Sentiment Analysis in Social Networks

SENTIMENT ANALYSIS

Discovering people opinions, emotions and feelings about a product or service

a0707o7
Huntelaar
dearevan
masahiko2009
david75610
TaiwanIsDog
ZenUp
[新聞] 洪仲丘家屬夠了 陳揮文：我們民
[新聞] 洪仲丘家屬夠了 陳揮文：我們民
[新聞] 不信故意操死洪仲丘 陳揮文：洪
[新聞] 洪仲丘案延燒 柯文哲：不想再
[新聞] 洪仲丘案延燒 柯文哲：不想再
[問卦] 洪仲丘案延燒 柯文哲：不想再
[問卦] 如果洪仲丘案發生在美軍有可能水落石
[問卦] 如果洪仲丘案發生在美軍有可能水落石
如果洪仲丘案發生在美軍有可能水落石
軍方否認製作洪仲丘案文宣資料
自稱洪仲丘「託夢」夜敲洪家門要上香
自稱洪仲丘「託夢」夜敲洪家門要上香
軍檢：洪仲丘案仍可能是意外

Hillary Clinton

Democratic Party

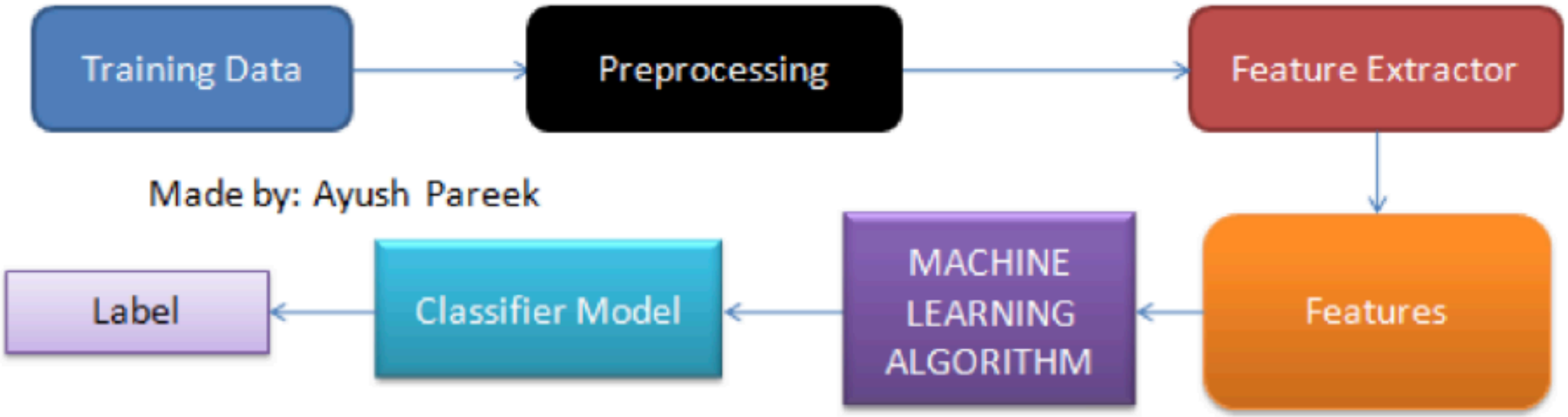
Mentions	
Total	785
No. of positive	151
No. of negative	155
No. of neutral	479

Donald Trump

Republican Party

Mentions	
1,412	Total
255	No. of positive
402	No. of negative
755	No. of neutral

Basic Framework for Sentiment Analysis



Big Idea

- Simple Linear Regression
- Measuring Error
- Error Surfaces
- Gradient Descent
- A Worked Example

Logistic Regression

- Logistic Function

Big Idea

Parameterized Prediction model

A **parameterized** 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.

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

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

We can define a multivariate linear regression model as:

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

Example

$$\begin{aligned}\text{Rental Price} = \mathbf{w}[0] &+ \mathbf{w}[1] \times \text{Size} + \mathbf{w}[2] \times \text{Floor} \\ &+ \mathbf{w}[3] \times \text{Broadband Rate}\end{aligned}$$

For example we set:

$\mathbf{w}[0] = -0.1513$, $\mathbf{w}[1] = 0.6270$, $\mathbf{w}[2] = -0.1781$, $\mathbf{w}[3] = 0.0714$.

$$\begin{aligned}\text{RENTAL PRICE} = \mathbf{w}[0] &+ \mathbf{w}[1] \times \text{SIZE} + \mathbf{w}[2] \times \text{FLOOR} \\ &+ \mathbf{w}[3] \times \text{BROADBAND RATE}\end{aligned}$$

This means that the model is rewritten as:

$$\begin{aligned}\text{RENTAL PRICE} = -0.1513 &+ 0.6270 \times \text{SIZE} \\ &- 0.1781 \times \text{FLOOR} \\ &+ 0.0714 \times \text{BROADBAND RATE}\end{aligned}$$

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:

$$\begin{aligned}\text{RENTAL PRICE} &= -0.1513 + 0.6270 \times 690 \\ &\quad -0.1781 \times 11 + 0.0714 \times 50 \\ &= 434.0896\end{aligned}$$

Cost Function: How Good a Set of Parameters is?

We can make the equation abovelook a little neater by inventing a dummy descriptive feature, **d**[0], that is always equal to 1:

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

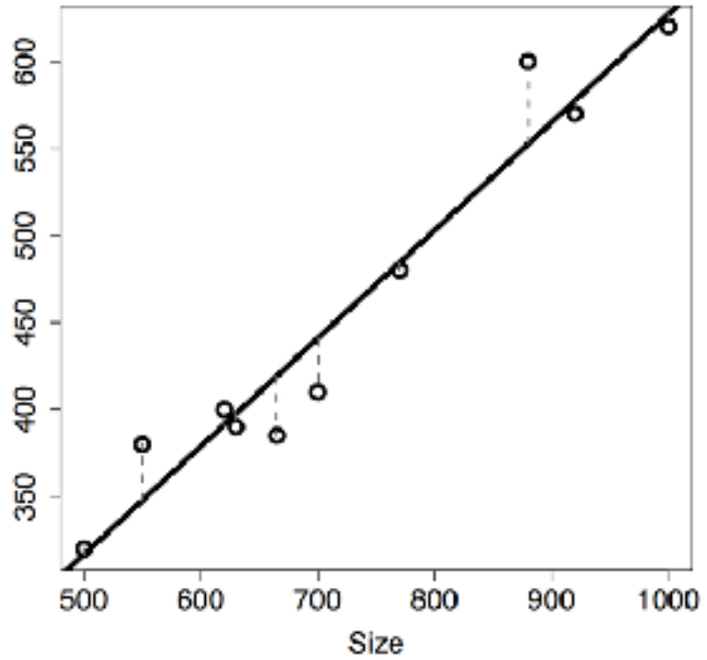
The sum of squared errors loss function

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

Example:The Performance of A Set of Parameters

$$L_2(\mathbb{M}_{\mathbf{w}}, \mathcal{D}) = \sum_{i=1}^{n} (t_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i[1]))^2$$

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



ID	Rental Price	Model Prediction	Error 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**
				2,835.82

→越小越好

with **w**[0] = 6.47 and **w**[1] = 0.62

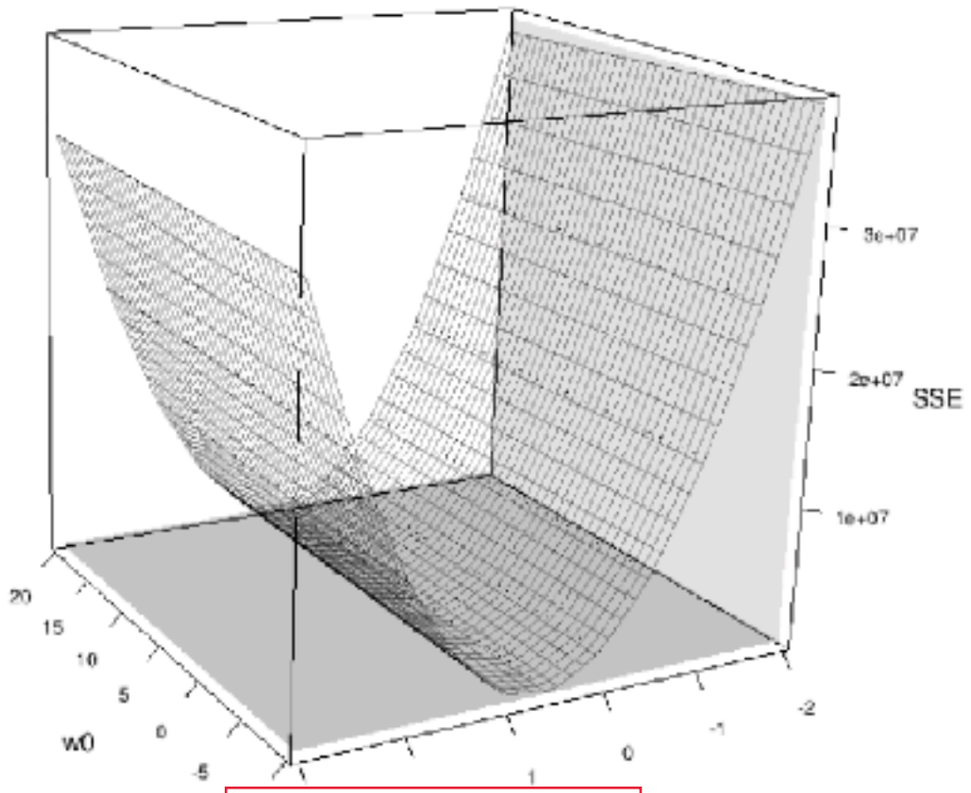
How to find the best parameters ?

Guide Search:
Gradient Descent

梯度下降法

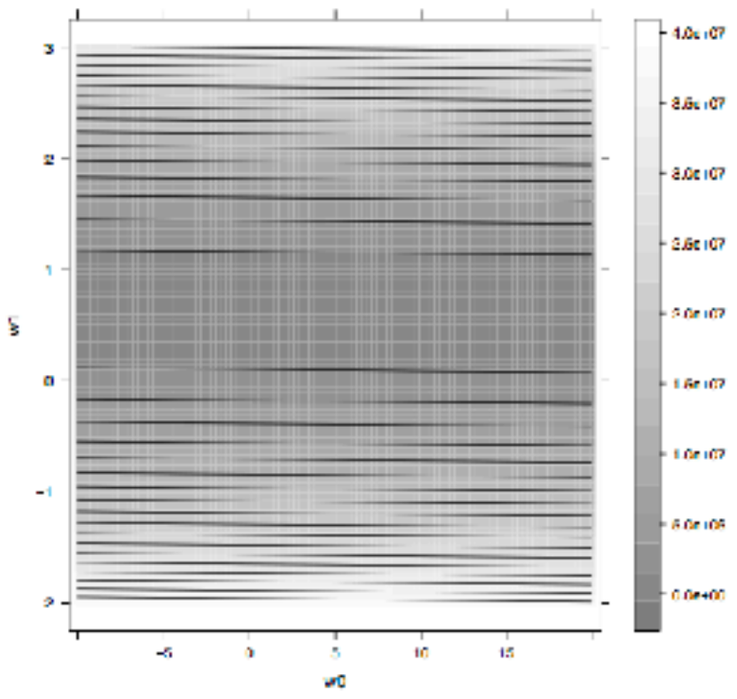

WAYNE GRANTON
THE DISTURBED

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.

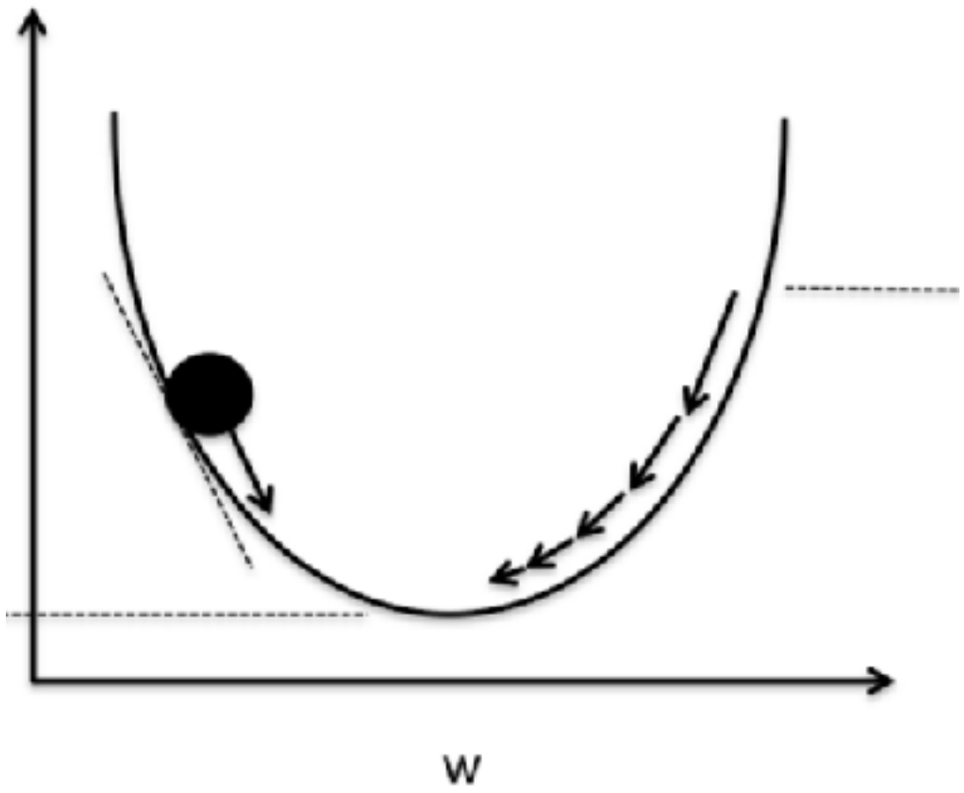


error surface

(a)



(b)



Toy Example

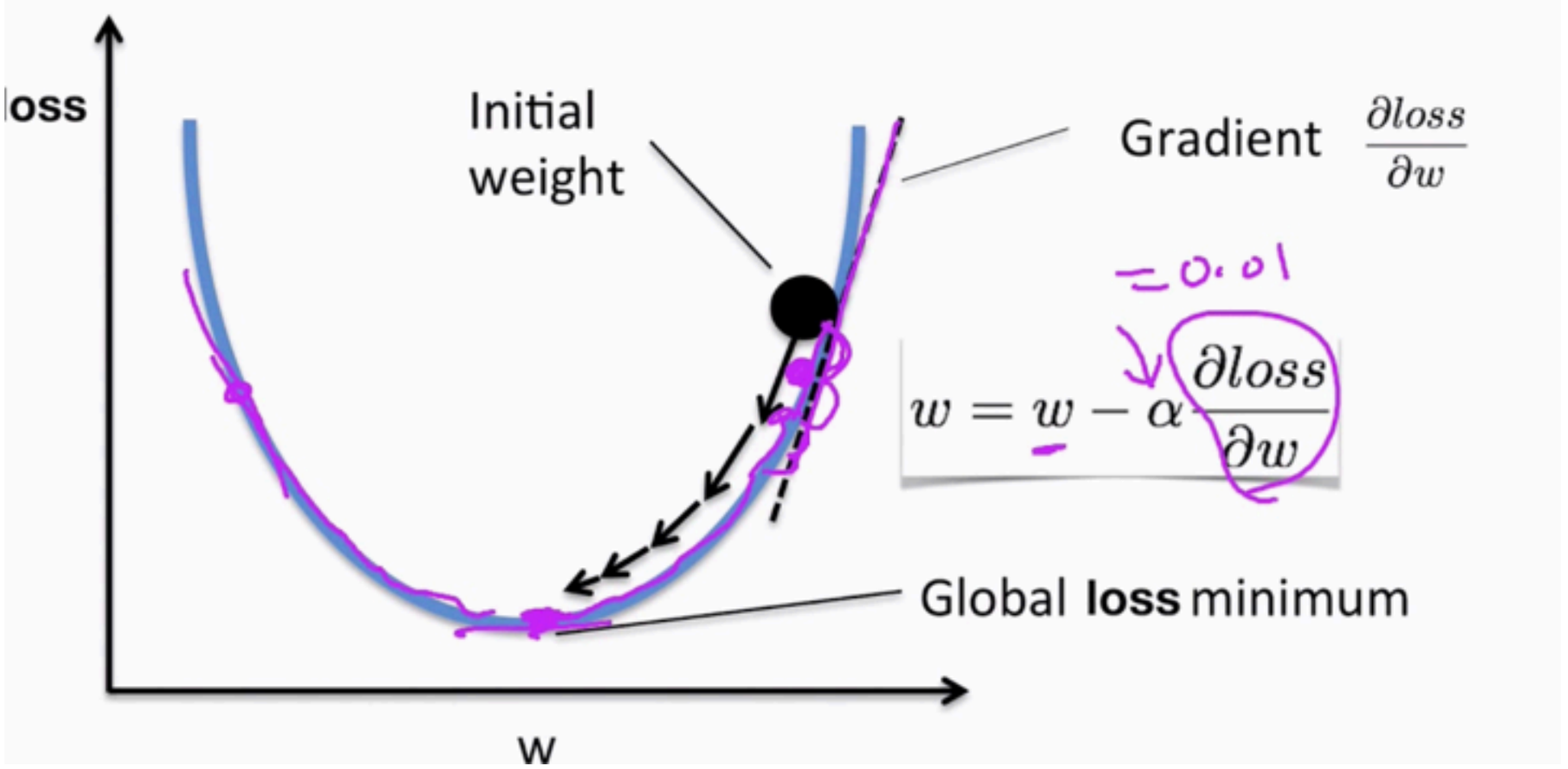
(4,2)

(2,1)

(6,3)

(8,4)

(12,6)



Gradient Descent Algorithm

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, **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

1: **w** $\leftarrow$ random starting point in the weight space
2: **repeat**
3: **for** each **w**[*j*] in **w** **do** update w
4: **w**[*j*] $\leftarrow$ **w**[*j*] $+ \alpha \times$ **errorDelta**($\mathcal{D}$, **w**[*j*]) gradient
5: **end for**
6: **until** convergence occurs

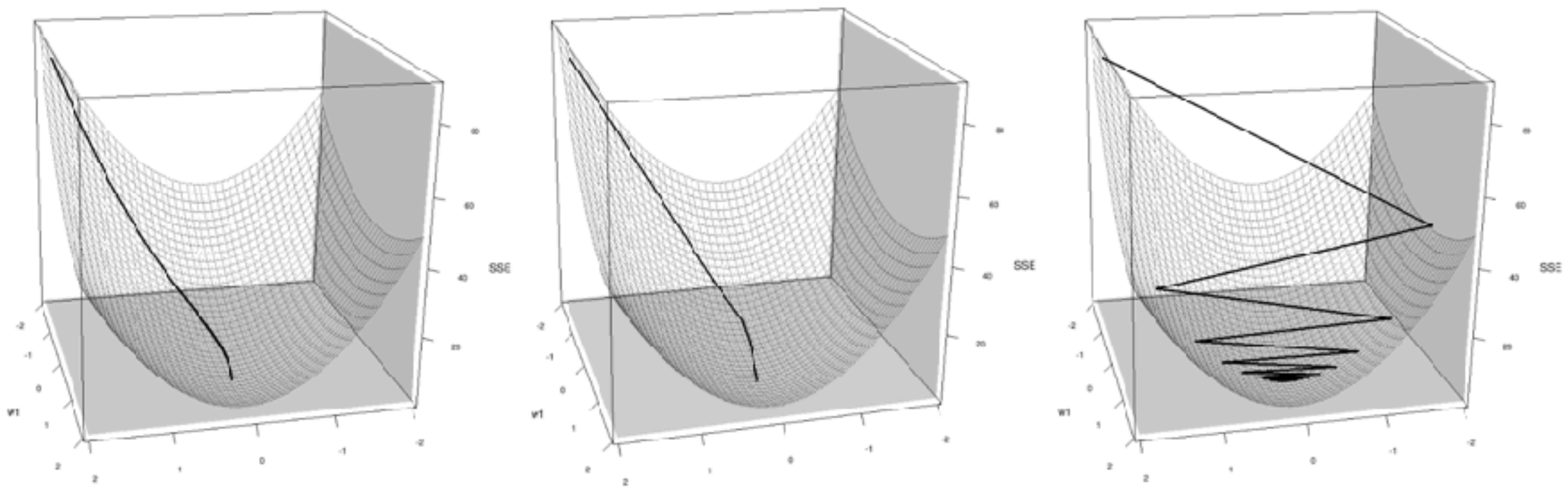
$$\mathbf{w}[j] \leftarrow \mathbf{w}[j] + \alpha \times \mathbf{errorDelta}(\mathcal{D}, \mathbf{w}[j])$$

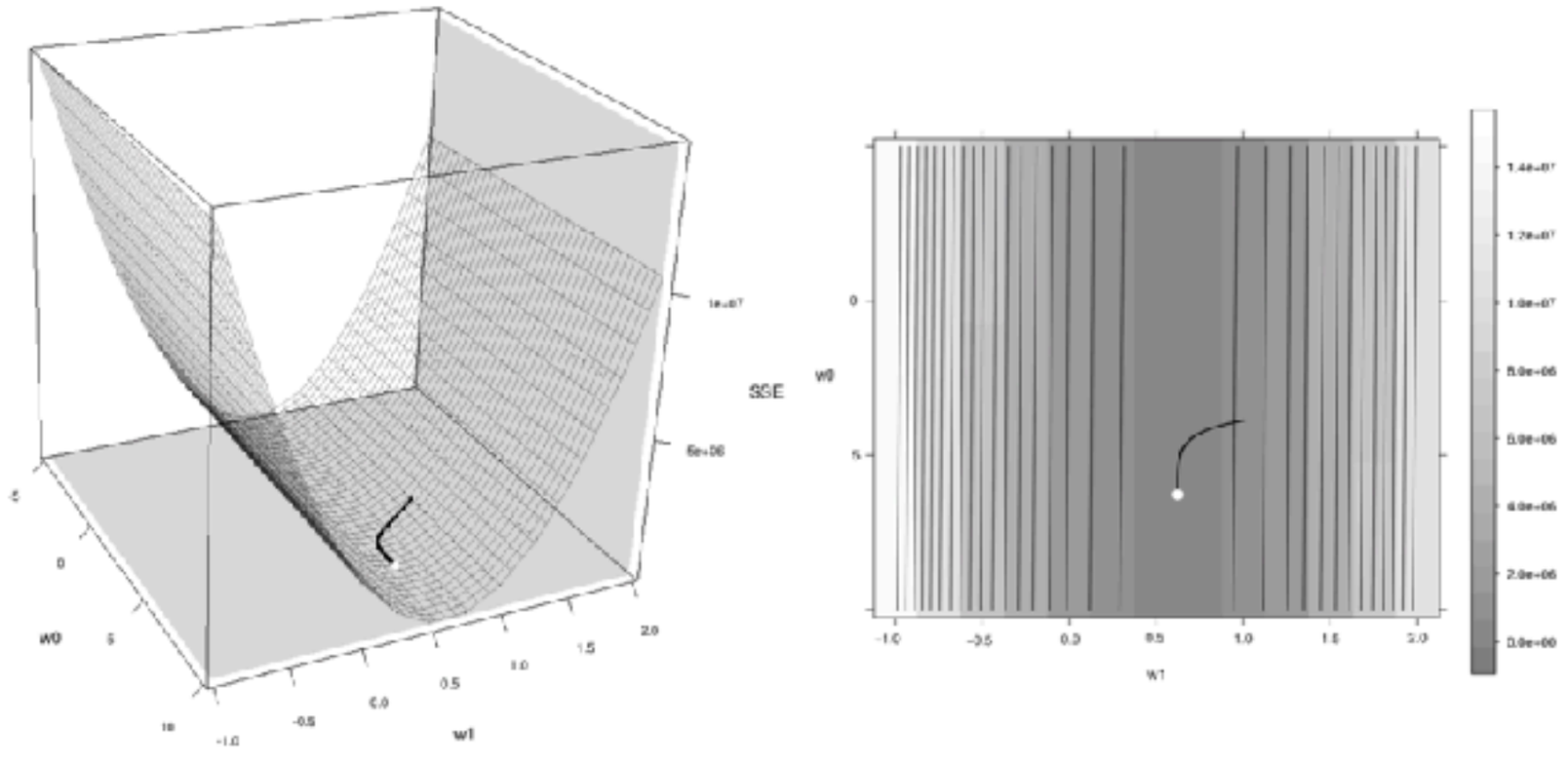
- Each weight is considered independently and for each one a small adjustment is made by adding a small **delta** value to the current weight, **w**[*j*].
- This adjustment should ensure that the change in the weight leads to a move *downwards* on the error surface.

実際値

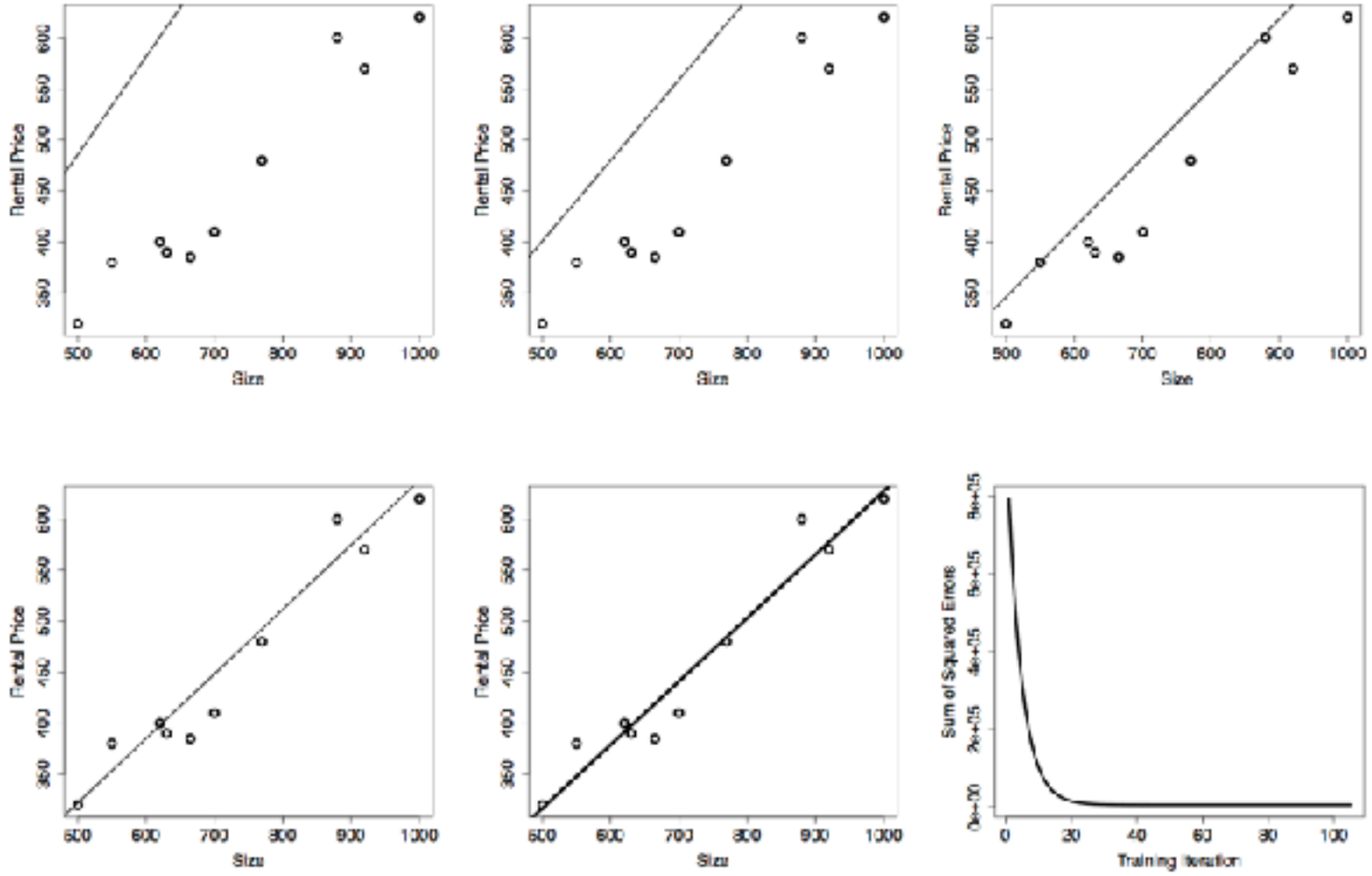
$$\mathbf{w}[j] \leftarrow \mathbf{w}[j] + \alpha \underbrace{\sum_{i=1}^{n} ((t_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i)) \times d_i[j])}_{errorDelta(\mathcal{D},\mathbf{w}[j])}$$

The learning rate α determines the size of the adjustment made to each weight at each step in the process.





The lines indicate the path that the gradient decent algorithm would take across this error surface from different starting positions to the global minimum



A Running Example for Gradient Decent Algorithm

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

For this example let's assume that:

- $\alpha = 0.00000002$

Initial Weights

w[0]:	-0.146	**w**[1]:	0.185	**w**[2]:	-0.044	**w**[3]:	0.119

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

d[0] = 1

Iteration 1

ID	Rental Price	Pred.	Error	Squared Error	**errorDelta**($\mathcal{D}$, **w**[i]) **w**[0]	**w**[1]	**w**[2]	**w**[3]
1	320	93.26	226.74	51411.08	226.74	113370.05	906.96	1813.92
2	380	107.41	272.59	74307.70	272.59	149926.92	1908.16	13629.72
3	400	115.15	284.85	81138.96	284.85	176606.39	2563.64	1993.94
4	390	119.21	270.79	73327.67	270.79	170598.22	1353.95	6498.98
5	385	134.64	250.36	62682.22	250.36	166492.17	2002.91	25036.42
6	410	130.31	279.69	78226.32	279.69	195782.78	1118.76	2237.52
7	480	142.89	337.11	113639.88	337.11	259570.96	3371.05	2359.74
8	600	168.32	431.68	186348.45	431.68	379879.24	5180.17	21584.05
9	570	170.63	399.37	159499.37	399.37	367423.83	5591.23	3194.99
10	620	187.58	432.42	186989.95	432.42	432423.35	3891.81	10378.16
			Sum	1067571.59	3185.61	2412073.90	27888.65	88727.43
Sum of squared errors (Sum/2)				533785.80				

第 j 維

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

226.74*500

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

Initial Weights

w[0]:	-0.146	**w**[1]:	0.185	**w**[2]:	-0.044	**w**[3]:	0.119

$\mathbf{w}[1] = 0.185 + 0.00000002 * 2,412,074 = 0.23324148$

New Weights (Iteration 1)

w[0]:		**w**[1]:	0.233	**w**[2]:		**w**[3]:	

w[0] = -0.146 + 0.00000002*3185.61 ~ -0.146

w[2] = -0.044 + 0.00000002*27888.65 ~ -0.043

w[3] = 0.119 + 0.00000002*88727.43 ~ 0.121

Iteration 1

ID	Rental Price	Pred.	Error	Squared Error	errorDelta($\mathcal{D}$, w[i]) w[0]	w[1]	w[2]	w[3]
1	320	93.26	226.74	51411.08	226.74	113370.05	906.96	1813.92
2	380	107.41	272.59	74307.70	272.59	149926.92	1908.16	13629.72
3	400	115.15	284.85	81138.96	284.85	176606.39	2563.64	1993.94
4	390	119.21	270.79	73327.67	270.79	170598.22	1353.95	6498.98
5	385	134.64	250.36	62682.22	250.36	166492.17	2002.91	25036.42
6	410	130.31	279.69	78226.32	279.69	195782.78	1118.76	2237.52
7	480	142.89	337.11	113639.88	337.11	259570.96	3371.05	2359.74
8	600	168.32	431.68	186348.45	431.68	379879.24	5180.17	21584.05
9	570	170.63	399.37	159499.37	399.37	367423.83	5591.23	3194.99
10	620	187.58	432.42	186989.95	432.42	432423.35	3891.81	10378.16
			Sum	1067571.59	3185.61	2412073.90	27888.65	88727.43
Sum of squared errors (Sum/2)				533785.80				

Iteration 2

ID	Rental Price	Pred.	Error	Squared Error	errorDelta($\mathcal{D}$, w[i]) w[0]	w[1]	w[2]	w[3]
1	320	117.40	202.60	41047.92	202.60	101301.44	810.41	1620.82
2	380	134.03	245.97	60500.69	245.97	135282.89	1721.78	12298.44
3	400	145.08	254.92	64985.12	254.92	158051.51	2294.30	1784.45
4	390	149.65	240.35	57769.68	240.35	151422.55	1201.77	5768.48
5	385	166.90	218.10	47568.31	218.10	145037.57	1744.81	21810.16
6	410	164.10	245.90	60468.86	245.90	172132.91	983.62	1967.23
7	480	180.06	299.94	89964.69	299.94	230954.68	2999.41	2099.59
8	600	210.87	389.13	151424.47	389.13	342437.01	4669.60	19456.65
9	570	215.03	354.97	126003.34	354.97	326571.94	4969.57	2839.76
10	620	187.58	432.42	186989.95	432.42	432423.35	3891.81	10378.16
			Sum	886723.04	2884.32	2195615.84	25287.08	80023.74
Sum of squared errors (Sum/2)				443361.52				

New Weights (Iteration 2)

w[0]:	-0.145	w[1]:	0.277	w[2]:	-0.043	w[3]:	0.123

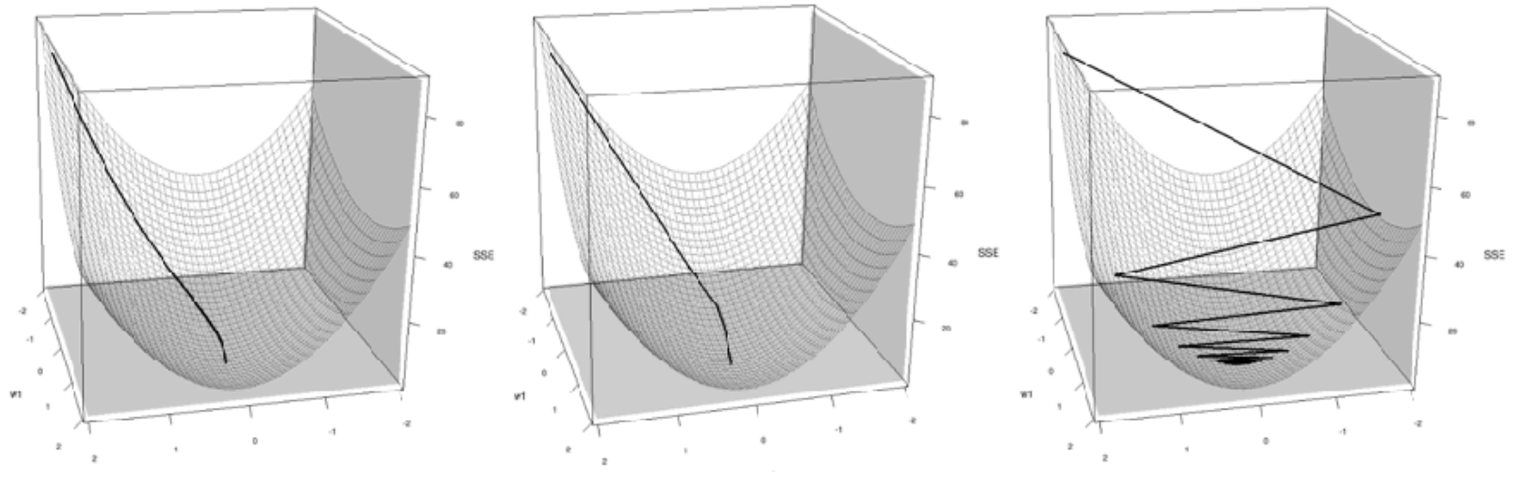
Final Result

The algorithm then keeps iteratively applying the weight update rule until it converges on a stable set of weights beyond which little improvement in model accuracy is possible.

After 100 iterations the final values for the weights are:

- $\mathbf{w}[0] = -0.1513$,
- $\mathbf{w}[1] = 0.6270$,
- $\mathbf{w}[2] = -0.1781$
- $\mathbf{w}[3] = 0.0714$

which results in a sum of squared errors value of $2,913.5$



Handling Categorical Features

Handling Categorical Features

The basic structure of the multivariable linear regression model allows for only continuous descriptive features, so we need a way to handle categorical descriptive features.

The most common approach to handling categorical features uses a transformation that converts a single categorical descriptive feature into a number of continuous descriptive feature values that can encode the levels of the categorical feature.

For example, the ENERGY RATING descriptive feature would be converted into three new continuous descriptive features, as it has 3 distinct levels: *'A'*, *'B'*, or *'C'*.

ID	Size	Floor	Broadband Rate	Energy Rating A	Energy Rating B	Energy Rating C	Rental Price
1	500	4	8	0	0	1	320
2	550	7	50	1	0	0	380
3	620	9	7	1	0	0	400
4	630	5	24	0	1	0	390
5	665	8	100	0	0	1	385
6	700	4	8	0	1	0	410
7	770	10	7	0	1	0	480
8	880	12	50	1	0	0	600
9	920	14	8	0	0	1	570
10	1 000	9	24	0	1	0	620

Returning to our example, the regression equation for this Rental Price model would change to

$$\begin{aligned}\text{Rental Price} = \mathbf{w}[0] &+ \mathbf{w}[1] \times \text{Size} + \mathbf{w}[2] \times \text{Floor} \\ &+ \mathbf{w}[3] \times \text{Broadband Rate} \\ &+ \mathbf{w}[4] \times \text{Energy Rating A} \\ &+ \mathbf{w}[5] \times \text{Energy Rating B} \\ &+ \mathbf{w}[6] \times \text{Energy Rating C}\end{aligned}$$

where the newly added categorical features allow the original Energy Rating feature to be included.

Logistic Regression

-Handling Categorical Target Features

Table: A dataset listing features for a number of generators.

ID	RPM	VIBRATION	STATUS
1	568	585	good
2	586	565	good
3	609	536	good
4	616	492	good
5	632	465	good
6	652	528	good
7	655	496	good
8	660	471	good
9	688	408	good
10	696	399	good
11	708	387	good
12	701	434	good
13	715	506	good
14	732	485	good
15	731	395	good
16	749	398	good
17	759	512	good
18	773	431	good
19	782	456	good
20	797	476	good
21	794	421	good
22	824	452	good
23	835	441	good
24	862	372	good
25	879	340	good
26	892	370	good
27	913	373	good
28	933	330	good

ID	RPM	VIBRATION	STATUS
29	562	309	faulty
30	578	346	faulty
31	593	357	faulty
32	626	341	faulty
33	635	252	faulty
34	658	235	faulty
35	663	299	faulty
36	677	223	faulty
37	685	303	faulty
38	698	197	faulty
39	699	311	faulty
40	712	257	faulty
41	722	193	faulty
42	735	259	faulty
43	738	314	faulty
44	753	113	faulty
45	767	286	faulty
46	771	264	faulty
47	780	137	faulty
48	784	131	faulty
49	798	132	faulty
50	820	152	faulty
51	834	157	faulty
52	858	163	faulty
53	888	91	faulty
54	891	156	faulty
55	911	79	faulty
56	939	99	faulty

Categorical Target Features v.s. Numerical Target Feature

ID	RPM	VIBRATION	STATUS
1	568	585	good
2	586	565	good
3	609	536	good
4	616	492	good
5	632	465	good
6	652	528	good
7	655	496	good
8	660	471	good
9	688	408	good
10	696	399	good
11	708	387	good
12	701	434	good
13	715	506	good
14	732	485	good

ID	RPM	VIBRATION	STATUS
29	562	309	faulty
30	578	346	faulty
31	593	357	faulty
32	626	341	faulty
33	635	252	faulty
34	658	235	faulty
35	663	299	faulty
36	677	223	faulty
37	685	303	faulty
38	698	197	faulty
39	699	311	faulty
40	712	257	faulty
41	722	193	faulty
42	735	259	faulty

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 Regression

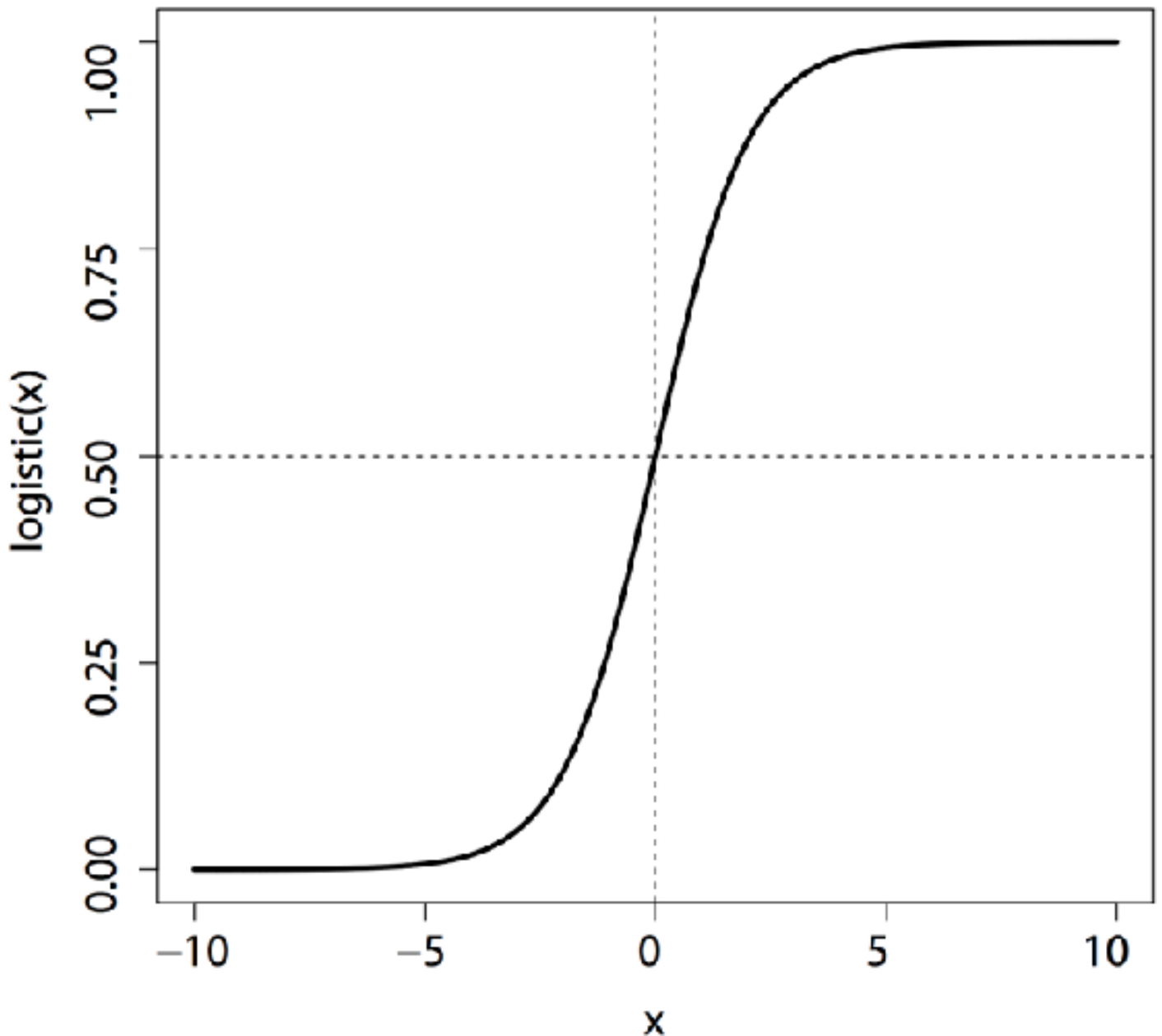
$$
\begin{aligned}
\mathbb{M}_{\mathbf{w}}(\mathbf{d}) &= \mathbf{w}[0] \times \mathbf{d}[0] + \mathbf{w}[1] \times \mathbf{d}[1] + \ldots \\
&= \sum_{j=0}^{m} \mathbf{w}[j] \times \mathbf{d}[j] \\
&= \mathbf{w} \cdot \mathbf{d}
\end{aligned}
$$

Logistic Regression

$$
\begin{aligned}
\mathbb{M}_{\mathbf{w}}(\mathbf{d}) &= \mathit{Logistic}(\mathbf{w} \cdot \mathbf{d}) \\
&= \frac{1}{1 + e^{-\mathbf{w} \cdot \mathbf{d}}}
\end{aligned}
$$

To employ a sophisticated threshold function that is continuous, and therefore differentiable, and that allows for the subtlety desired: the **logistic function**

$$Logistic(x) = \frac{1}{1 + e^{-x}}$$



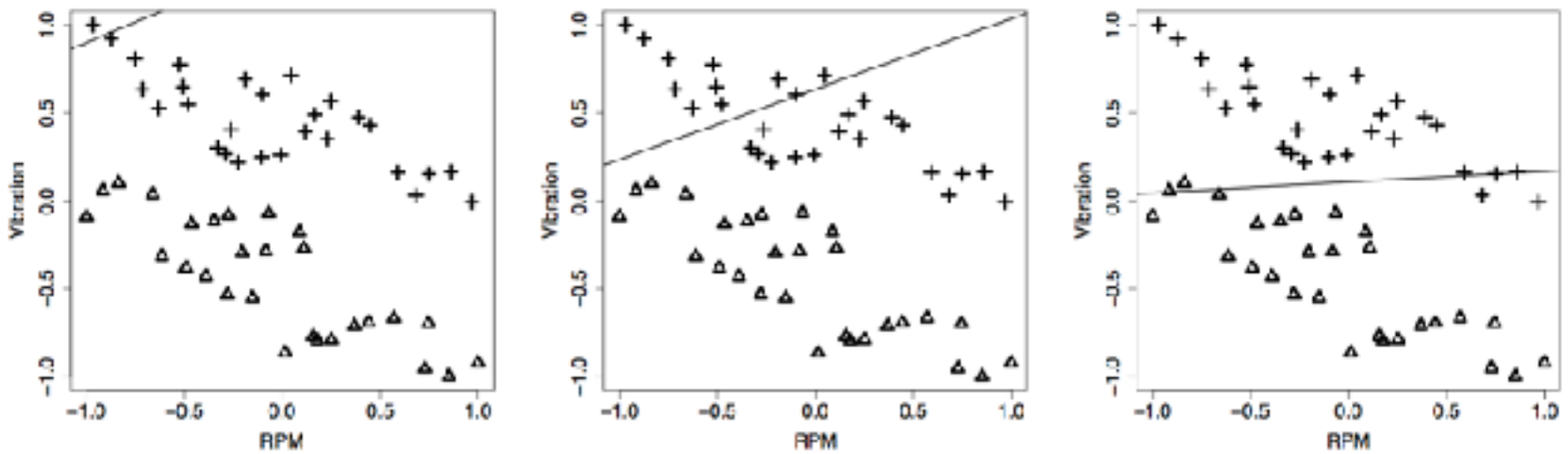
To build a logistic regression model, we simply pass the output of the basic linear regression model through the logistic function

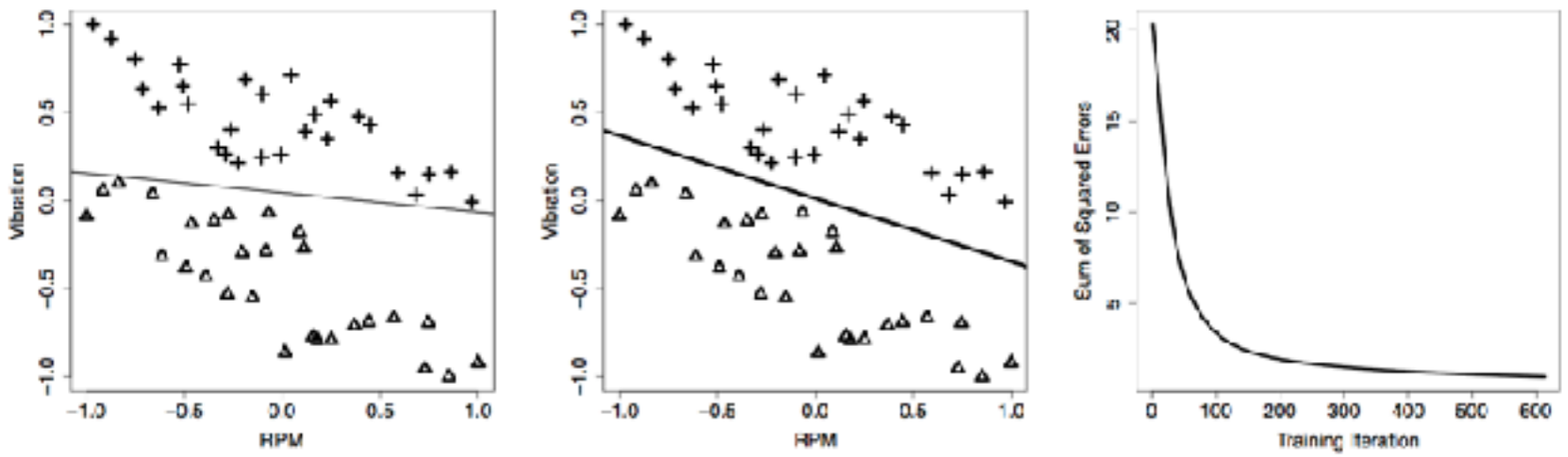
$$\begin{aligned}\mathbb{M}_{\mathbf{w}}(\mathbf{d}) &= \mathit{Logistic}(\mathbf{w} \cdot \mathbf{d}) \\ &= \frac{1}{1 + e^{-\mathbf{w} \cdot \mathbf{d}}}\end{aligned}$$

$$\begin{aligned}&\mathbb{M}_{\mathbf{w}}(\langle \text{RPM}, \text{VIBRATION} \rangle) \\ &= \frac{1}{1 + e^{-(-0.4077 + 4.1697 \times \text{RPM} + 6.0460 \times \text{VIBRATION})}}\end{aligned}$$

The error of the model on each instance is then the difference between the target feature (0 or 1) and the value of the prediction [0, 1]

e.g., {(0, 0.01), (1, 0.99), ….}



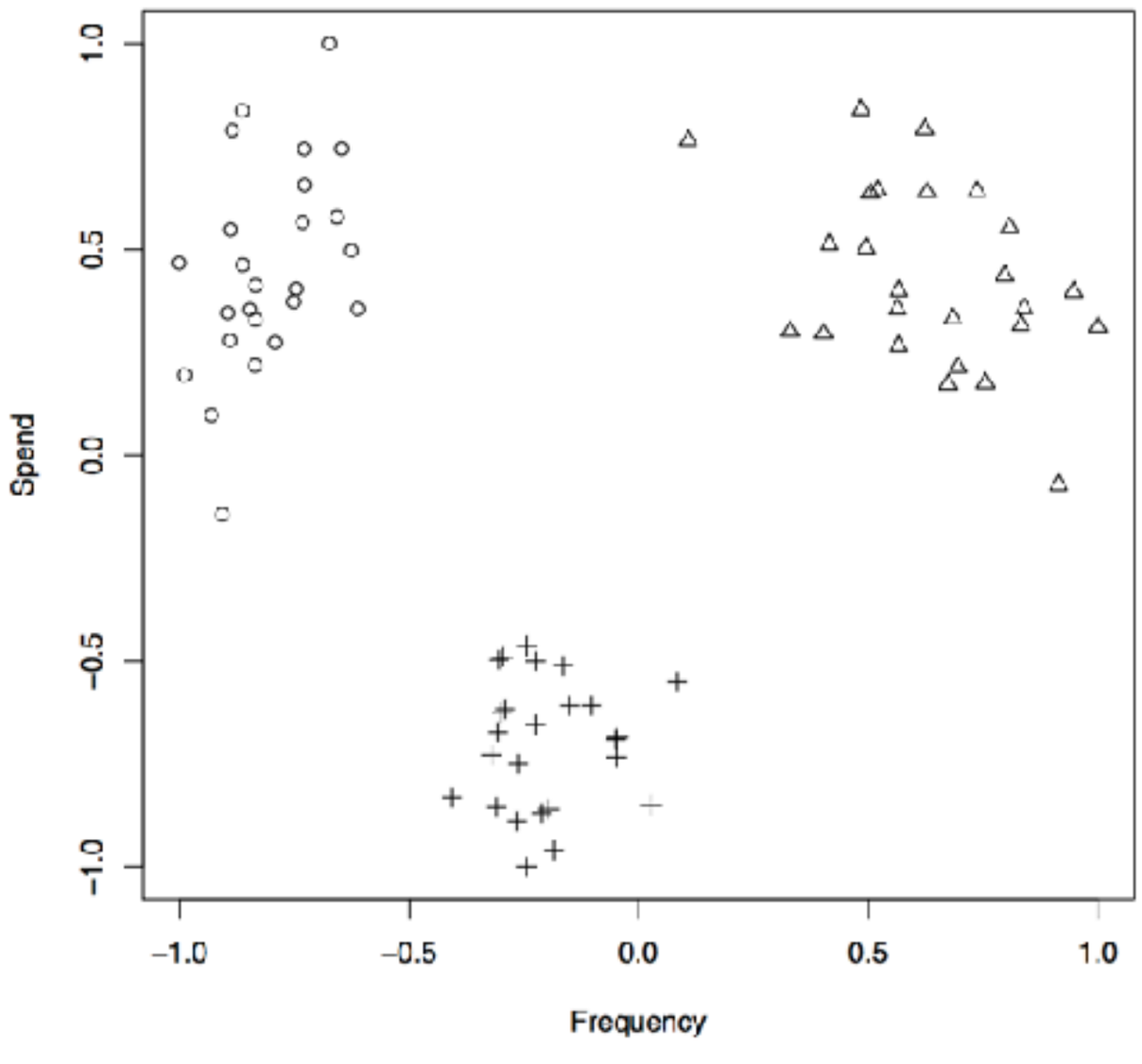


$$\mathbb{M}_{\mathbf{w}}(\langle \text{RPM}, \text{VIBRATION} \rangle) = \frac{1}{1 + e^{-(-0.4077 + 4.1697 \times \text{RPM} + 6.0460 \times \text{VIBRATION})}}$$

Multinomial Logistic Regression

ID	SPEND	FREQ	TYPE
1	21.6	5.4	single
2	25.7	7.1	single
3	18.9	5.6	single
4	25.7	6.8	single
		⋮	
26	107.9	5.8	business
27	92.9	5.5	business

ID	SPEND	FREQ	TYPE
28	122.6	6.0	business
29	107.7	5.7	business
		⋮	
47	53.2	2.6	family
48	52.4	2.0	family
49	46.1	1.4	family
50	65.3	2.2	family

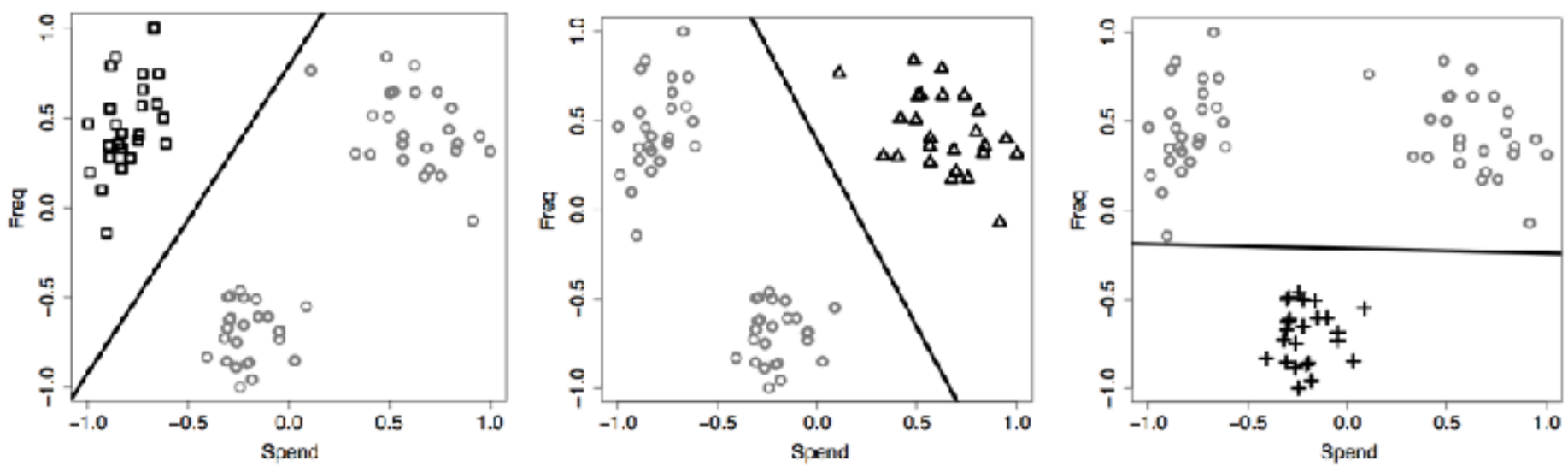


One-versus-all Model for Multinomial Regression

For r target feature levels, we build r separate logistic regression models $\mathbb{M}_{\mathbf{w}_1}$ to $\mathbb{M}_{\mathbf{w}_r}$:

$$\mathbb{M}_{\mathbf{w}_1}(\mathbf{d}) = \mathit{logistic}(\mathbf{w}_1 \cdot \mathbf{d})$$
$$\mathbb{M}_{\mathbf{w}_2}(\mathbf{d}) = \mathit{logistic}(\mathbf{w}_2 \cdot \mathbf{d})$$
$$\vdots$$
$$\mathbb{M}_{\mathbf{w}_r}(\mathbf{d}) = \mathit{logistic}(\mathbf{w}_r \cdot \mathbf{d})$$

where $\mathbb{M}_{\mathbf{w}_1}$ to $\mathbb{M}_{\mathbf{w}_r}$ are r different one-versus-all logistic regression models, and $\mathbf{w}_1$ to $\mathbf{w}_r$ are r different sets of weights.



(a) (b) (c)

$$\mathbb{M}_{\mathbf{w}_1}(\mathbf{d}) = \mathit{logistic}(\mathbf{w_1} \cdot \mathbf{d})$$
$$\mathbb{M}_{\mathbf{w}_2}(\mathbf{d}) = \mathit{logistic}(\mathbf{w_2} \cdot \mathbf{d})$$
$$\vdots$$
$$\mathbb{M}_{\mathbf{w}_r}(\mathbf{d}) = \mathit{logistic}(\mathbf{w_r} \cdot \mathbf{d})$$

$$\mathbb{M}'_{\mathbf{w}_k}(\mathbf{d}) = \frac{\mathbb{M}_{\mathbf{w}_k}(\mathbf{d})}{\sum_{l \in \mathit{levels}(t)} \mathbb{M}_{\mathbf{w}_l}(\mathbf{d})}$$

A revised, normalized prediction for the one-versus-all model for the target level *k*

e.g.,

$$\mathbb{M}_{\mathbf{w}_{'single'}}(\mathbf{q}) = \mathit{Logistic}(0.7993 - 15.9030 \times \textsc{Spend} + 9.5974 \times \textsc{Freq})$$
$$\mathbb{M}_{\mathbf{w}_{'family'}}(\mathbf{q}) = \mathit{Logistic}(3.6526 + -0.5809 \times \textsc{Spend} - 17.5886 \times \textsc{Freq})$$
$$\mathbb{M}_{\mathbf{w}_{'business'}}(\mathbf{q}) = \mathit{Logistic}(4.6419 + 14.9401 \times \textsc{Spend} - 6.9457 \times \textsc{Freq})$$

e.g.,

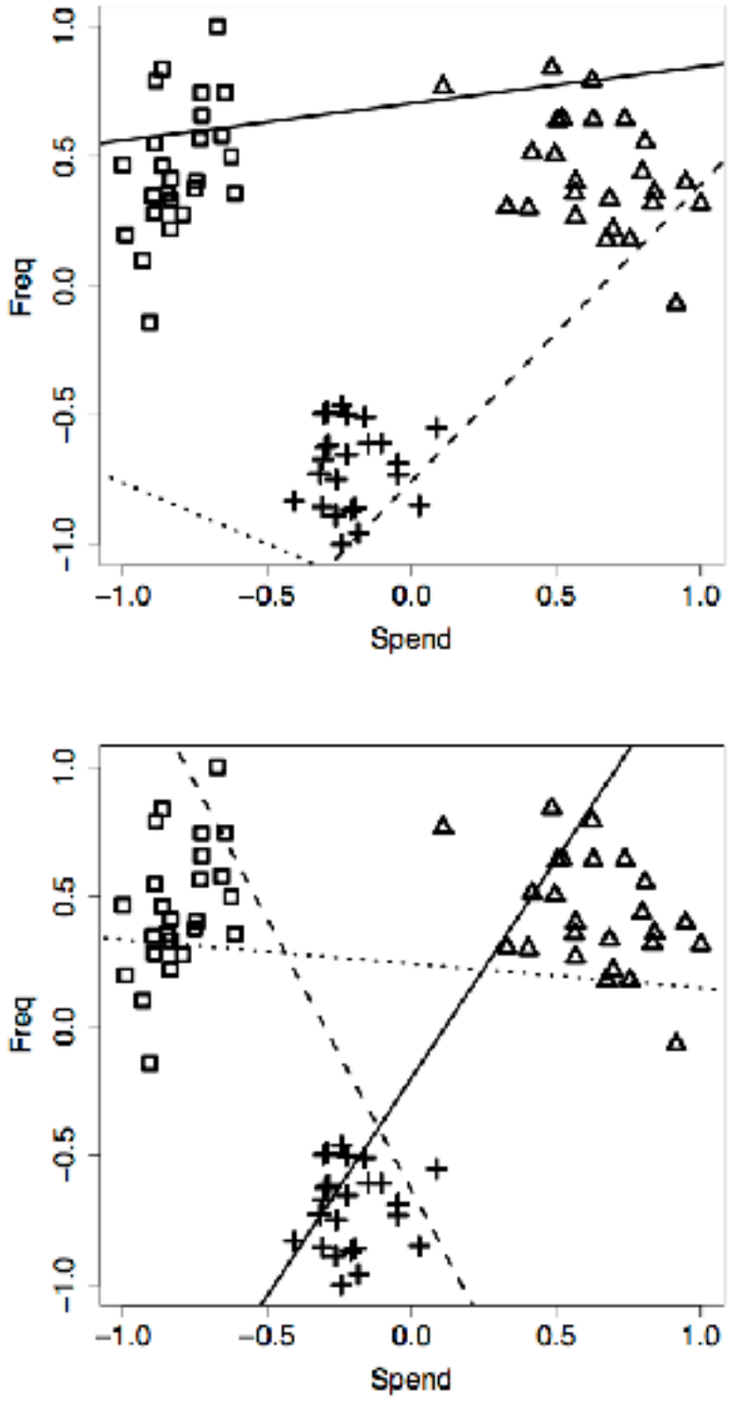
$$\mathbb{M}'_{\mathbf{w}_{'single'}}(\mathbf{q}) = \frac{0.9999}{0.9999 + 0.01278 + 0.0518} = 0.9393$$
$$\mathbb{M}'_{\mathbf{w}_{'family'}}(\mathbf{q}) = \frac{0.01278}{0.9999 + 0.01278 + 0.0518} = 0.0120$$
$$\mathbb{M}'_{\mathbf{w}_{'business'}}(\mathbf{q}) = \frac{0.0518}{0.9999 + 0.01278 + 0.0518} = 0.0487$$

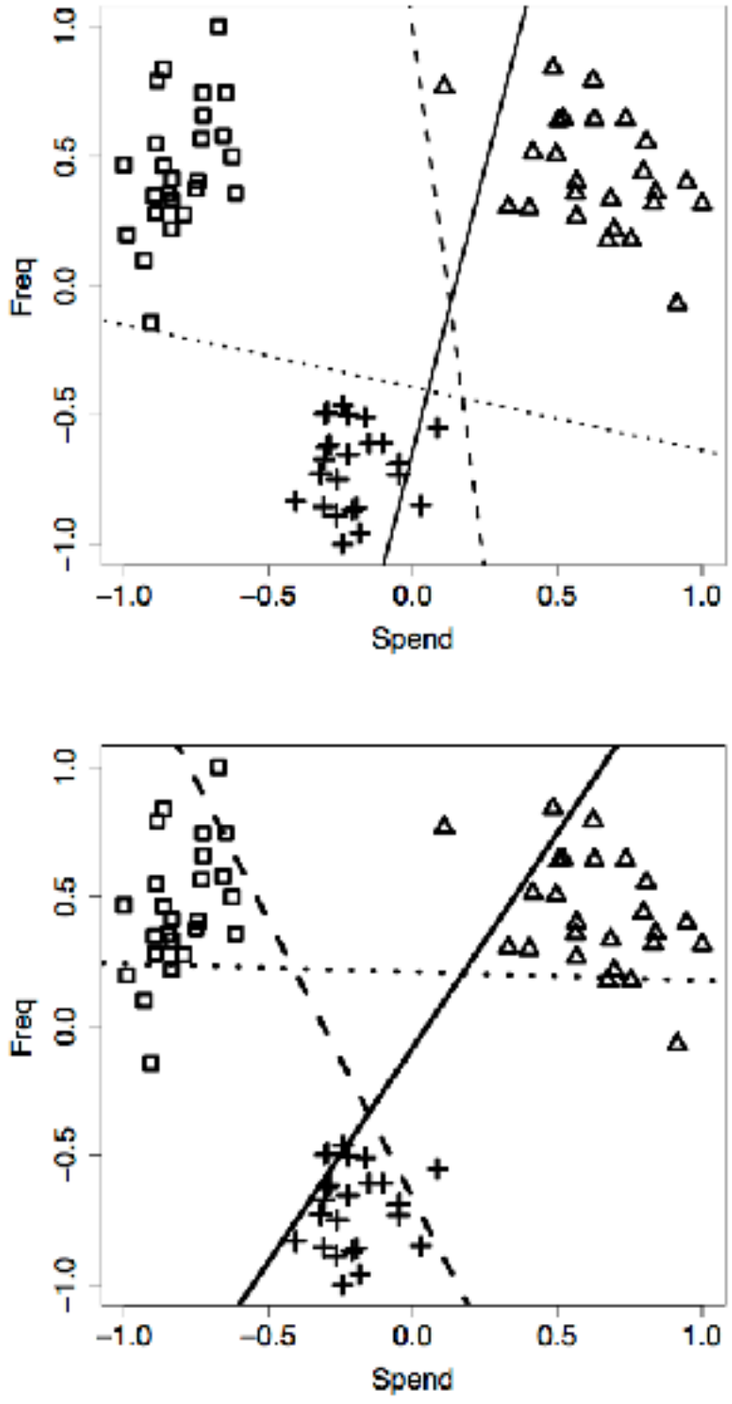
For a query instance with SPEND $= 25.67$ and FREQ $= 6.12$, which are normalized to SPEND $= -0.7279$ and FREQ $= 0.4789$, the predictions of the individual models would be

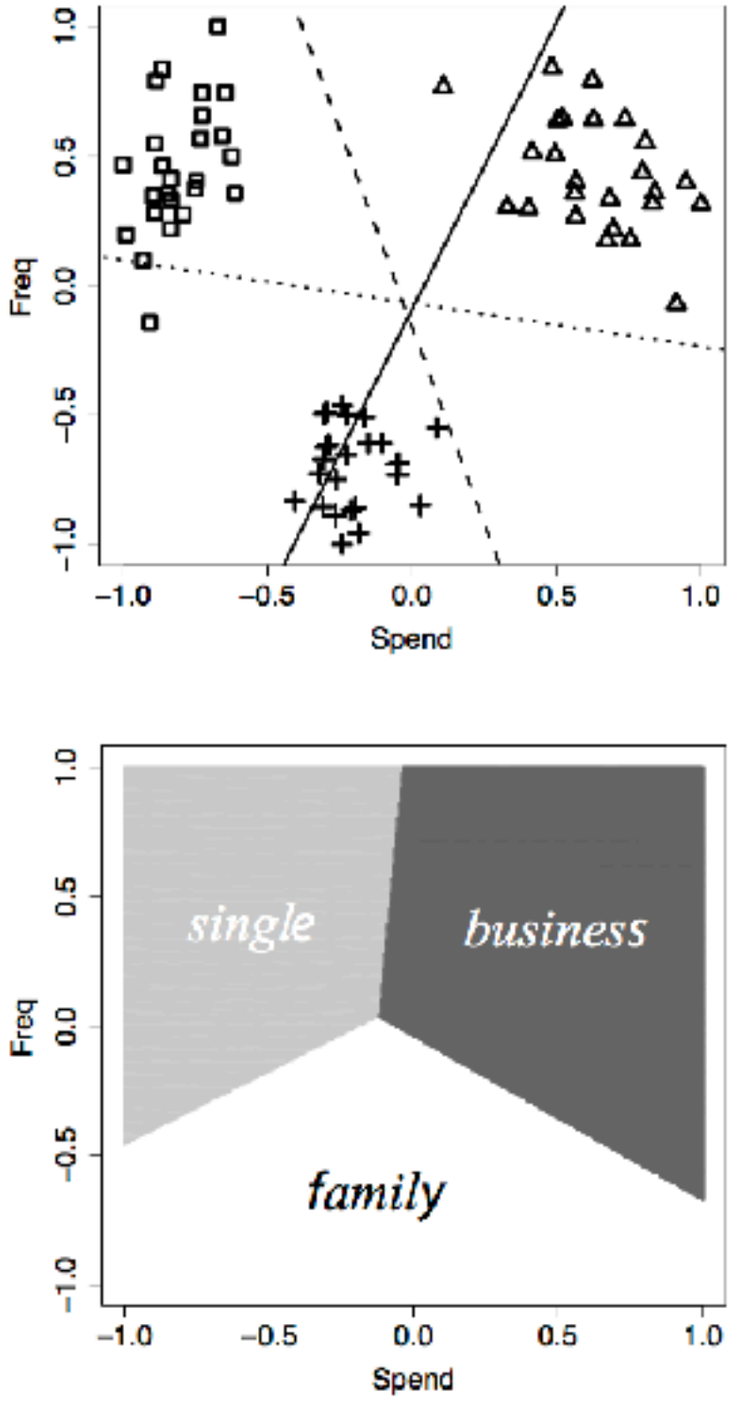
$$\begin{aligned}
\mathbb{M}_{\mathbf{w}_{'single'}}(\mathbf{q}) &= \mathit{Logistic}(0.7993 - 15.9030 \times (-0.7279) + 9.5974 \times 0.4789) \\
&= 0.9999 \\
\mathbb{M}_{\mathbf{w}_{'family'}}(\mathbf{q}) &= \mathit{Logistic}(3.6526 + -0.5809 \times (-0.7279) - 17.5886 \times 0.4789) \\
&= 0.01278 \\
\mathbb{M}_{\mathbf{w}_{'business'}}(\mathbf{q}) &= \\
&=
\end{aligned}$$

$$\mathbb{M}(\mathbf{q}) = \underset{l \in \mathit{levels}(t)}{\operatorname{argmax}} \mathbb{M}'_{\mathbf{w}_l}(\mathbf{q})$$

$$\begin{aligned}
\mathbb{M}'_{\mathbf{w}_{'single'}}(\mathbf{q}) &= \frac{0.9999}{0.9999 + 0.01278 + 0.0518} = 0.9393 \\
\mathbb{M}'_{\mathbf{w}_{'family'}}(\mathbf{q}) &= \frac{0.01278}{0.9999 + 0.01278 + 0.0518} = 0.0120 \\
\mathbb{M}'_{\mathbf{w}_{'business'}}(\mathbf{q}) &= \frac{0.0518}{0.9999 + 0.01278 + 0.0518} = 0.0487
\end{aligned}$$


Freq
Spend
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0


Freq
Spend
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0


single
business
family
Freq
Spend
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0