

At the end of the class you should be able to:

- identify a supervised learning problem
- characterize how the prediction is a function of the error measure
- avoid mixing the training and test sets

Given:

- a set of **inputs features** X_1, \dots, X_n
- a set of **target features** Y_1, \dots, Y_k
- a set of **training examples** where the values for the input features and the target features are given for each example
- a new example, where only the values for the input features are given

predict the values for the target features for the new example.

Given:

- a set of **inputs features** X_1, \dots, X_n
- a set of **target features** Y_1, \dots, Y_k
- a set of **training examples** where the values for the input features and the target features are given for each example
- a new example, where only the values for the input features are given

predict the values for the target features for the new example.

- **classification** when the Y_i are discrete
- **regression** when the Y_i are continuous

Example Data Representations

A travel agent wants to predict the preferred length of a trip, which can be from 1 to 6 days. (No input features).

Example Data Representations

A travel agent wants to predict the preferred length of a trip, which can be from 1 to 6 days. (No input features).

Two representations of the same data:

- Y is the length of trip chosen.
- Each Y_i is an **indicator variable** that has value 1 if the chosen length is i , and is 0 otherwise.

Example	Y	Example	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6
e_1	1	e_1	1	0	0	0	0	0
e_2	6	e_2	0	0	0	0	0	1
e_3	6	e_3	0	0	0	0	0	1
e_4	2	e_4	0	1	0	0	0	0
e_5	1	e_5	1	0	0	0	0	0

What is a prediction?

Evaluating Predictions

Suppose we want to make a prediction of a value for a target feature on example e :

- o_e is the observed value of target feature on example e .

Evaluating Predictions

Suppose we want to make a prediction of a value for a target feature on example e :

- o_e is the observed value of target feature on example e .
- p_e is the predicted value of target feature on example e .

Evaluating Predictions

Suppose we want to make a prediction of a value for a target feature on example e :

- o_e is the observed value of target feature on example e .
- p_e is the predicted value of target feature on example e .
- The **error** of the prediction is a measure of how close p_e is to o_e .

Evaluating Predictions

Suppose we want to make a prediction of a value for a target feature on example e :

- o_e is the observed value of target feature on example e .
- p_e is the predicted value of target feature on example e .
- The **error** of the prediction is a measure of how close p_e is to o_e .
- There are many possible errors that could be measured.

Sometimes p_e can be a real number even though o_e can only have a few values.

E is a sequence of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- **absolute error** $L_1(E) = \sum_{e \in E} |o_e - p_e|$

E is a sequence of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- **absolute error** $L_1(E) = \sum_{e \in E} |o_e - p_e|$
- **sum of squares error** $L_2^2(E) = \sum_{e \in E} (o_e - p_e)^2$

E is a sequence of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- **absolute error** $L_1(E) = \sum_{e \in E} |o_e - p_e|$
- **sum of squares error** $L_2^2(E) = \sum_{e \in E} (o_e - p_e)^2$
- **worst-case error**: $L_\infty(E) = \max_{e \in E} |o_e - p_e|$

E is a sequence of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- **absolute error** $L_1(E) = \sum_{e \in E} |o_e - p_e|$
- **sum of squares error** $L_2^2(E) = \sum_{e \in E} (o_e - p_e)^2$
- **worst-case error**: $L_\infty(E) = \max_{e \in E} |o_e - p_e|$
- **number wrong**: $L_0(E) = \#\{e : o_e \neq p_e\}$

E is a sequence of examples, with single target feature. For $e \in E$, o_e is observed value and p_e is predicted value:

- **absolute error** $L_1(E) = \sum_{e \in E} |o_e - p_e|$
- **sum of squares error** $L_2^2(E) = \sum_{e \in E} (o_e - p_e)^2$
- **worst-case error**: $L_\infty(E) = \max_{e \in E} |o_e - p_e|$
- **number wrong**: $L_0(E) = \#\{e : o_e \neq p_e\}$
- A **cost-based error** takes into account costs of errors.

Measures of error (cont.)

With binary feature: $o_e \in \{0, 1\}$:

- likelihood of the data

$$\prod_{e \in E} p_e^{o_e} (1 - p_e)^{(1 - o_e)}$$

Measures of error (cont.)

With binary feature: $o_e \in \{0, 1\}$:

- **likelihood of the data**

$$\prod_{e \in E} p_e^{o_e} (1 - p_e)^{(1 - o_e)}$$

- **log likelihood**

$$\sum_{e \in E} (o_e \log p_e + (1 - o_e) \log(1 - p_e))$$

in terms of information:

With binary feature: $o_e \in \{0, 1\}$:

- **likelihood of the data**

$$\prod_{e \in E} p_e^{o_e} (1 - p_e)^{(1 - o_e)}$$

- **log likelihood**

$$\sum_{e \in E} (o_e \log p_e + (1 - o_e) \log(1 - p_e))$$

in terms of information:

It is negative of number of bits to encode the data given a code based on p_e .

Information theory overview

- A **bit** is a binary digit.
- 1 bit can distinguish

Information theory overview

- A **bit** is a binary digit.
- 1 bit can distinguish 2 items
- k bits can distinguish

Information theory overview

- A **bit** is a binary digit.
- 1 bit can distinguish 2 items
- k bits can distinguish 2^k items
- n items can be distinguished using

Information theory overview

- A **bit** is a binary digit.
- 1 bit can distinguish 2 items
- k bits can distinguish 2^k items
- n items can be distinguished using $\log_2 n$ bits

- A **bit** is a binary digit.
- 1 bit can distinguish 2 items
- k bits can distinguish 2^k items
- n items can be distinguished using $\log_2 n$ bits
- Can we do better?

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a	0	b	10	c	110	d	111
-----	---	-----	----	-----	-----	-----	-----

This code uses bits.

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a	0	b	10	c	110	d	111
-----	---	-----	----	-----	-----	-----	-----

This code uses 1 to 3 bits. On average, it uses

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a 0 b 10 c 110 d 111

This code uses 1 to 3 bits. On average, it uses

$$\begin{aligned} &P(a) \times 1 + P(b) \times 2 + P(c) \times 3 + P(d) \times 3 \\ &= \frac{1}{2} + \frac{2}{4} + \frac{3}{8} + \frac{3}{8} = 1\frac{3}{4} \text{ bits.} \end{aligned}$$

The string *aacabbda* has code

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a 0 b 10 c 110 d 111

This code uses 1 to 3 bits. On average, it uses

$$\begin{aligned} &P(a) \times 1 + P(b) \times 2 + P(c) \times 3 + P(d) \times 3 \\ &= \frac{1}{2} + \frac{2}{4} + \frac{3}{8} + \frac{3}{8} = 1\frac{3}{4} \text{ bits.} \end{aligned}$$

The string *aacabbda* has code 00110010101110.

The code 0111110010100 represents string

Consider a code to distinguish elements of $\{a, b, c, d\}$ with

$$P(a) = \frac{1}{2}, P(b) = \frac{1}{4}, P(c) = \frac{1}{8}, P(d) = \frac{1}{8}$$

Consider the code:

a 0 b 10 c 110 d 111

This code uses 1 to 3 bits. On average, it uses

$$\begin{aligned} &P(a) \times 1 + P(b) \times 2 + P(c) \times 3 + P(d) \times 3 \\ &= \frac{1}{2} + \frac{2}{4} + \frac{3}{8} + \frac{3}{8} = 1\frac{3}{4} \text{ bits.} \end{aligned}$$

The string *aacabbda* has code 00110010101110.

The code 0111110010100 represents string *adcabba*

- To identify x , we need $-\log_2 P(x)$ bits.
- Give a distribution over a set, to identify a member, the expected number of bits

$$\sum_x -P(x) \times \log_2 P(x).$$

is the **information content** or **entropy** of the distribution.

- The expected number of bits it takes to describe a distribution given evidence e :

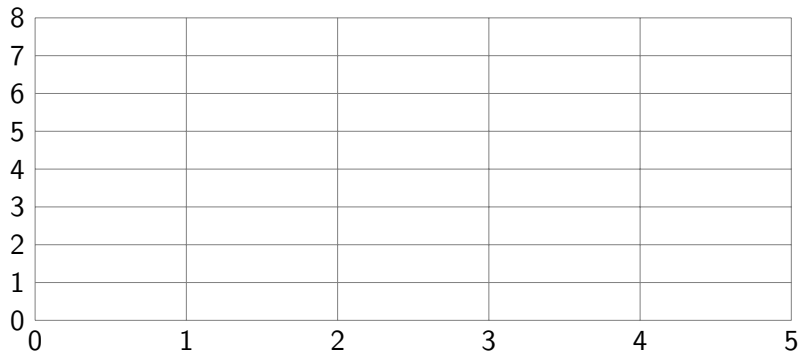
$$I(e) = \sum_x -P(x|e) \times \log_2 P(x|e).$$

Given a test that can distinguish the cases where α is true from the cases where α is false, the **information gain** from this test is:

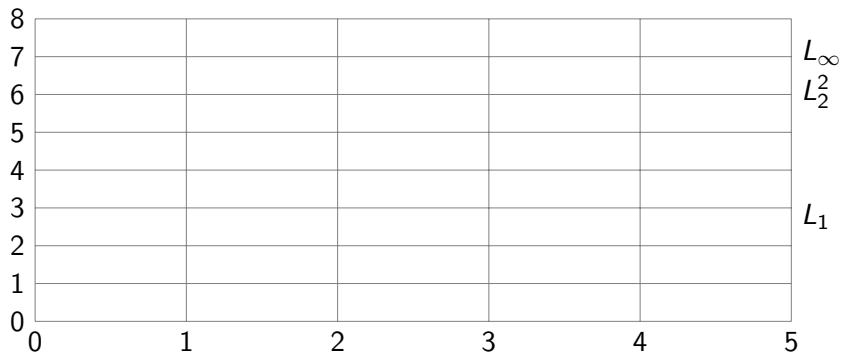
$$I(\text{true}) - (P(\alpha) \times I(\alpha) + P(\neg\alpha) \times I(\neg\alpha)).$$

- $I(\text{true})$ is the expected number of bits needed before the test
- $P(\alpha) \times I(\alpha) + P(\neg\alpha) \times I(\neg\alpha)$ is the expected number of bits after the test.

Linear Predictions



Linear Predictions



Point Estimates

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is

Point Estimates

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .

Point Estimates

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is

Point Estimates

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .

Point Estimates

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .
- The prediction that minimizes the number wrong on E is

Point Estimates

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .
- The prediction that minimizes the number wrong on E is the mode of Y .

Point Estimates

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .
- The prediction that minimizes the number wrong on E is the mode of Y .
- The prediction that minimizes the worst-case error on E is

Point Estimates

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .
- The prediction that minimizes the number wrong on E is the mode of Y .
- The prediction that minimizes the worst-case error on E is $(\textit{maximum} + \textit{minimum})/2$

Point Estimates

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .
- The prediction that minimizes the number wrong on E is the mode of Y .
- The prediction that minimizes the worst-case error on E is $(\textit{maximum} + \textit{minimum})/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .
- The prediction that minimizes the number wrong on E is the mode of Y .
- The prediction that minimizes the worst-case error on E is $(\textit{maximum} + \textit{minimum})/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is the empirical frequency.

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .
- The prediction that minimizes the number wrong on E is the mode of Y .
- The prediction that minimizes the worst-case error on E is $(\textit{maximum} + \textit{minimum})/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is the empirical frequency.
- When Y has values $\{0, 1\}$, the prediction that minimizes the entropy on E is

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .
- The prediction that minimizes the number wrong on E is the mode of Y .
- The prediction that minimizes the worst-case error on E is $(\textit{maximum} + \textit{minimum})/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is the empirical frequency.
- When Y has values $\{0, 1\}$, the prediction that minimizes the entropy on E is the empirical frequency.

Predict single value for numerical feature Y on examples E .

- The prediction that minimizes the sum of squares error on E is the mean (average) value of Y .
- The prediction that minimizes the absolute error on E is the median value of Y .
- The prediction that minimizes the number wrong on E is the mode of Y .
- The prediction that minimizes the worst-case error on E is $(\textit{maximum} + \textit{minimum})/2$
- When Y has values $\{0, 1\}$, the prediction that maximizes the likelihood on E is the empirical frequency.
- When Y has values $\{0, 1\}$, the prediction that minimizes the entropy on E is the empirical frequency.

But that doesn't mean that these predictions minimize the error for future predictions....

Training and Test Sets

To evaluate how well a learner will work on future predictions, we divide the examples into:

- **training examples** that are used to train the learner
- **test examples** that are used to evaluate the learner

...these must be kept separate.