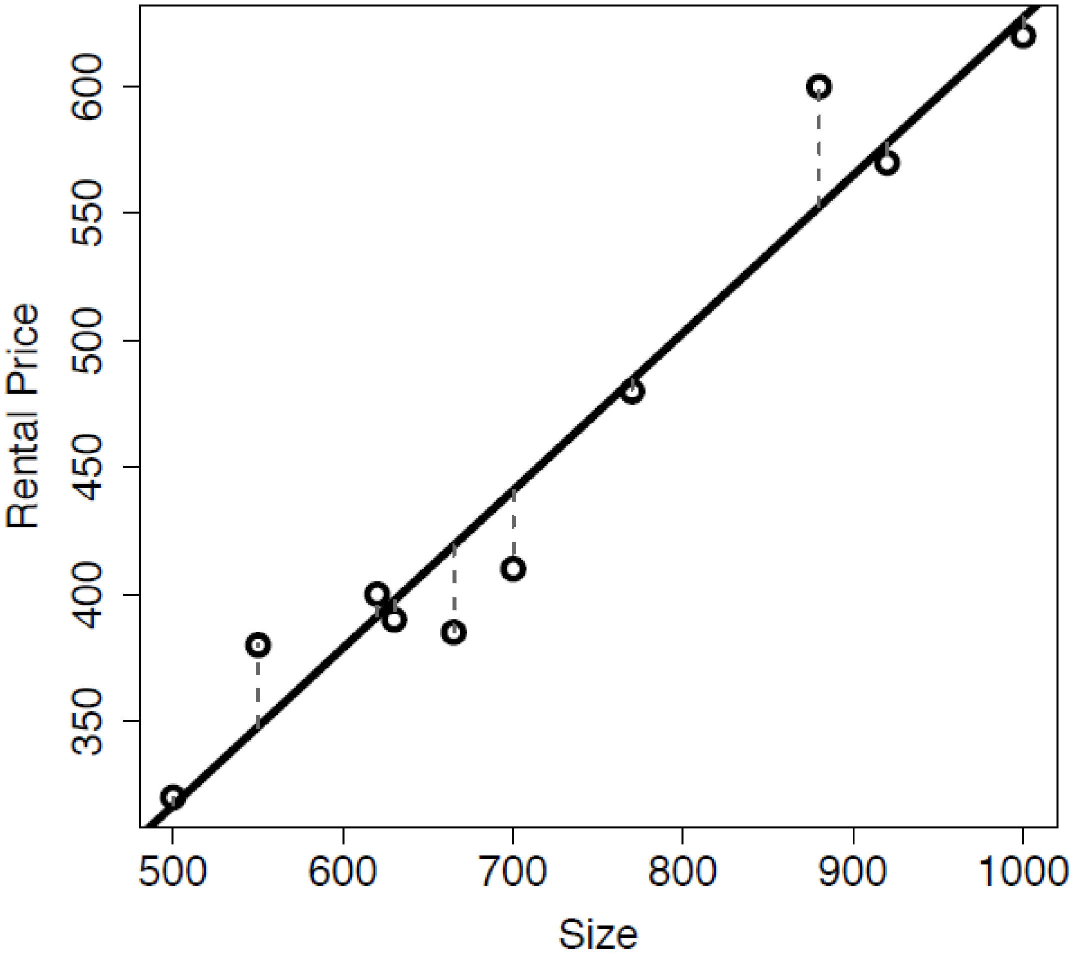


Figure:

A scatter plot of the SIZE and RENTAL PRICE features from the office rentals dataset showing a candidate prediction model (with **w**[0] = 6.47 and **w**[1] = 0.62) and the resulting errors.

Simple Linear Regression

Measuring Error

<u>L2 Loss Function</u>

$$
\begin{aligned}
L_2(\mathbb{M}_{\mathbf{w}}, \mathcal{D}) &= \frac{1}{2}\sum_{i=1}^{n}\left(t_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i[1])\right)^2 && (3) \\
&= \frac{1}{2}\sum_{i=1}^{n}\left(t_i - (\mathbf{w}[0] + \mathbf{w}[1] \times \mathbf{d}_i[1])\right)^2 && (4)
\end{aligned}
$$

Simple Linear Regression

Measuring Error

Table: Calculating the sum of squared errors for the candidate model (with **w**[0] = 6.47 and **w**[1] = 0.62) making predictions for the office rentals dataset.

ID	Rental Price	Model Prediction	Error	Squared Error
1	320	316.79	3.21	10.32
2	380	347.82	32.18	1,035.62
3	400	391.26	8.74	76.32
4	390	397.47	-7.47	55.80
5	385	419.19	-34.19	1,169.13
6	410	440.91	-30.91	955.73
7	480	484.36	-4.36	19.01
8	600	552.63	47.37	2,243.90
9	570	577.46	-7.46	55.59
10	620	627.11	-7.11	50.51
			Sum	**5,671.64**
			Sum of squared errors (Sum/2)	**2,835.82**

Simple Linear Regression

Error Surfaces

For every possible combination of weights, **w**[0] and **w**[1], there is a corresponding sum of squared errors value that can be joined together to make a surface.

Simple Linear Regression

Error Surfaces

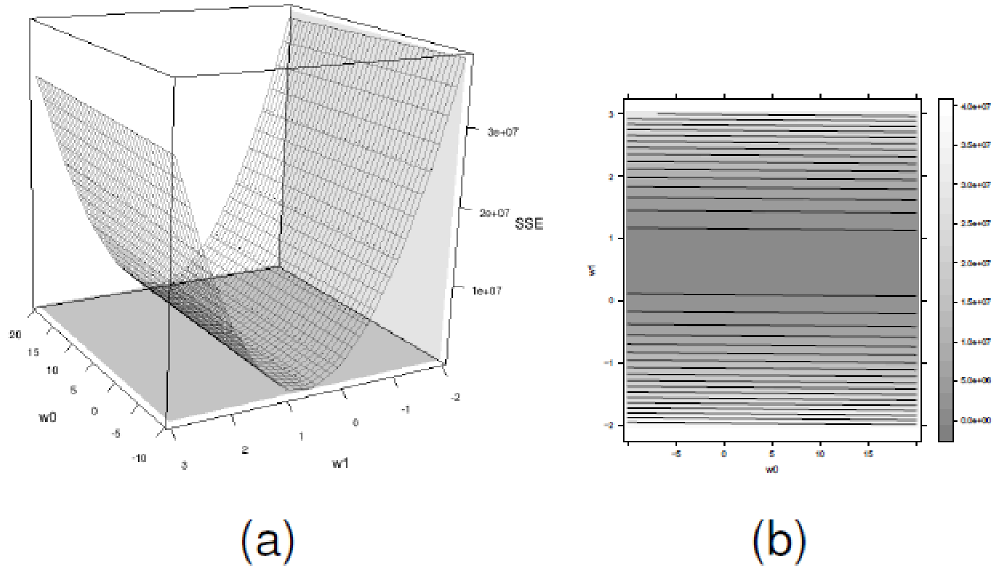


Figure: (a) A 3D surface plot and (b) a contour plot of the error surface generated by plotting the sum of squared errors value for the office rentals training set for each possible combination of values for **w**[0] (from the range $[-10, 20]$) and **w**[1] (from the range $[-2, 3]$).

Simple Linear Regression

Error Surfaces

- Using Equation (4) we can formally define this point on the error surface as the point at which:

$$\frac{\partial}{\partial \mathbf{w}[0]} \frac{1}{2} \sum_{i=1}^{n} (t_i - (\mathbf{w}[0] + \mathbf{w}[1] \times \mathbf{d}_i[1]))^2 = 0 \qquad (5)$$

and

$$\frac{\partial}{\partial \mathbf{w}[1]} \frac{1}{2} \sum_{i=1}^{n} (t_i - (\mathbf{w}[0] + \mathbf{w}[1] \times \mathbf{d}_i[1]))^2 = 0 \qquad (6)$$

- There are a number of different ways to find this point.
- We will describe a **guided search** approach known as the **gradient descent** algorithm.

THE UNIVERSITY OF WINNIPEG | Professional, Applied and Continuing Education

Standard Approach: Multivariate Linear Regression with Gradient Descent

Multivariate Linear Regression

Table: The **office rentals dataset**: a dataset that includes office rental prices and a number of descriptive features for 10 Dublin city-centre offices.

ID	Size	Floor	Broadband Rate	Energy Rating	Rental Price
1	500	4	8	C	320
2	550	7	50	A	380
3	620	9	7	A	400
4	630	5	24	B	390
5	665	8	100	C	385
6	700	4	8	B	410
7	770	10	7	B	480
8	880	12	50	A	600
9	920	14	8	C	570
10	1,000	9	24	B	620

Multivariate Linear Regression

- We can define a multivariate linear regression model as:

$$\mathbb{M}_{\mathbf{w}}(\mathbf{d}) = \mathbf{w}[0] + \mathbf{w}[1] \times \mathbf{d}[1] + \cdots + \mathbf{w}[m] \times \mathbf{d}[m] \quad (7)$$

$$= \mathbf{w}[0] + \sum_{j=1}^{m} \mathbf{w}[j] \times \mathbf{d}[j] \quad (8)$$

Multivariate Linear Regression

- We can make Equation (8) look a little neater by inventing a dummy descriptive feature, **d**[0], that is always equal to 1:

$$
\begin{aligned}
\mathbb{M}_{\mathbf{w}}(\mathbf{d}) &= \mathbf{w}[0] \times \mathbf{d}[0] + \mathbf{w}[1] \times \mathbf{d}[1] + \ldots + \mathbf{w}[m] \times \mathbf{d}[m] && (9)\\
&= \sum_{j=0}^{m} \mathbf{w}[j] \times \mathbf{d}[j] && (10)\\
&= \mathbf{w} \cdot \mathbf{d} && (11)
\end{aligned}
$$

Multivariate Linear Regression

- The sum of squared errors loss function, L2, definition that we gave earlier in Equation (4) changes only very slightly to reflect the new regression equation:

$$
\begin{aligned}
L_2(\mathbb{M}_{\mathbf{w}}, \mathcal{D}) &= \frac{1}{2}\sum_{i=1}^{n}(t_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i))^2 && (12)\\
&= \frac{1}{2}\sum_{i=1}^{n}(t_i - (\mathbf{w} \cdot \mathbf{d}_i))^2 && (13)
\end{aligned}
$$

Multivariate Linear Regression

- This multivariate model allows us to include all but one of the descriptive features in the dataset in a regression model to predict office rental prices.

- The resulting multivariate regression model equation is:

RENTAL PRICE = **w**[0] + **w**[1] x SIZE + **w**[2] x FLOOR
+ **w**[3] x BROADBAND RATE

Multivariate Linear Regression

- We will see in the next section how the best-fit set of weights for this equation are found, but for now we will set:
  - w[0] = - 0.1513
  - w[1] = 0.6270
  - w[2] = - 0.1781
  - w[3] = 0.0714
- This means that the model is rewritten as:

RENTAL PRICE = - 0.1513 + 0.6270 x SIZE - 0.1781 x FLOOR

+ 0.0714 x BROADBAND RATE

Multivariate Linear Regression

- Using this model :

$$\text{RENTAL PRICE} = -\,0.1513 + 0.6270 \times \text{SIZE} - 0.1781 \times \text{FLOOR} + 0.0714 \times \text{BROADBAND RATE}$$

- We can, for example, predict the expected rental price of a 690 square foot office on the 11th floor of a building with a broadband rate of 50 Mb per second as:

$$\text{RENTAL PRICE} = -\,0.1513 + 0.6270 \times 690 - 0.1781 \times 11 + 0.0714 \times 50$$
$$= 434.0896$$

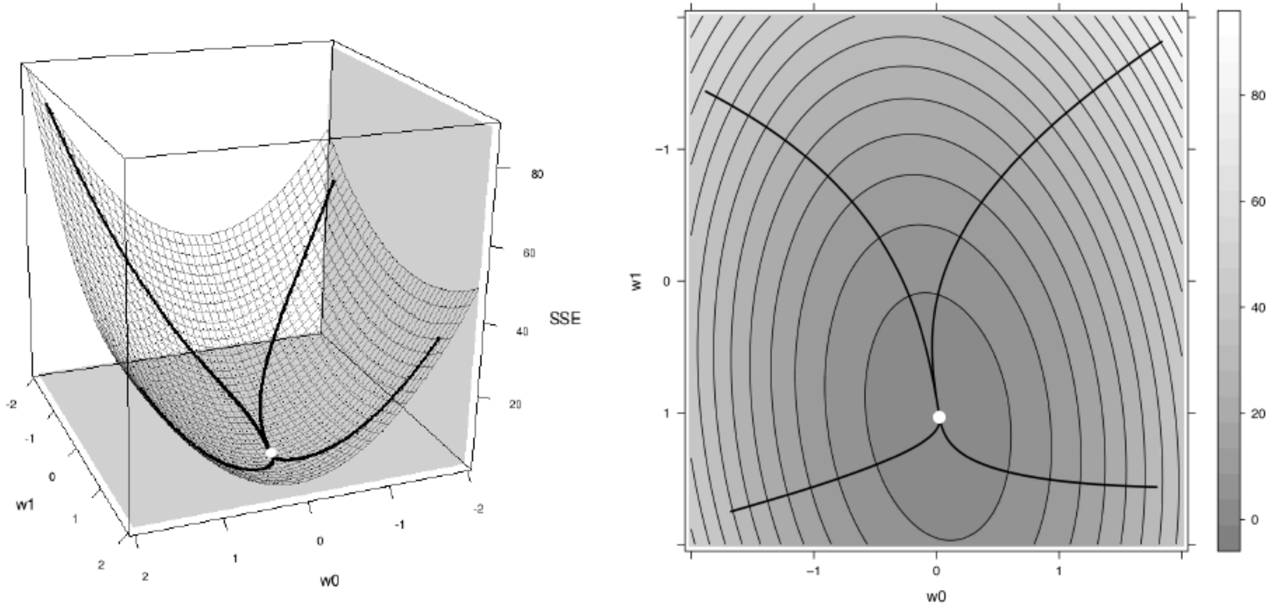
Multivariate Linear Regression

Gradient Descent

- The gradient descent algorithm is used for training multivariate linear regression models.

Multivariate Linear Regression

Gradient Descent



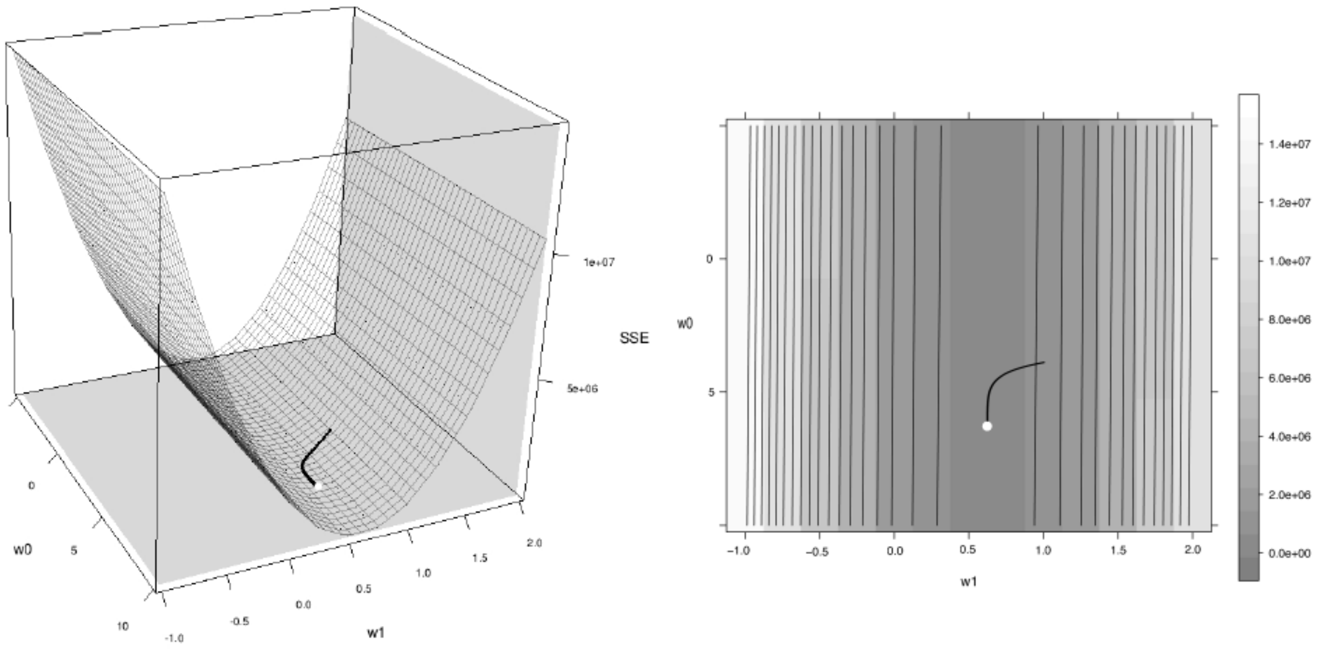
(a) (b)

Figure: (a) A 3D surface plot and (b) a contour plot of the same error surface. The lines indicate the path that the gradient decent algorithm would take across this error surface from different starting positions to the global minimum - marked as the white dot in the centre.

Multivariate Linear Regression

The journey across the error surface that is taken by the gradient descent algorithm when training the simple version of the office rentals example - involving just SIZE and RENTAL PRICE.

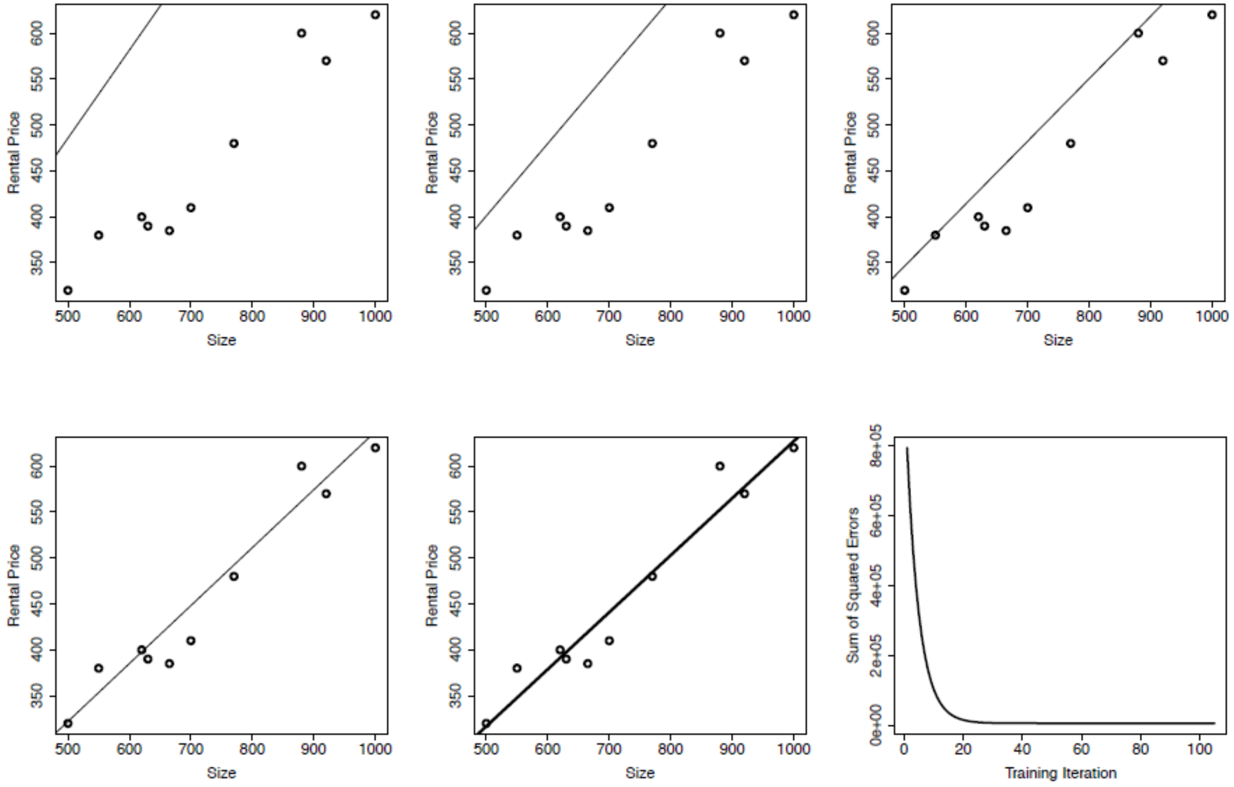
Figure: (a) A 3D surface plot and (b) a contour plot of the error surface for the office rentals dataset showing the path that the gradient descent algorithm takes towards the best fit model.



(a) (b)

Multivariate Linear Regression

- **Figure:** A selection of the simple linear regression models developed during the gradient descent process for the office rentals dataset.
  - The final panel shows the sum of squared error values generated during the gradient descent process.



Multivariate Linear Regression

Gradient Descent

Require: set of training instances $\mathcal{D}$

Require: a learning rate α that controls how quickly the algorithm converges

Require: a function, **errorDelta**, that determines the direction in which to adjust a given weight, $\mathbf{w}[j]$, so as to move down the slope of an error surface determined by the dataset, $\mathcal{D}$

Require: a convergence criterion that indicates that the algorithm has completed

$\mathbf{w} \leftarrow$ random starting point in the weight space

repeat

  for each $\mathbf{w}[j]$ in $\mathbf{w}$ **do**

    $\mathbf{w}[j] \leftarrow \mathbf{w}[j] + \alpha \times \mathbf{errorDelta}(\mathcal{D}, \mathbf{w}[j])$

  end for

until convergence occurs

Multivariate Linear Regression

Gradient Descent

The most important part to the gradient descent algorithm is Line Rule 4 on which the weights are updated.

$$\mathbf{w}[j] \leftarrow \mathbf{w}[j] + \alpha \times \mathbf{errorDelta}(\mathcal{D}, \mathbf{w}[j])$$

Each weight is considered independently and for each one a small adjustment is made by adding a small **delta** value to the current weight, $\mathbf{w}[j]$.

This adjustment should ensure that the change in the weight leads to a move *downwards* on the error surface.

Multivariate Linear Regression

Gradient Descent

Imagine for a moment that our training dataset, $\mathcal{D}$ contains just one training example: $(\mathbf{d}, t)$

The gradient of the error surface is given as the partial derivative of L_2 with respect to each weight, $\mathbf{w}[j]$:

$$
\begin{aligned}
\frac{\partial}{\partial \mathbf{w}[j]} L_2(\mathbb{M}_{\mathbf{w}}, \mathcal{D}) &= \frac{\partial}{\partial \mathbf{w}[j]} \left(\frac{1}{2} (t - \mathbb{M}_{\mathbf{w}}(\mathbf{d}))^2 \right) && (14) \\
&= (t - \mathbb{M}_{\mathbf{w}}(\mathbf{d})) \times \frac{\partial}{\partial \mathbf{w}[j]} (t - \mathbb{M}_{\mathbf{w}}(\mathbf{d})) && (15) \\
&= (t - \mathbb{M}_{\mathbf{w}}(\mathbf{d})) \times \frac{\partial}{\partial \mathbf{w}[j]} (t - (\mathbf{w} \cdot \mathbf{d})) && (16) \\
&= (t - \mathbb{M}_{\mathbf{w}}(\mathbf{d})) \times -\mathbf{d}[j] && (17)
\end{aligned}
$$

Multivariate Linear Regression

Gradient Descent

Adjusting the calculation to take into account multiple training instances:

$$\frac{\partial}{\partial \mathbf{w}[j]} L_2(\mathbb{M}_{\mathbf{w}}, \mathcal{D}) = \sum_{i=1}^{n} ((t_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i)) \times \mathbf{d}_i[j])$$

We use this equation to define the **errorDelta** in our gradient descent algorithm.

$$\mathbf{w}[j] \leftarrow \mathbf{w}[j] + \alpha \underbrace{\sum_{i=1}^{n} ((t_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i)) \times d_i[j])}_{errorDelta(\mathcal{D}, \mathbf{w}[j])}$$

Multivariate Linear Regression

Choosing the Learning Rate

- The learning rate, α, determines the size of the adjustment made to each weight at each step in the process.
- Unfortunately, choosing learning rates is not a well defined science. Most practitioners use rules of thumb and trial and error.
- A typical range for learning rates is [>0.000001, <1], commonly set to 0.1 or 0.01
- Based on empirical evidence, choosing random initial weights uniformly from the range [-0.2, 0.2] tends to work well.

Multivariate Linear Regression

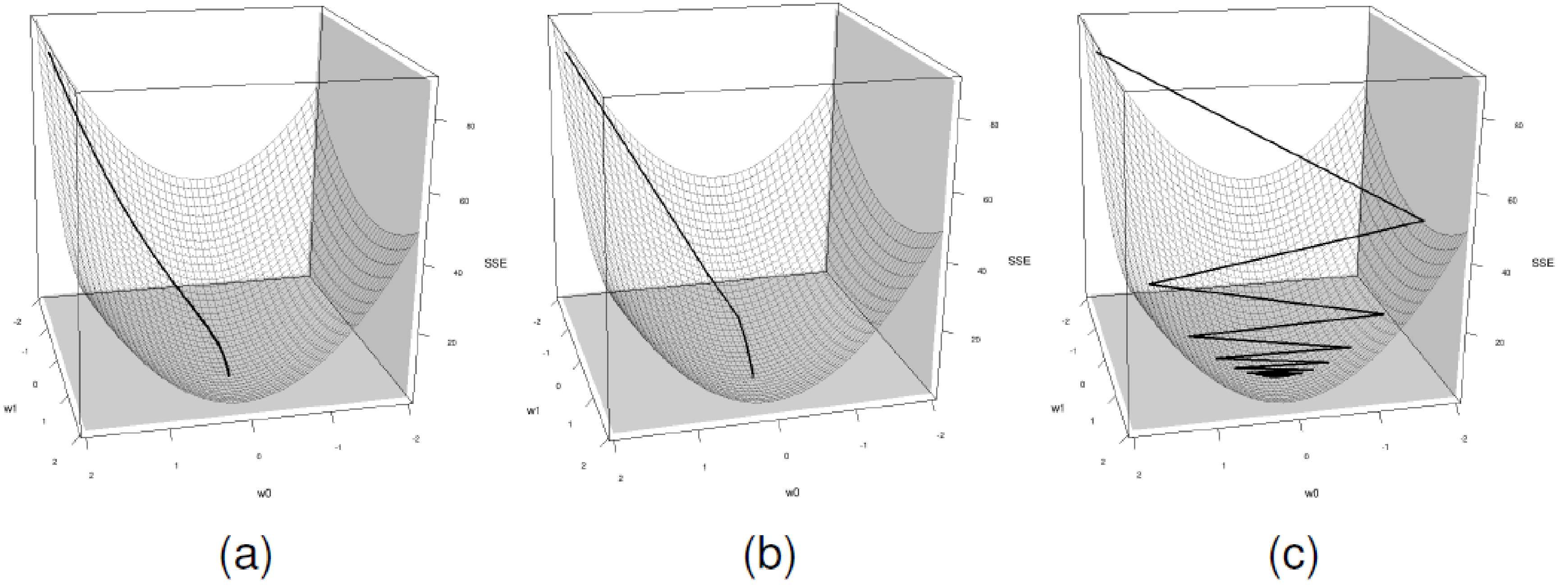


Figure: Plots of the journeys made across the error surface for the simple office rentals prediction problem for different learning rates: (a) a very small learning rate (0.002), (b) a medium learning rate (0.08) and (c) a very large learning rate (0.18).