Classification using Logistic Regression I

Swati Mishra

Applications of Machine Learning (4AL3)

Fall 2024

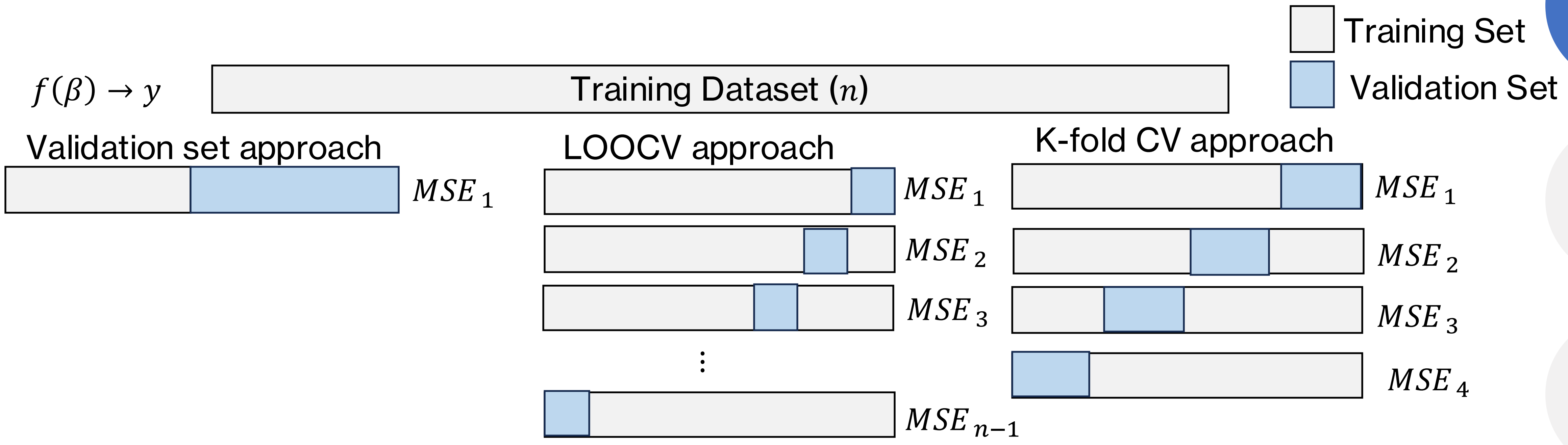
McMaster University

ENGINEERING

Review

- Polynomial Regression – Fundamental and Implementation
- Test MSE, Train MSE, Overfitting and Underfitting
- Brief encounter with Bias Variance Trade Off
- Cross Validation (Validation Set Approach, LOOCV, k-fold CV)

Review Evaluation



Review Evaluation

Training Set

Validation Set

$f(\beta) \rightarrow y$ Training Dataset (n)

Validation set approach

MSE_1

LOOCV approach

MSE_1

MSE_2

MSE_3

⋮

MSE_{n-1}

K-fold CV approach

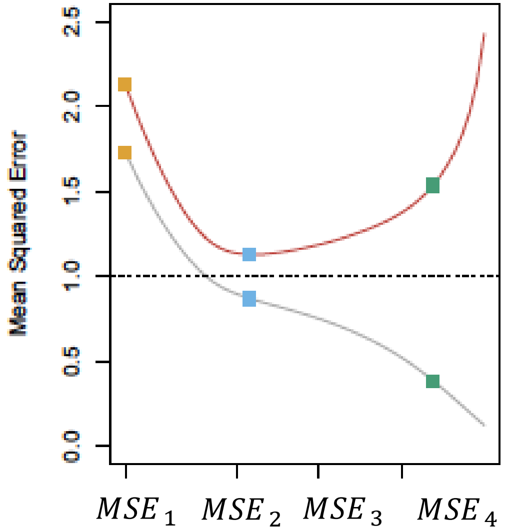
MSE_1

MSE_2

MSE_3

MSE_4

Plot the MSE_i



Final performance = Avg(MSE_i)

Review Evaluation

Training Set

Validation Set

$f(\beta) \rightarrow y$ Training Dataset (n)

MSE_1

MSE_1

MSE_2

MSE_3

⋮

MSE_{n-1}

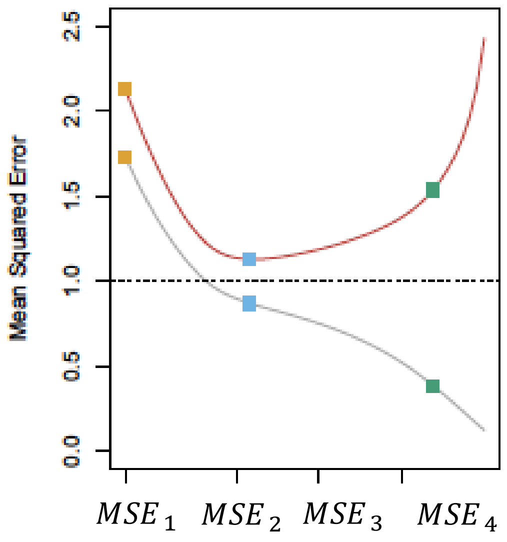
MSE_1

MSE_2

MSE_3

MSE_4

Plot the MSE_i



Final performance = Avg(MSE_i)

Simple models trained on different samples of the data do not differ much from each other (Underfitting)

Complex models trained on different samples of the data are very different from each other (high variance)

McMaster University

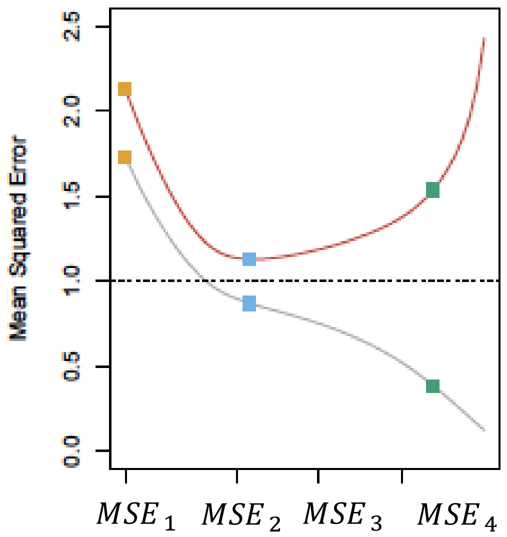
Review Evaluation

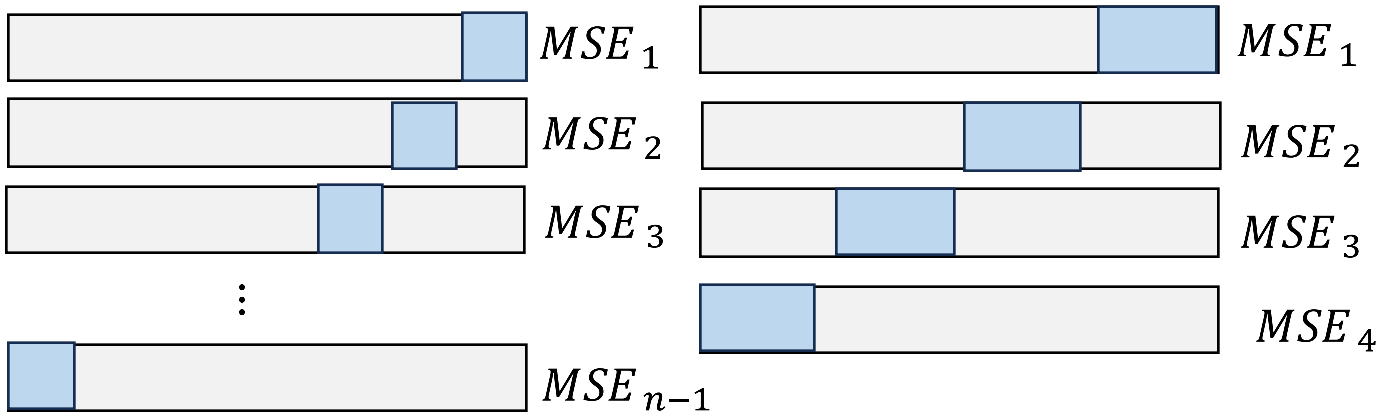
Training Set

Validation Set

$f(\beta) \rightarrow y$ | Training Dataset (n)

Plot the MSE_i





Final performance = Avg(MSE_i)

Simple models trained on different samples of the data do not differ much from each other (Underfitting)

Complex models trained on different samples of the data are very different from each other (high variance)

McMaster University

Classification problems

- So far, our goal was to study where y (target) is a continuous quantitative value and x_i (input features) are also quantitative

$$f(x_1, x_2, \dots, x_p) \rightarrow y$$

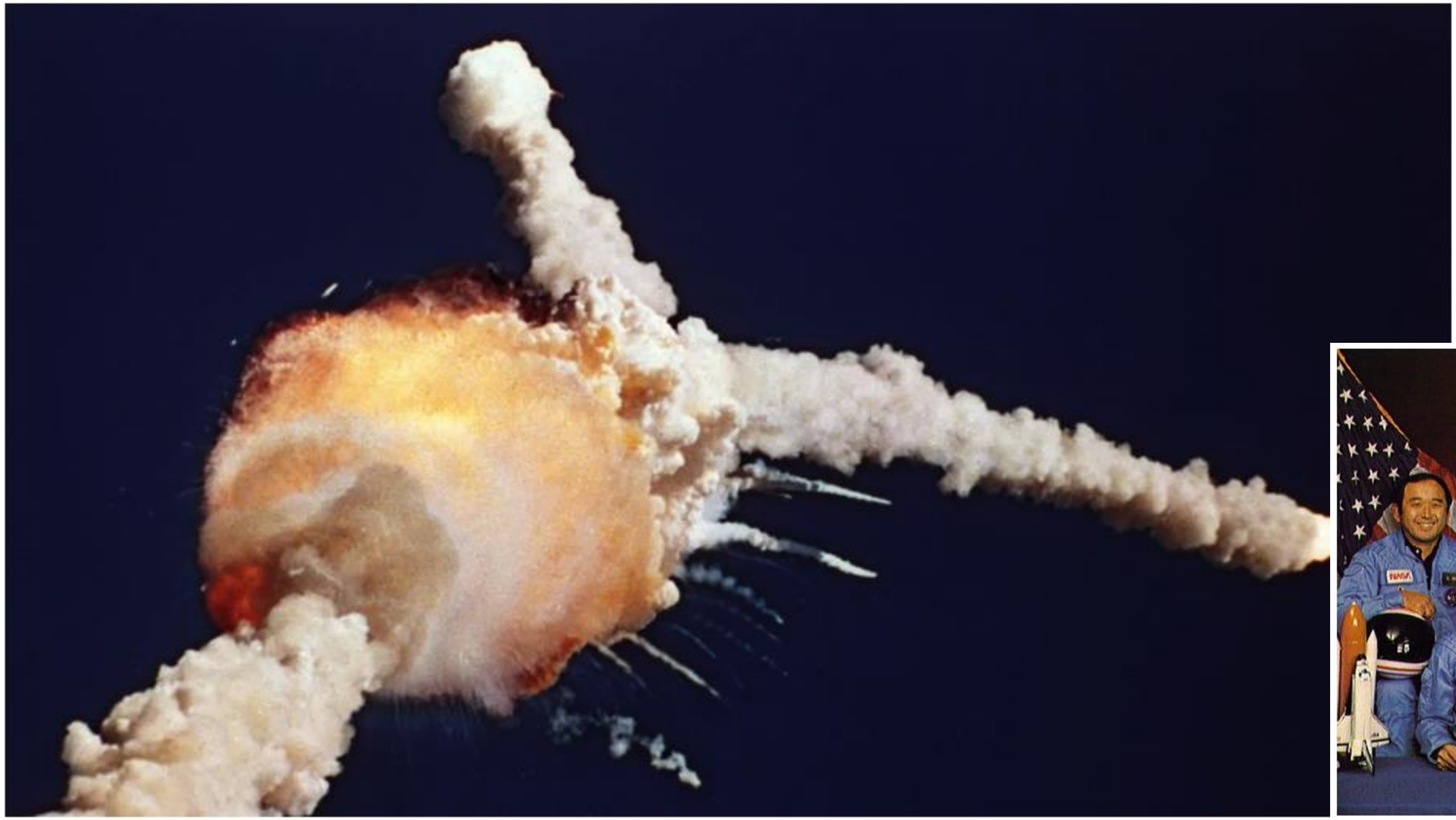
Classification problems

- So far, our goal was to study where y (target) is a continuous quantitative value and x_i (input features) are also quantitative

$$f(x_1, x_2, \ldots, x_p) \rightarrow y$$

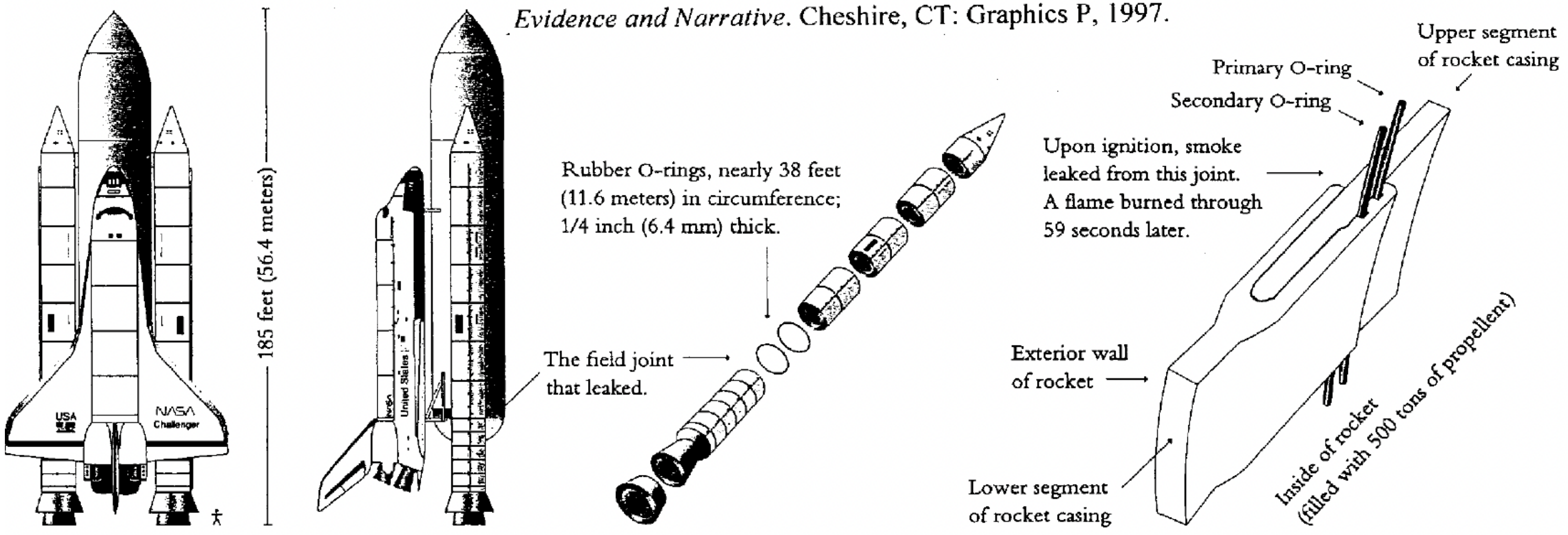
What if y is categorical variable ?

The Challenger Disaster



The Challenger Disaster

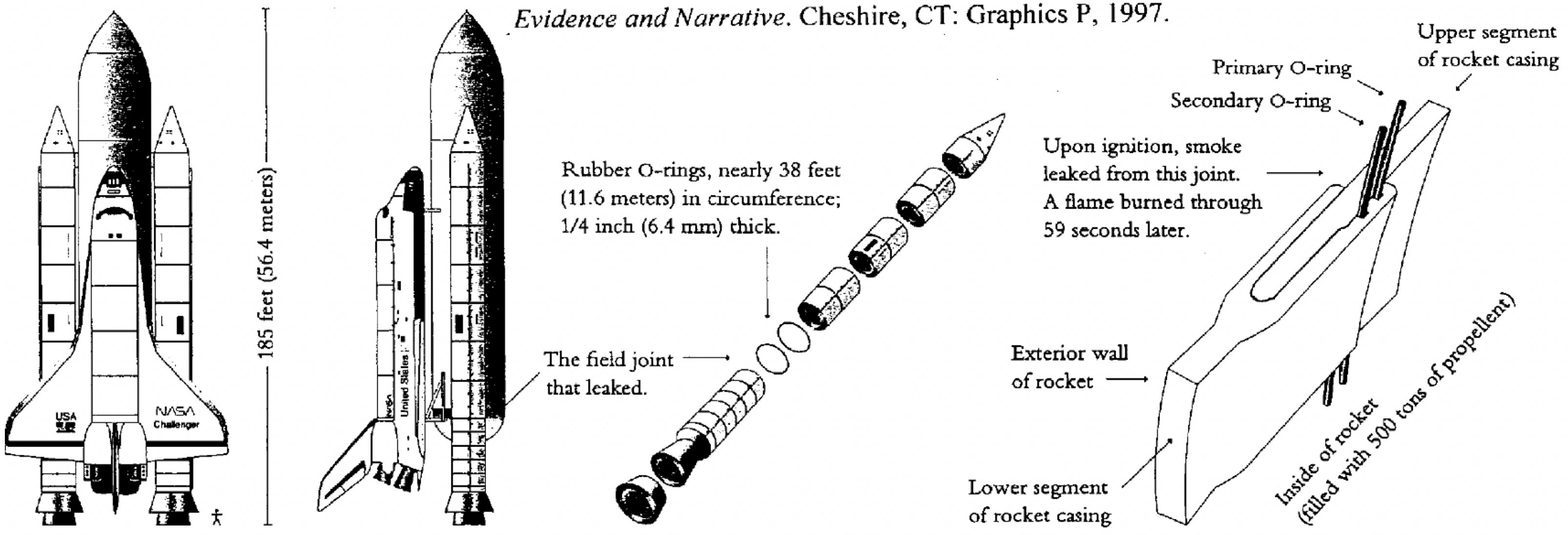
From Tufte, Edward. *Visual Explanations: Images and Quantities, Evidence and Narrative*. Cheshire, CT: Graphics P, 1997.



The shuttle consists of an *orbiter* (which carries the crew and has powerful engines in the back), a large liquid-fuel *tank* for the orbiter engines, and 2 solid-fuel *booster rockets* mounted on the sides of the central tank. Segments of the booster rockets are shipped to the launch site, where they are assembled to make the solid-fuel rockets. Where these segments mate, each joint is sealed by two rubber O-rings as shown above. In the case of the Challenger accident, one of these joints leaked, and a torch-like flame burned through the side of the booster rocket.

The Challenger Disaster

From Tufte, Edward. *Visual Explanations: Images and Quantities, Evidence and Narrative*. Cheshire, CT: Graphics P, 1997.



The shuttle consists of an *orbiter* (which carries the crew and has powerful engines in the back), a large liquid-fuel *tank* for the orbiter engines, and 2 solid-fuel *booster rockets* mounted on the sides of the central tank. Segments of the booster rockets are shipped to the launch site, where they are assembled to make the solid-fuel rockets. Where these segments mate, each joint is sealed by two rubber O-rings as shown above. In the case of the Challenger accident, one of these joints leaked, and a torch-like flame burned through the side of the booster rocket.

Will the O-rings fail catastrophically on the launch day because of the cold weather?

The Challenger Disaster

HISTORY OF O-RING DAMAGE ON SRM FIELD JOINTS

Oct 30, 1985

AFT

	SRM No.	Cross Sectional View: Erosion Depth (in.)	Perimeter Affected (deg)	Nominal Dia. (in.)	Top View: Length Of Max Erosion (in.)	Total Heat Affected Length (in.)	Clocking Location (deg)
61A LH Center Field**	22A	None	None	0.280	None	None	36° --66°
61A LH ~~CENTER~~ FIELD**	22A	NONE	NONE	0.280	NONE	NONE	338°-18°
51C LH Forward Field**	15A	0.010	154.0	0.280	4.25	5.25	163
51C RH Center Field (prim)***	15B	0.038	130.0	0.280	12.50	58.75	354
51C RH Center Field (sec)***	15B	None	45.0	0.280	None	29.50	354
41D RH Forward Field	13B	0.028	110.0	0.280	3.00	None	275
41C LH Aft Field*	11A	None	None	0.280	None	None	--
41B LH Forward Field	10A	0.040	217.0	0.280	3.00	14.50	351
July STS-2 RH Aft Field	2B	0.053	116.0	0.280	--	--	90

*Hot gas path detected in putty. Indication of heat on O-ring, but no damage.
**Soot behind primary O-ring.
***Soot behind primary O-ring, heat affected secondary O-ring.

BLOW BY HISTORY

SRM-15 WORST BLOW-BY

- 2 CASE JOINTS (80°), (110°) ARC
- MUCH WORSE VISUALLY THAN SRM-22

SRM 22 BLOW-BY

- 2 CASE JOINTS (30-40°)

SRM-13A, 15, 16A, 18, 23A 24A

- NOZZLE BLOW-BY

HISTORY OF O-RING TEMPERATURES (DEGREES - F)

MOTOR	MBT	AMB	O-RING	WIND
DM-4	68	36	47	10 MPH
DM-2	76	45	52	10 MPH
QM-3	72.5	40	48	10 MPH
QM-4	76	48	51	10 MPH
SRM-15	52	64	53	10 MPH
SRM-22	77	78	75	10 MPH
SRM-25	55	26	29 27	10 MPH 25 MPH

Flight	Date	Temperature °F	Erosion incidents	Blow-by incidents	Damage index	Comments
51-C	01.24.85	53°	3	2	11	Most erosion any flight; blow-by; back-up rings heated.
41-B	02.03.84	57°	1		4	Deep, extensive erosion.
61-C	01.12.86	58°	1		4	O-ring erosion on launch two weeks before Challenger.
41-C	04.06.84	63°	1		2	O-rings showed signs of heating, but no damage.
1	04.12.81	66°			0	Coolest (66°) launch without O-ring problems.
6	04.04.83	67°			0	
51-A	11.08.84	67°			0	
51-D	04.12.85	67°			0	
5	11.11.82	68°			0	
3	03.22.82	69°			0	
2	11.12.81	70°	1		4	Extent of erosion not fully known.
9	11.28.83	70°			0	
41-D	08.30.84	70°	1		4	
51-G	06.17.85	70°			0	
7	06.18.83	72°			0	
8	08.30.83	73°			0	
51-B	04.29.85	75°			0	
61-A	10.30.85	75°		2	4	No erosion. Soot found behind two primary O-rings.
51-I	08.27.85	76°			0	
61-B	11.26.85	76°			0	
41-G	10.05.84	78°			0	
51-J	10.03.85	79°			0	
4	06.27.82	80°			?	O-ring condition unknown; rocket casing lost at sea.
51-F	07.29.85	81°			0	

Analysis for Go-no-go!

The Challenger Disaster

Date	Launch Temp (F)	Leak Check Pressure	Thermal distress	Number of O-rings
4/12/81	66	50	0	6
11/12/81	70	50	1	6
3/22/82	69	50	0	6
11/11/82	68	50	0	6
4/4/83	67	50	0	6
6/18/83	72	50	0	6
8/30/83	73	50	0	6
11/28/83	70	100	0	6
2/3/84	57	100	1	6
4/6/84	63	200	1	6
8/30/84	70	200	1	6
10/5/84	78	200	0	6

Date	Launch Temp (F)	Leak Check Pressure	Thermal distress	Number of O-rings
11/8/84	67	200	0	6
1/24/85	53	200	2	6
4/12/85	67	200	0	6
4/29/85	75	200	0	6
6/17/85	70	200	0	6
7/29/85	81	200	0	6
8/27/85	76	200	0	6
10/3/85	79	200	0	6
10/30/85	75	200	2	6
11/26/85	76	200	0	6
1/12/86	58	200	1	6

Binary Classification

Linear model equation: $y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$

McMaster University

Binary Classification

Linear model equation: $y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$

- What happens, if the right side of the equation was a not a continuous set of values, but, binary values?

Binary Classification

Linear model equation: $y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$

- What happens, if the right side of the equation was a not a continuous set of values, but, binary values?

- In this case, the right side is continuous unbounded, but the left side is binary, which means our y is either 0 or 1

Logistic Regression

Linear model equation: $y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$

- What happens, if the right side of the equation was a not a continuous set of values, but, binary values?
- In this case, the right side is continuous unbounded, but the left side is binary, which means our y is either 0 or 1.
- Our goal is to predict the **probability** that a given instance belongs to 1 class of y. Therefore, we can modify the above equation to,

Logistic Regression

Linear model equation: $y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$

- What happens, if the right side of the equation was a not a continuous set of values, but, binary values?
- In this case, the right side is continuous unbounded, but the left side is binary, which means our y is either 0 or 1.
- Our goal is to predict the **probability** that a given instance belongs to 1 class of y. Therefore, we can modify the above equation to,

$$P(y = 1|x) \propto \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

Logistic Regression

Linear model equation: $y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$

- What happens, if the right side of the equation was a not a continuous set of values, but, binary values?
- In this case, the right side is continuous unbounded, but the left side is binary, which means our y is either 0 or 1.
- Our goal is to predict the **probability** that a given instance belongs to 1 class of y. Therefore, we can modify the above equation to,

1 = success ; 0 = failure

$$P(y = 1|x) \propto \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

$P(y = 1|x)$ means the probability that at least one O-Ring has failed

McMaster University

Logistic Regression

$$P(y = 1|x) \propto \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

- Above equation compares two inputs and identifies which of them leads to a higher probability of y belonging to a given class, in other words, we can **rank these probabilities**

Logistic Regression

$$P(y = 1|x) \propto \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

- Above equation compares two inputs and identifies which of them leads to a higher probability of y belonging to a given class, in other words, we can **rank these probabilities**

- So far, we know that β is a parameter matrix, and x represents the features.

Logistic Regression

$$P(y = 1|x) \propto \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

- Above equation compares two inputs and identifies which of them leads to a higher probability of y belonging to a given class, in other words, we can **rank these probabilities**

- So far, we know that β is a parameter matrix, and x represents the features. Therefore, above equation can be re-written as

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

Logistic Regression

$$P(y = 1|x) \propto \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

- Above equation compares two inputs and identifies which of them leads to a higher probability of y belonging to a given class, in other words, we can **rank these probabilities**

- So far, we know that β is a parameter matrix, and x represents the features. Therefore, above equation can be re-written as

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

- In above equation, we are taking sum of the **weighted features**, which is a dot product, and we are adding them to a **bias term**.

Logistic Regression

$$P(y = 1|x) \propto \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

- Above equation compares two inputs and identifies which of them leads to a higher probability of y belonging to a given class, in other words, we can **rank these probabilities**

- So far, we know that β is a parameter matrix, and x represents the features. Therefore, above equation can be re-written as

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

- In above equation, we are taking sum of the **weighted features**, which is a dot product, and we are adding them to a **bias term**.

Logistic Regression

$$P(y = 1|x) \propto \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

- Above equation compares two inputs and identifies which of them leads to a higher probability of y belonging to a given class, in other words, we can **rank these probabilities**

- So far, we know that β is a parameter matrix, and x represents the features. Therefore, above equation can be re-written as

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

Our algorithm needs to learn W and b

- In above equation, we are taking sum of the **weighted features**, which is a dot product, and we are adding them to a **bias term**.

Logistic Regression

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

- But our equation still does not provide a probability, and it looks like a linear model.

Logistic Regression

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

- But our equation still does not provide a probability, and it looks like a linear model.

- What we really want to do is take this linear model and squash its outputs into the interval (0,1)

Logistic Regression

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

- But our equation still does not provide a probability, and it looks like a linear model.
- What we really want to do is take this linear model and squash its outputs into the interval (0,1).
- This transformation can be done using a logistic function or more formally known as **sigmoid function**

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Logistic Regression

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

- But our equation still does not provide a probability, and it looks like a linear model.
- What we really want to do is take this linear model and squash its outputs into the interval (0,1) .
- This transformation can be done using a logistic function or more formally known as **sigmoid function**

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Logistic Regression

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

- But our equation still does not provide a probability, and it looks like a linear model.
- What we really want to do is take this linear model and squash its outputs into the interval (0,1).
- This transformation can be done using a logistic function or more formally known as **sigmoid function**

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

It is differentiable!

Logistic Regression

$$P(y = 1|x) = b + \boldsymbol{W}.\boldsymbol{X}$$

- But our equation still does not provide a probability, and it looks like a linear model.
- What we really want to do is take this linear model and squash its outputs into the interval (0,1).
- This transformation can be done using a logistic function or more formally known as **sigmoid function**

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

It is differentiable!

Has the property: $1 - \sigma(z) = \sigma(-z)$

Logistic Regression

$$P(y = 1 | x) = \sigma(b + \mathbf{w}.\mathbf{x})$$

- Apply this transformation to our linear model.

Logistic Regression

$$P(y = 1|\, x) \;=\; \sigma(b + \mathbf{w}.\mathbf{x})$$

- Apply this transformation to our linear model.

Probability of an input to belong to class y=1 is given by:

$$P(y = 1) \;=\; \frac{1}{1 \;+\; e^{-(b+\mathbf{w}.\mathbf{x})}}$$

Logistic Regression

$$P(y = 1|\, x) \;=\; \sigma(b + \mathbf{w}.\mathbf{x})$$

- Apply this transformation to our linear model.

Probability of an input to belong to class y=1 is given by:

$$P(y = 1) \;=\; \frac{1}{1 \;+\; e^{-(b+\mathbf{w}.\mathbf{x})}}$$

Probability of an input to belong to class y=0 is given by:

$$P(y = 0) \;=\; 1 \;-\; \sigma(b + \boldsymbol{W}.\boldsymbol{X}) = \frac{e^{-(b+\mathbf{w}.\mathbf{x})}}{1 + e^{-(b+\mathbf{w}.\mathbf{x})}}$$

Logistic Regression

$$P(y = 1|\, x) \;=\; \sigma(b + \mathbf{w}.\mathbf{x})$$

- Apply this transformation to our linear model.

Probability of an input to belong to class y=1 is given by:

$$P(y = 1) \;=\; \frac{1}{1 \;+\; e^{-(b+\mathbf{w}.\mathbf{x})}}$$

Probability of an input to belong to class y=0 is given by:

$$P(y = 0) \;=\; 1 \;-\; \sigma(b + \boldsymbol{W}.\boldsymbol{X}) = \frac{e^{-(b+\mathbf{w}.\mathbf{x})}}{1 + e^{-(b+\mathbf{w}.\mathbf{x})}}$$

Therefore, log of the odds ratio can be given by

$$log\;(\frac{p(x)}{1 - p(x)}) \;=\; \beta_0 \;+\; \sum_{i=1}^{p} \beta_i x_i$$

Logistic Regression

$$P(y = 1|\, x) \;=\; \sigma(b + \mathbf{w.x})$$

- Apply this transformation to our linear model.

Probability of an input to belong to class y=1 is given by:

$$P(y = 1) \;=\; \frac{1}{1 \;+\; e^{-(b+\mathbf{w.x})}}$$

Probability of an input to belong to class y=0 is given by:

$$P(y = 0) \;=\; 1 \;-\; \sigma(b + \boldsymbol{W}.\boldsymbol{X}) = \frac{e^{-(b+\mathbf{w.x})}}{1 + e^{-(b+\mathbf{w.x})}}$$

Therefore, log of the odds ratio can be given by,

Logit function

$$log\,(\frac{p(x)}{1 - p(x)}) \;=\; \beta_0 \;+\; \sum_{i=1}^{p} \beta_i x_i$$

Logistic regression model

Logistic Regression

$$log\ (\frac{p(x)}{1-p(x)}) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i$$

- We can predict a $y = 1$ if the probability is high, and $y = 0$ if probability is low.

Logistic Regression

$$log\ (\frac{p(x)}{1-p(x)}) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i$$

- We can predict a $y = 1$ if the probability is high, and $y = 0$ if probability is low.

How do we decide what is high and what is low ?

Logistic Regression

$$log\ (\frac{p(x)}{1-p(x)})\ =\ \beta_0\ +\ \sum_{i=1}^{p}\beta_i x_i$$

- We can predict a $y = 1$ if the probability is high, and $y = 0$ if probability is low.
- We arbitrarily we select a threshold, which is usually 0.5

Organizing data

$$log\ (\frac{p(x)}{1-p(x)})\ =\ \beta_0\ +\ \sum_{i=1}^{p}\beta_i x_i \qquad \Rightarrow\ y = b + \boldsymbol{W}.\boldsymbol{X}$$

p = number of observations
n = number of features

Organizing data

$$log\,(\frac{p(x)}{1-p(x)}) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i \quad \Rightarrow\ y = b + \boldsymbol{W}.\boldsymbol{X}$$

p = number of observations
n = number of features

$$y = \begin{vmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{vmatrix} \qquad X = \begin{vmatrix} x_{11} & x_{12}\dots & x_{1n} \\ x_{21} & x_{22}\dots & x_{2n} \\ \vdots & \vdots\ \dots & \vdots \\ x_{p1} & x_{p2}\dots & x_{pn} \end{vmatrix} \qquad W = \begin{vmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{vmatrix} \qquad b = \begin{vmatrix} b \\ b_2 \\ \vdots \\ b_p \end{vmatrix}$$

Organizing data

$$log\ (\frac{p(x)}{1-p(x)})\ =\ \beta_0\ +\ \sum_{i=1}^{p}\beta_i x_i \qquad \Rightarrow\ y = b + \boldsymbol{W}.\boldsymbol{X}$$

p = number of observations
n = number of features

$$y = \begin{vmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{vmatrix} \qquad X\ = \begin{vmatrix} x_{11} & x_{12} \dots & x_{1n} \\ x_{21} & x_{22} \dots & x_{2n} \\ \vdots & \vdots\ \dots & \vdots \\ x_{p1} & x_{p2} \dots & x_{pn} \end{vmatrix} \qquad W\ = \begin{vmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{vmatrix} \qquad b\ = \begin{vmatrix} b \\ b_2 \\ \vdots \\ b_p \end{vmatrix}$$

$p * 1$ $\qquad$ $p\ *\ n$ $\qquad$ $p\ * 1$ $\qquad$ $p\ * 1$

Organizing data

$$log\,(\frac{p(x)}{1-p(x)}) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i \quad \Rightarrow\; y = b + \boldsymbol{W}.\boldsymbol{X}$$

p = number of observations
n = number of features

$$y = \begin{vmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{vmatrix} \qquad X = \begin{vmatrix} x_{11} & x_{12}\ldots & x_{1n} \\ x_{21} & x_{22}\ldots & x_{2n} \\ \vdots & \vdots\;\;\ldots & \vdots \\ x_{p1} & x_{p2}\ldots & x_{pn} \end{vmatrix} \qquad W = \begin{vmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{vmatrix} \qquad b = \begin{vmatrix} b \\ b_2 \\ \vdots \\ b_p \end{vmatrix}$$

$$p * 1 \qquad p * n \qquad p * 1 \qquad p * 1$$

Dot multiplication of X and W ?

Organizing data

$$X = \begin{vmatrix} x_{11} & x_{12}\cdots & x_{1n} \\ x_{21} & x_{22}\cdots & x_{2n} \\ \vdots & \vdots \quad \cdots & \vdots \\ x_{p1} & x_{p2}\cdots & x_{pn} \end{vmatrix} \qquad W = \begin{vmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{vmatrix}$$

$$p * n \qquad\qquad p * 1$$

```
a = np.array([
                [1,2,3,4],
                [3,2,1,4],
                [5,4,6,7],
                [11,12,13,14]
            ])


b = np.array([[[2,3],[11,9],[32,21],[28,17]],
              [[2,3],[1,9],[3,21],[28,7]],
               [[2,3],[1,9],[3,21],[28,7]],
            ])
```

```
dot mutliplication:
 [[[232 152]
  [125 112]
  [125 112]]

 [[172 116]
  [123  76]
  [123  76]]

 [[442 296]
  [228 226]
  [228 226]]

 [[962 652]
  [465 512]
  [465 512]]]
```

```
print('dot mutliplication:\n {}\n'.format(np.dot(a,b)))
```

Dot multiplication of X and W ?

Organizing data

$$X = \begin{vmatrix} x_{11} & x_{12}\cdots & x_{1n} \\ x_{21} & x_{22}\cdots & x_{2n} \\ \vdots & \vdots \;\; \cdots & \vdots \\ x_{p1} & x_{p2}\cdots & x_{pn} \end{vmatrix} \qquad W = \begin{vmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{vmatrix}$$

$$p * n \qquad\qquad p * 1$$

```python
a = np.array([
            [1,2,3,4],
            [3,2,1,4],
            [5,4,6,7],
            [11,12,13,14]
        ])


b = np.array([[[2,3],[11,9],[32,21],[28,17]],
            [[2,3],[1,9],[3,21],[28,7]],
            [[2,3],[1,9],[3,21],[28,7]],
        ])
```

```
matrix multiplication:
 [[[232 152]
  [172 116]
  [442 296]
  [962 652]]

 [[125 112]
  [123  76]
  [228 226]
  [465 512]]

 [[125 112]
  [123  76]
  [228 226]
  [465 512]]]
```

```python
print('matrix multiplication:\n {}\n'.format(np.matmul(a,b)))
```

Matrix multiplication of X and W ?

Organizing data

$$log\left(\frac{p(x)}{1-p(x)}\right) = \beta_0 + \sum_{i=1}^{p}\beta_i x_i \quad \Rightarrow \; y = b + \boldsymbol{W}.\boldsymbol{X}$$

p = number of observations
n = number of features

$$y = \begin{vmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{vmatrix} \qquad X = \begin{vmatrix} x_{11} & x_{12}\dots & x_{1n} \\ x_{21} & x_{22}\dots & x_{2n} \\ \vdots & \vdots \;\; \dots & \vdots \\ x_{p1} & x_{p2}\dots & x_{pn} \end{vmatrix} \qquad W = \begin{vmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{vmatrix} \qquad b = \begin{vmatrix} b \\ b_2 \\ \vdots \\ b_p \end{vmatrix}$$

$p * 1$ $\qquad$ $p * n$ $\qquad$ $p * 1$ $\qquad$ $p * 1$

Dot multiplication of X and W ?

Matrix multiplication of X and W ?

Organizing data

$$log\,(\frac{p(x)}{1-p(x)}) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i \quad \Rightarrow \; y = b + \boldsymbol{W}.\boldsymbol{X}$$

p = number of observations
n = number of features

$$y = \begin{vmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{vmatrix} \qquad X = \begin{vmatrix} x_{11} & x_{12}\dots & x_{1n} \\ x_{21} & x_{22}\dots & x_{2n} \\ \vdots & \vdots\;\dots & \vdots \\ x_{p1} & x_{p2}\dots & x_{pn} \end{vmatrix} \qquad W = \begin{vmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{vmatrix} \qquad b = \begin{vmatrix} b \\ b_2 \\ \vdots \\ b_p \end{vmatrix}$$

$p * 1$ $\qquad$ $p * n$ $\qquad$ $p * 1$ $\qquad$ $p * 1$

$y = X.dot(W) + b$ To align our vectors for multiplication

Organizing data

$$log\ (\frac{p(x)}{1-p(x)}) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i \quad \Rightarrow\ y = b + \boldsymbol{W}.\boldsymbol{X}$$

p = number of observations
n = number of features

$$y = \begin{vmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{vmatrix} \qquad X = \begin{vmatrix} x_{11} & x_{12}\dots & x_{1n} \\ x_{21} & x_{22}\dots & x_{2n} \\ \vdots & \vdots\ \dots & \vdots \\ x_{p1} & x_{p2}\dots & x_{pn} \end{vmatrix} \qquad W = \begin{vmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{vmatrix} \qquad b = \begin{vmatrix} b \\ b_2 \\ \vdots \\ b_p \end{vmatrix}$$

$p * 1$ $\qquad$ $p * n$ $\qquad$ $p * 1$ $\qquad$ $p * 1$

Try this at home

```
tensor1 = torch.tensor([5, 8, 2])
tensor2 = torch.tensor([4, 7, 2])

tensor1 @ tensor2
torch.dot(tensor1, tensor2)
torch.matmul(tensor1, tensor2)
```

McMaster University

Logistic Regression

$$log\,(\frac{p(x)}{1-p(x)}) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i$$

- We can predict a $y = 1$ if the probability is high, and $y = 0$ if probability is low.
- We arbitrarily we select a threshold, which is usually 0.5.
- How do we train Logistic Regression?

McMaster University

Logistic Regression

$$log\ (\frac{p(x)}{1-p(x)}) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i$$

- We can predict a $y = 1$ if the probability is high, and $y = 0$ if probability is low.
- We arbitrarily we select a threshold, which is usually 0.5.
- How do we train Logistic Regression?

Select a cost function:

The likelihood function is the conditional probability of the data conditional on the given set of parameters. If x_i and y_i are two given points, then

$$P(\, x_i, y_i \mid \beta) = \begin{cases} p(x_i) & , if\ y_i = 1 \\ 1 - p(x_i) & , if\ y_i = 0 \end{cases}$$

McMaster University

Logistic Regression

$$P(\, x_i, y_i \mid \beta) \; = \; \begin{cases} p(x_i) & , if \; y_i \; = \; 1 \\ 1 - p(x_i) & , if \; y_i \; = \; 0 \end{cases}$$

We can take the above equation, generalize it across N data points, apply log on both sides and get the complete **log-likelihood** function over the entire dataset as

$$\ell(\beta) = \sum_{i=1}^{N} log\big(P(\,, x_i, \; y_i \;\; \beta)\big) \;\; = \; \sum_{i=1}^{N} \{y_i log\big(p(x_i)\big) \; + \; (1 - y_i) \; log(\, 1 - p(x_i)\,)\}$$

Logistic Regression

$$P(\,x_i, y_i \mid \beta) \;=\; \begin{cases} p(x_i) & , if\ y_i \;=\; 1 \\ 1 - p(x_i) & , if\ y_i \;=\; 0 \end{cases}$$

We can take the above equation, generalize it across N data points, apply log on both sides and get the complete **log-likelihood** function over the entire dataset as

$$\ell(\beta) = \sum_{i=1}^{N} log\left(P(\,, x_i, \; y_i \;\; \beta)\right) \;\; = \underbrace{\sum_{i=1}^{N} \left\{ y_i log\left(p(x_i)\right) \; + \; (1 - y_i)\; log(\,1 - p(x_i)\,) \right\}}_{\text{Likelihood estimator / Cost function}}$$

Challenger Dataset

Date	Launch Temp (F)	Leak Check Pressure	Thermal distress	Number of O-rings
4/12/81	66	50	0	6
11/12/81	70	50	1	6
3/22/82	69	50	0	6
11/11/82	68	50	0	6
4/4/83	67	50	0	6
6/18/83	72	50	0	6
8/30/83	73	50	0	6
11/28/83	70	100	0	6
2/3/84	57	100	1	6
4/6/84	63	200	1	6
8/30/84	70	200	1	6
10/5/84	78	200	0	6

Date	Launch Temp (F)	Leak Check Pressure	Thermal distress	Number of O-rings
11/8/84	67	200	0	6
1/24/85	53	200	2	6
4/12/85	67	200	0	6
4/29/85	75	200	0	6
6/17/85	70	200	0	6
7/29/85	81	200	0	6
8/27/85	76	200	0	6
10/3/85	79	200	0	6
10/30/85	75	200	2	6
11/26/85	76	200	0	6
1/12/86	58	200	1	6

Challenger Dataset

0 = failure of event   1 = success of event

Thermal distress	Launch Temp (F)	Did 0-ring get damaged
0	66	0
1	70	1
0	69	0
0	68	0
0	67	0
0	72	0
0	73	0
0	70	0
1	57	1
1	63	1
1	70	1
0	78	0

Thermal distress	Launch Temp (F)	Did 0-ring get damaged
0	67	0
2	53	1
0	67	0
0	75	0
0	70	0
0	81	0
0	76	0
0	79	0
2	75	1
0	76	0
1	58	1

Challenger Dataset

0 = failure of event 1 = success of event

Thermal distress	Launch Temp (F)	Did O-ring get damaged
0	66	0
1	70	1
0	69	0
0	68	0
0	67	0
0	72	0
0	73	0
0	70	0
1	57	1
1	63	1
1	70	1
0	78	0

Thermal distress	Launch Temp (F)	Did O-ring get damaged
0	67	0
2	53	1
0	67	0
0	75	0
0	70	0
0	81	0
0	76	0
0	79	0
2	75	1
0	76	0
1	58	1

$$p(x) = b + \boldsymbol{W}.\boldsymbol{X}$$

Challenger Dataset

0 = failure of event 1 = success of event

Thermal distress	Launch Temp (F)	Did O-ring get damaged
0	66	0
1	70	1
0	69	0
0	68	0
0	67	0
0	72	0
0	73	0
0	70	0
1	57	1
1	63	1
1	70	1
0	78	0

Thermal distress	Launch Temp (F)	Did O-ring get damaged
0	67	0
2	53	1
0	67	0
0	75	0
0	70	0
0	81	0
0	76	0
0	79	0
2	75	1
0	76	0
1	58	1

$$p(x) = b + \boldsymbol{W}.\boldsymbol{X}$$

$$b = 10.875 \quad W = -0.171$$

Challenger Dataset

0 = failure of event   1 = success of event

Thermal distress	Launch Temp (F)	Did 0-ring get damaged
0	66	0
1	70	1
0	69	0
0	68	0
0	67	0
0	72	0
0	73	0
0	70	0
1	57	1
1	63	1
1	70	1
0	78	0

Thermal distress	Launch Temp (F)	Did 0-ring get damaged
0	67	0
2	53	1
0	67	0
0	75	0
0	70	0
0	81	0
0	76	0
0	79	0
2	75	1
0	76	0
1	58	1

$$p(x) = b + \boldsymbol{W}.\boldsymbol{X}$$

$$b = 10.875 \quad W = -0.171$$

$$log\left(\frac{p(x)}{1 - p(x)}\right) = 10.875 - 0.171x$$

McMaster University

Challenger Dataset

0 = failure of event   1 = success of event

Thermal distress	Launch Temp (F)	Did O-ring get damaged
0	66	0
1	70	1
0	69	0
0	68	0
0	67	0
0	72	0
0	73	0
0	70	0
1	57	1
1	63	1
1	70	1
0	78	0

Thermal distress	Launch Temp (F)	Did O-ring get damaged
0	67	0
2	53	1
0	67	0
0	75	0
0	70	0
0	81	0
0	76	0
0	79	0
2	75	1
0	76	0
1	58	1

$$p(x) = b + \boldsymbol{W}.\boldsymbol{X}$$

$$b = 10.875 \quad W = -0.171$$

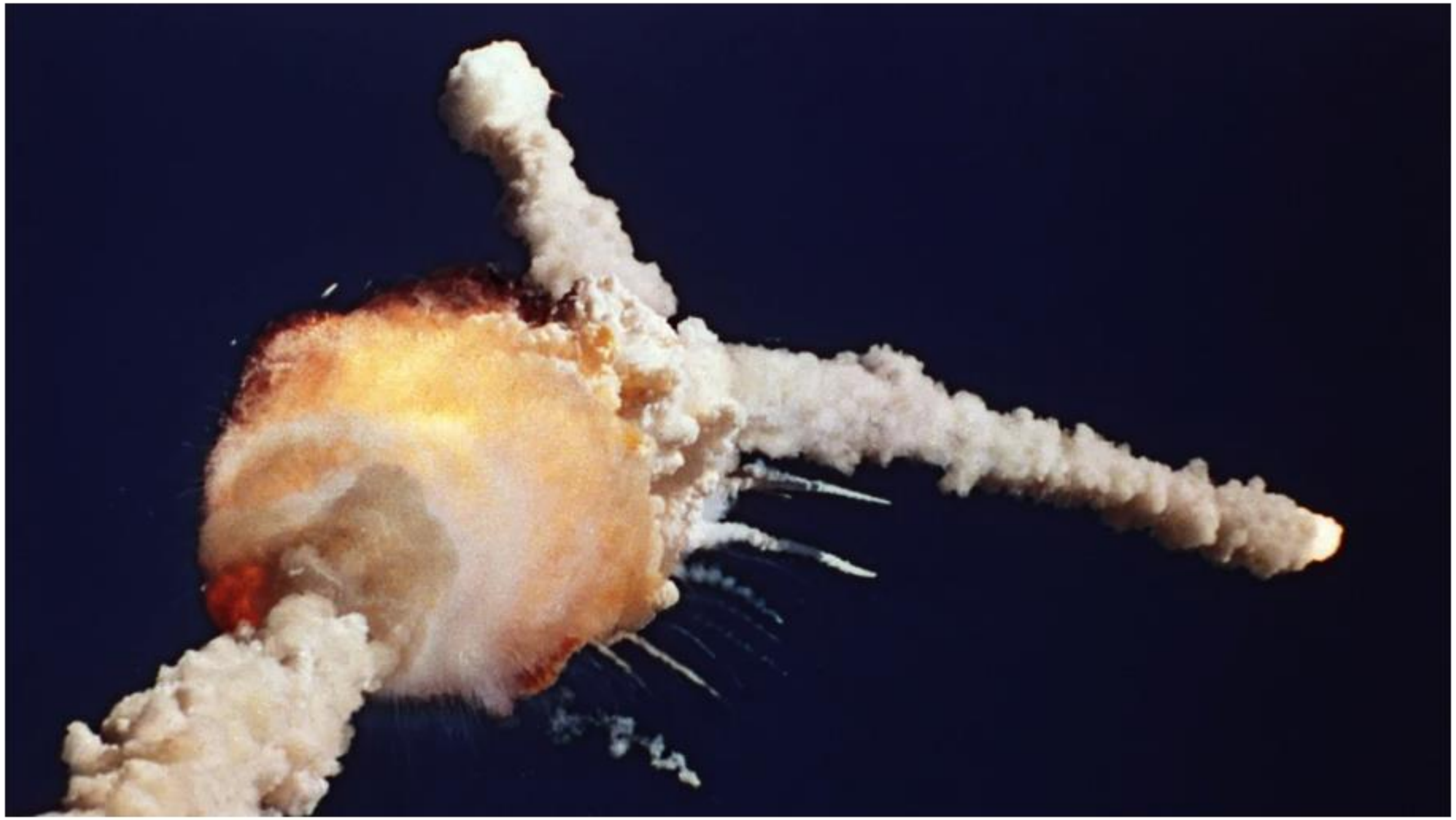
$$log\left(\frac{p(x)}{1-p(x)}\right) = 10.875 - 0.171x$$

On that day it was $31°\mathrm{F}$

$$P(y=1) = \frac{1}{1+e^{-(10.875+\mathbf{0.171x31})}} = 0.996$$

McMaster University

Challenger Dataset



96% probability that at least 1 O-ring would fail on that day!

Next Lecture

- How do we build and train a logistic regression model?
- No readings for this lecture.

Thank You

McMaster University