Digital World (2019)

Week 10, S2: Classification Preliminaries

Chris Poskitt

SINGAPORE UNIVERSITY OF
TECHNOLOGY AND DESIGN

Refresher: slicing NumPy arrays

b.socrative.com, POSKITT5665

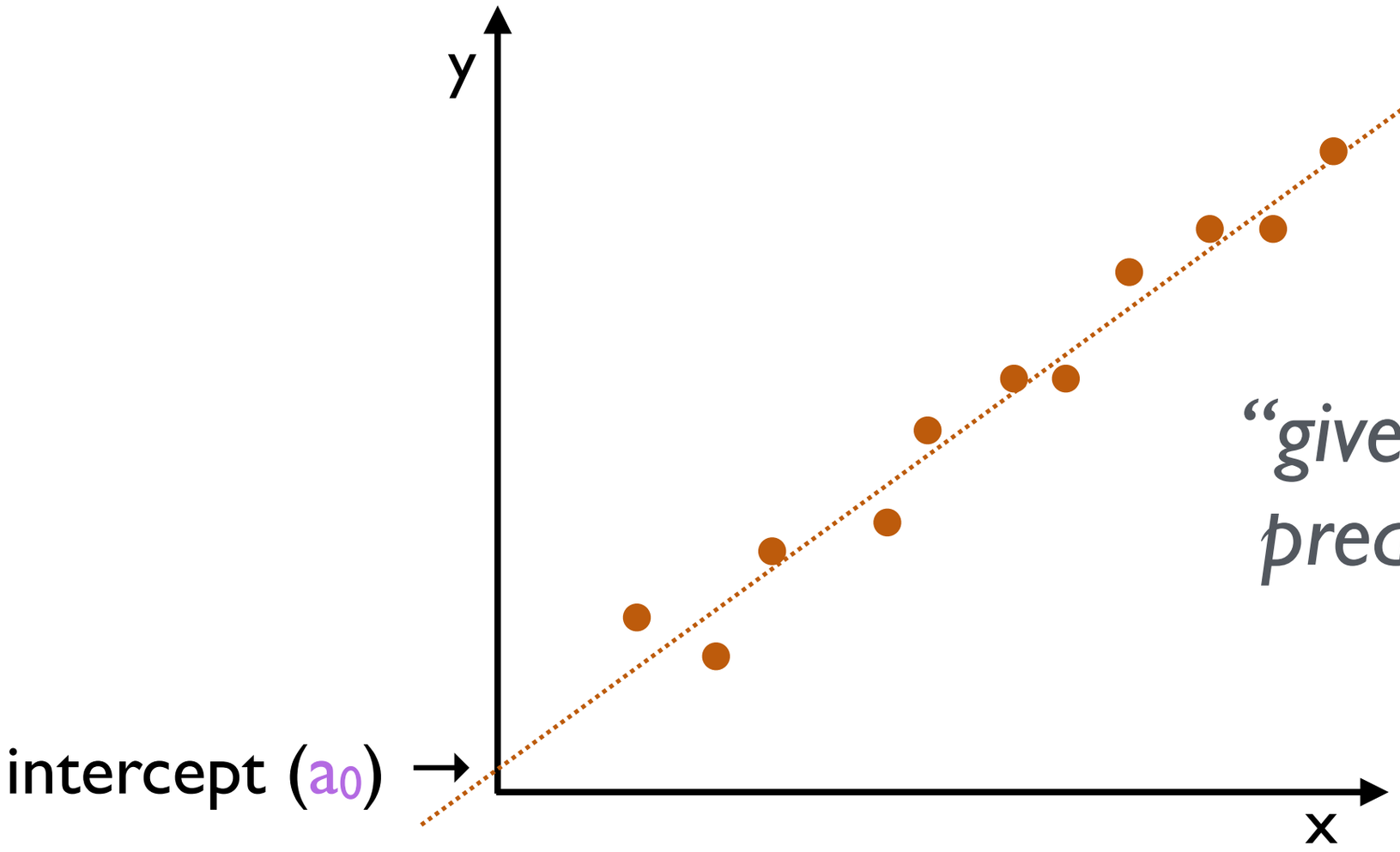
```
col_index = [2]
col_values = bunchobject.data[:, col_index]
idx_max = np.argmax(col_values)
idx_min = np.argmin(col_values)
```

Which of the following will return a **2D array** with the two rows that contain the **max** and **min** values **of** column `col_index`? The two rows should include the data from **all columns**, not just that of column `col_index`.

```
(a) data = bunchobject.data[idx_max, idx_min, :]

(b) data = bunchobject.data[[idx_max, idx_min], :]

(c) data = bunchobject.data[:, (idx_max, idx_min)]

(d) data = bunchobject.data[:, idx_max, idx_min]
```

Classification

Last time we saw a *regression* model



"given a new x value predict the y value"

$$y = a_0 + a_1 x$$

What then is a *classification* model?

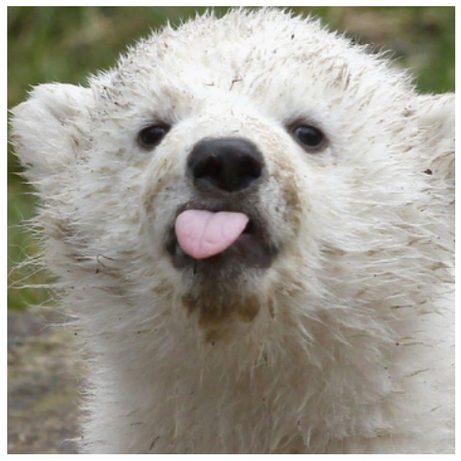
- given some input, predict the category of the output

input




output local Western Western local

input

output teddy real bear teddy teddy

What then is a *classification* model?

- categories of output are also known as targets (or labels)

- we will only consider binary classification

  => *every photo is either a teddy or a real bear*
  => *every breast mass is either benign or malignant*

- the labels of breast cancer patients are in the bunch object

```
bunchobject.target_names
bunchobject.target[23]
```

- several algorithms for training classification models

  => *k-Nearest Neighbours; Support Vector Machines; neural networks; …*

k-NN Preliminaries

Confusion matrix

- a confusion matrix allows us to visualise the performance of a classification algorithm *(e.g. k-NN)*

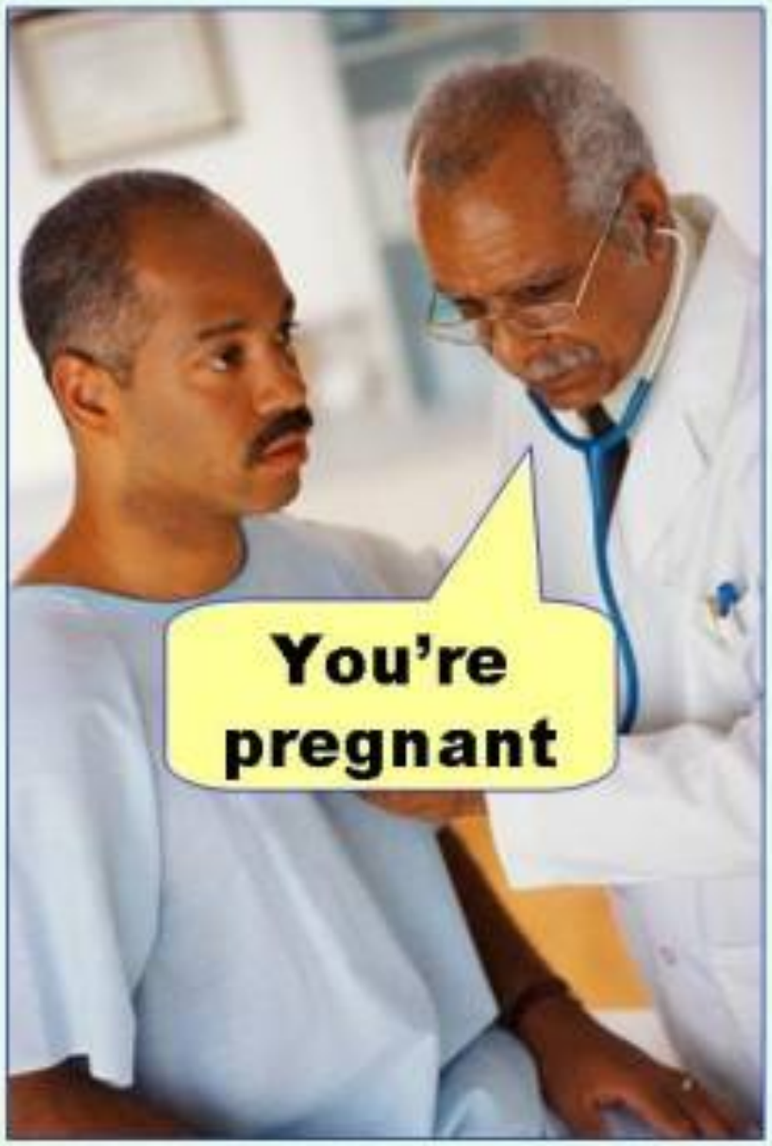
- suppose our algorithm classifies patients as pregnant or not

```
actual = ['pregnant', 'pregnant', 'pregnant', 'not', 'not', 'not']
pred = ['pregnant', 'pregnant', 'not', 'not', 'pregnant', 'pregnant']
```

	predicted not pregnant	*predicted pregnant*
actually not pregnant	1	2
actually pregnant	1	2

Type I error

(false positive)



Type II error

(false negative)

Metrics for classification algorithms

- we designated being pregnant as the 'positive case'

  => *why? because it's what we really want to predict*

- the confusion matrix gives us classification metrics with respect to the positive case

	predicted not pregnant	*predicted pregnant*
actually not pregnant	true negatives	false positives
actually pregnant	false negatives	true positives

use to determine: ***accuracy, sensitivity, false positive rate***

extract metrics from a confusion matrix

normalise data (to between 0 and 1)

Today: questions CS1, CS2, and CS3 only

five number summary of some data

Summary

- regression models predict values, whereas classification models predict categories of output (aka targets / labels)

- extract classification metrics from a confusion matrix

- many classification algorithms (e.g. k-NN) require training data to be normalised first

- homework: finish CS1, CS2, CS3