Python programming and machine learning

ILCB Summer School 2023 - Class 1/4

Thomas Schatz, August 28th 2023

Course material will be available after the class at https://thomas.schatz.cogserver.net/teaching/

Tentative outline

Class 1

- What is machine learning and why should you care ?
- Case study introduction: classification of cat and dog images

Class 2

- Programming basics (in python)

Class 3

- Implementing classification of cat and dog images (with linear and nearest neighbor classification algorithms)

Class 4

- Testing our implementation
- General discussion

Class 1

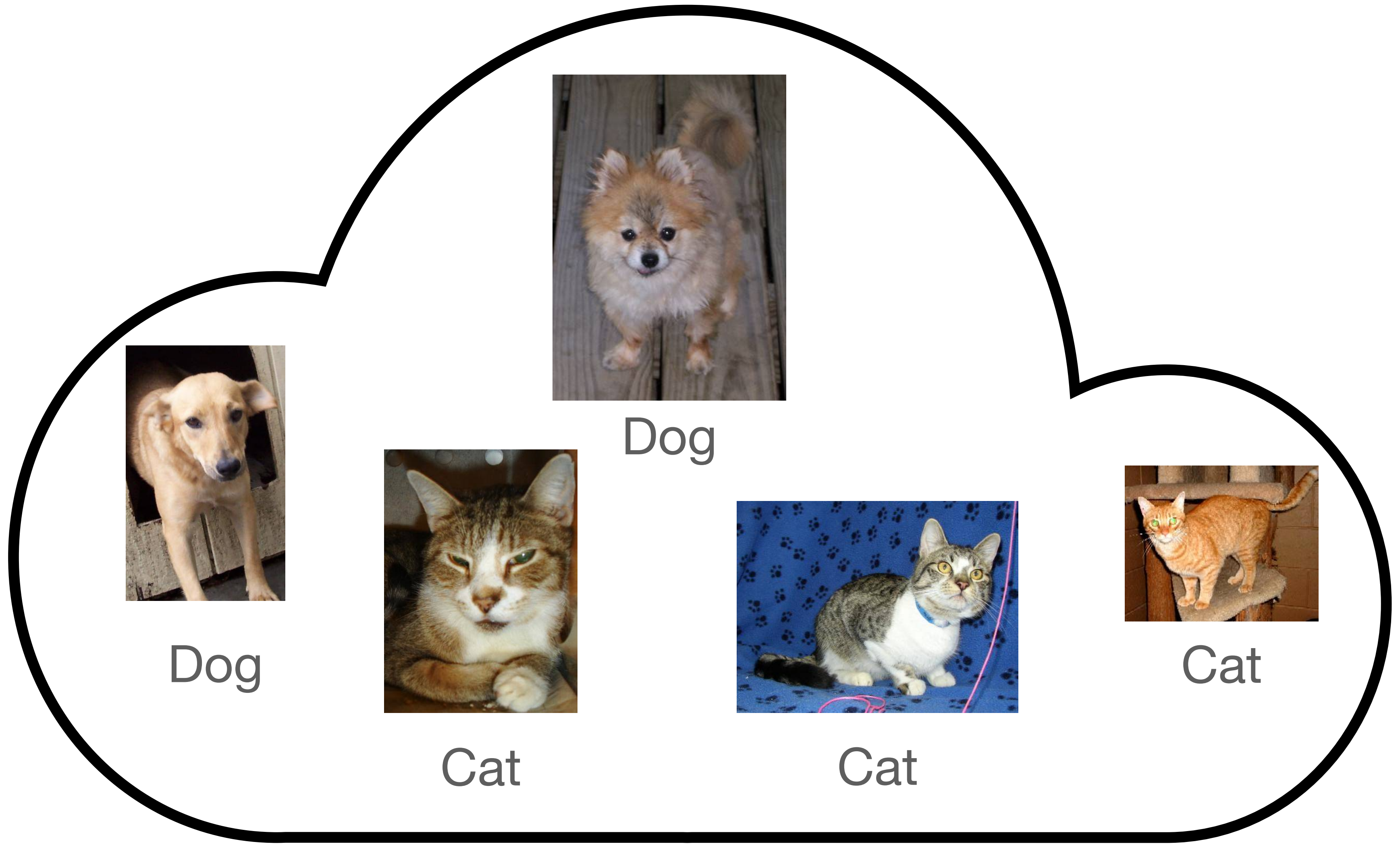
- What is machine learning and why should you care ?
- Case study introduction: classification of cat and dog images

Why should you care about Machine Learning?

Examples from cognitive (neuro)science

Developing models of neural/cognitive processes

-> “Deep learning” systems trained to classify object labels



Classifier

Training

Classifier

predicted_label (‘cat’ or ‘dog’)

Test

Example 1

(Images from kaggle’s dogs vs cats competition)

Why should you care about Machine Learning?

Examples from cognitive (neuro)science

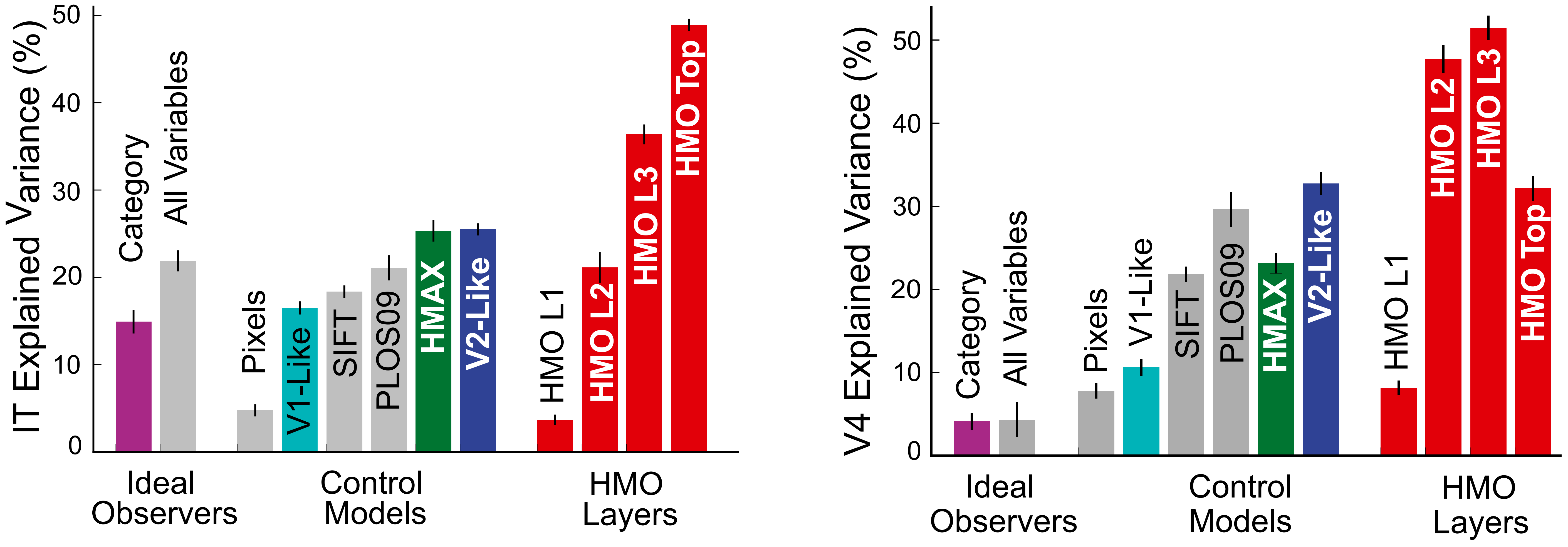
Developing models of neural/cognitive processes

-> “Deep learning” systems trained to classify object labels

Model	Top-1	Top-5
Sparse coding [2]	*47.1%*	*28.2%*
SIFT + FVs [24]	*45.7%*	*25.7%*
CNN	**37.5%**	**17.0%**

Table 1: Comparison of results on ILSVRC-2010 test set. In *italics* are best results achieved by others.

Krizhevsky, Sutskever & Hinton, NeurIPS (2012)



Yamins et al. PNAS (2014)

Example 1

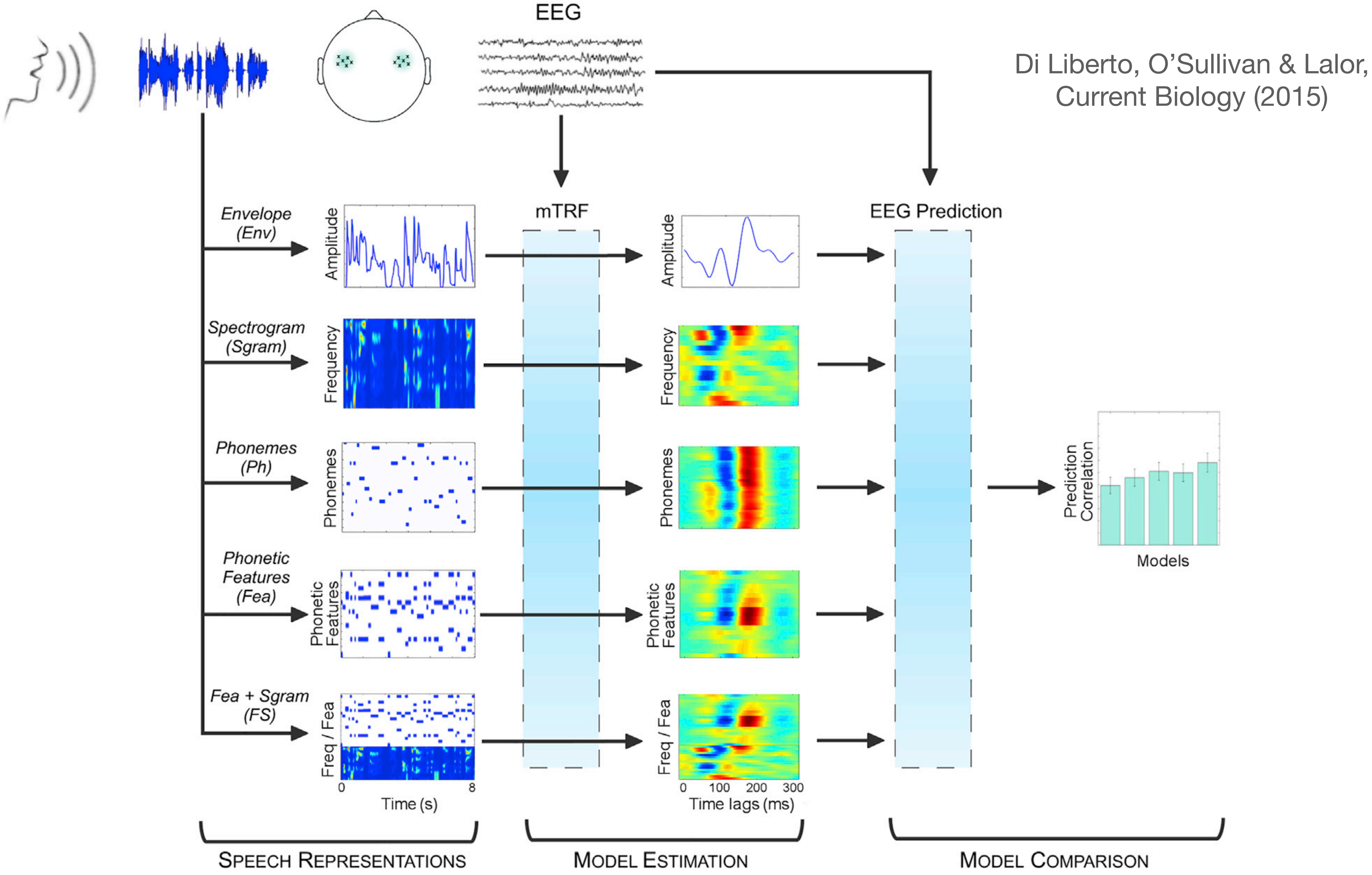
Why should you care about Machine Learning?

Examples from cognitive (neuro)science

Interpreting recordings of brain activity

-> “Encoding models” trained to predict brain activity from hypothesised cognitive representations

Example 2

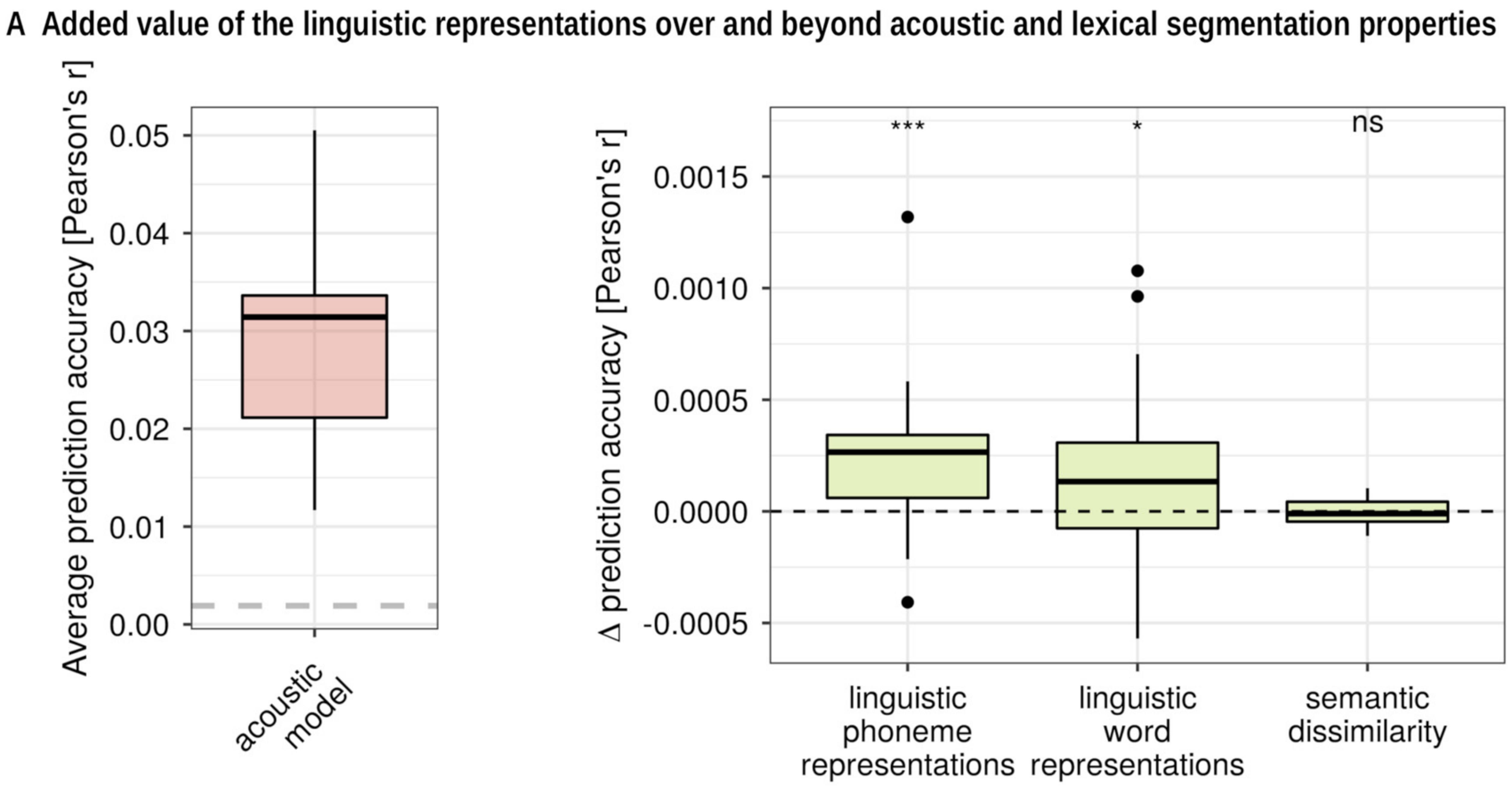

EEG
Di Liberto, O'Sullivan & Lalor,
Current Biology (2015)
Envelope (Env)
Amplitude
Spectrogram (Sgram)
Frequency
Phonemes (Ph)
Phonemes
Phonetic Features (Fea)
Phonetic Features
Fea + Sgram (FS)
Freq / Fea
0
8
Time (s)
mTRF
Amplitude
Frequency
Phonemes
Phonetic Features
Freq / Fea
0 100 200 300
Time lags (ms)
EEG Prediction
Prediction Correlation
Models
Speech Representations
Model Estimation
Model Comparison

Why should you care about Machine Learning?

Examples from cognitive (neuro)science

Interpreting recordings of brain activity

-> "Encoding models" trained to predict brain activity from hypothesised cognitive representations



Gillis et al., Journal of Neuroscience (2021)

Example 2

Why should you care about Machine Learning?

Examples from cognitive (neuro)science

Generating experimental stimuli

-> 3D Face synthesis system

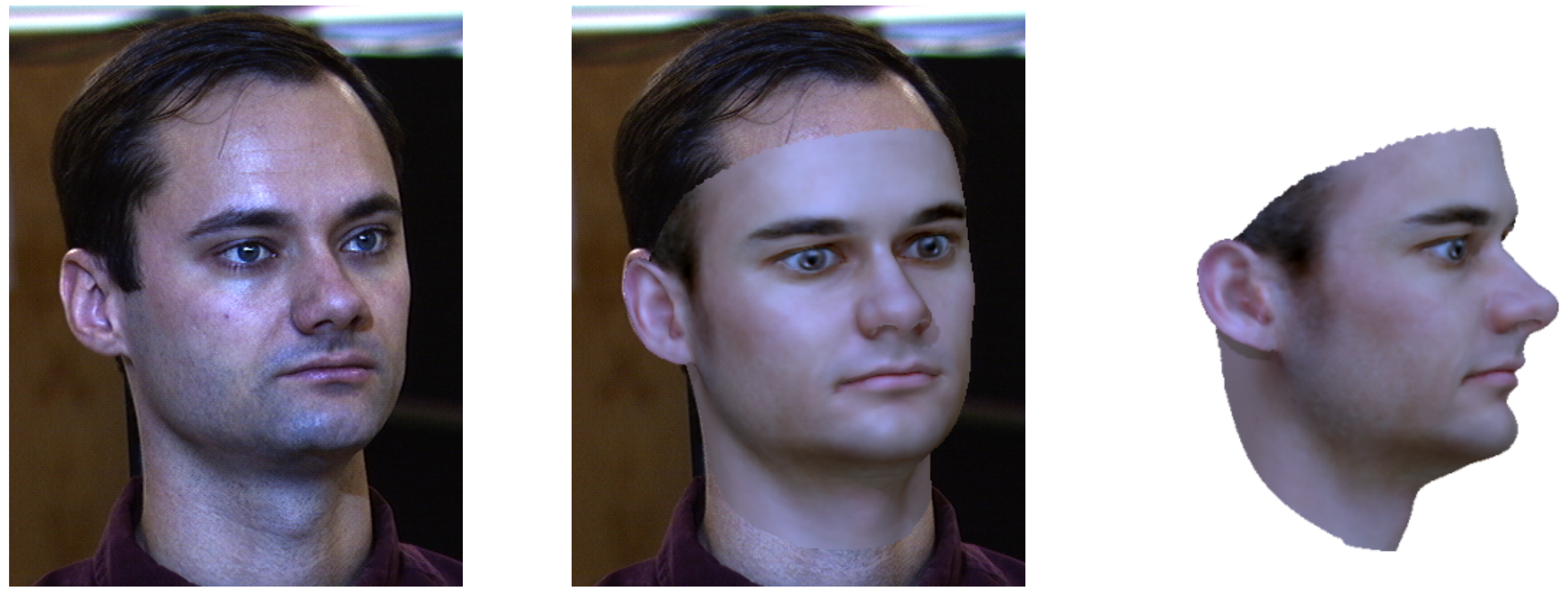


Figure 6. Exemplary fitting result for CMU-PIE with BFM Face Model. Left the original image, middle row the fitting result rendered into the image and right the resulting 3D model.

Paysan et al. (2009)

Example 3

Why should you care about Machine Learning?

Summary

(At least) three broad use cases for ML in cognitive (neuro)science

- Automatisation of time-consuming tasks (annotation, stimuli preparation…)
- Analysis of experimental data (brain imaging, behavior in the lab, online experiments…)
- Development of models of cognitive and neural processes (perception, language, decision-making, navigation, memory…)

What is Machine Learning?

What do the examples have in common?

- About generalisation, i.e. learning from experience/examples
    - That’s **statistics**
- What distinguishes machine learning within statistics?
    - **Computational aspect**: finding (and applying) ML solutions to problems requires a computer

What is Machine Learning?

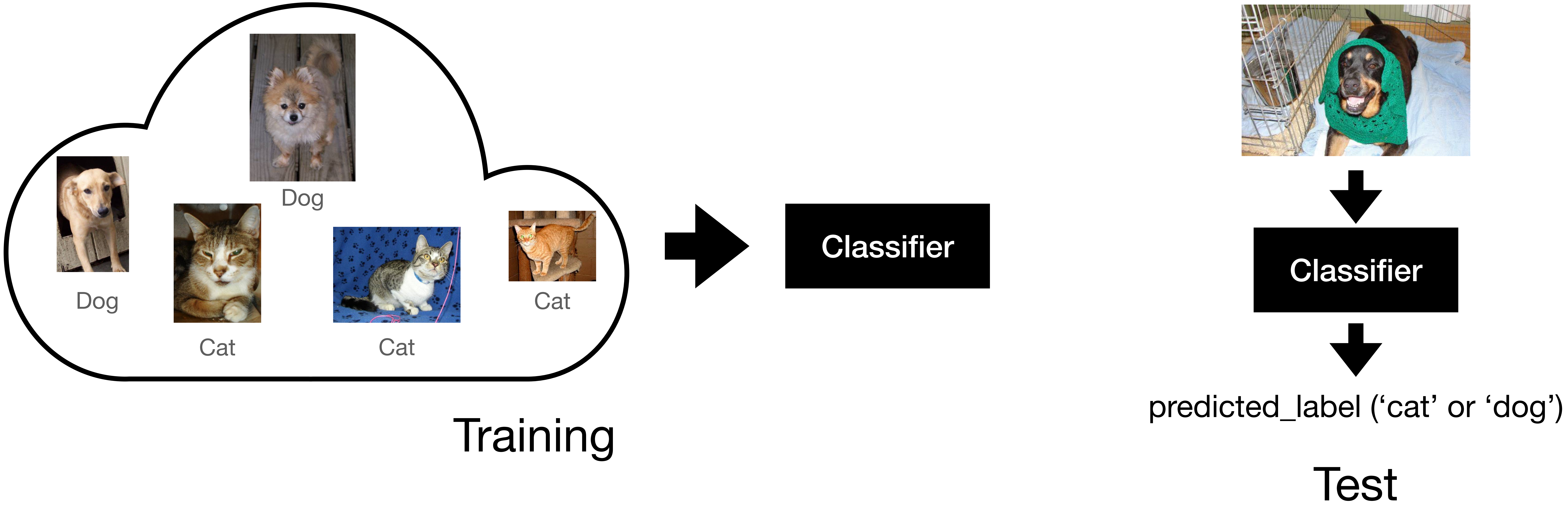
ML == statistics + computer science

Central ML concepts: **generalisation** and **algorithms**

Central objectives of ML: finding **statistically** and **computationally** **efficient** **algorithms** to solve **generalisation** problems

Case study: classification of cat and dog images

Case study: classification of cat and dog images



(Images from kaggle’s dogs vs cats competition)

Case study: classification of cat and dog images

Classifier

Case study: classification of cat and dog images

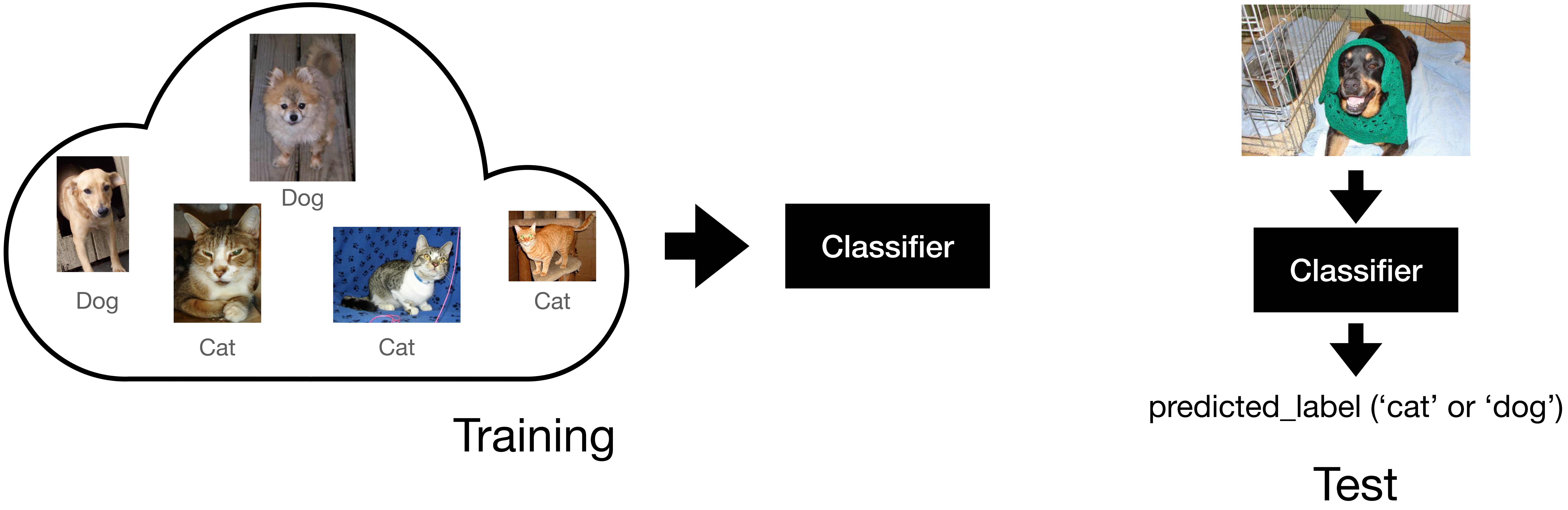
Input

Classifier

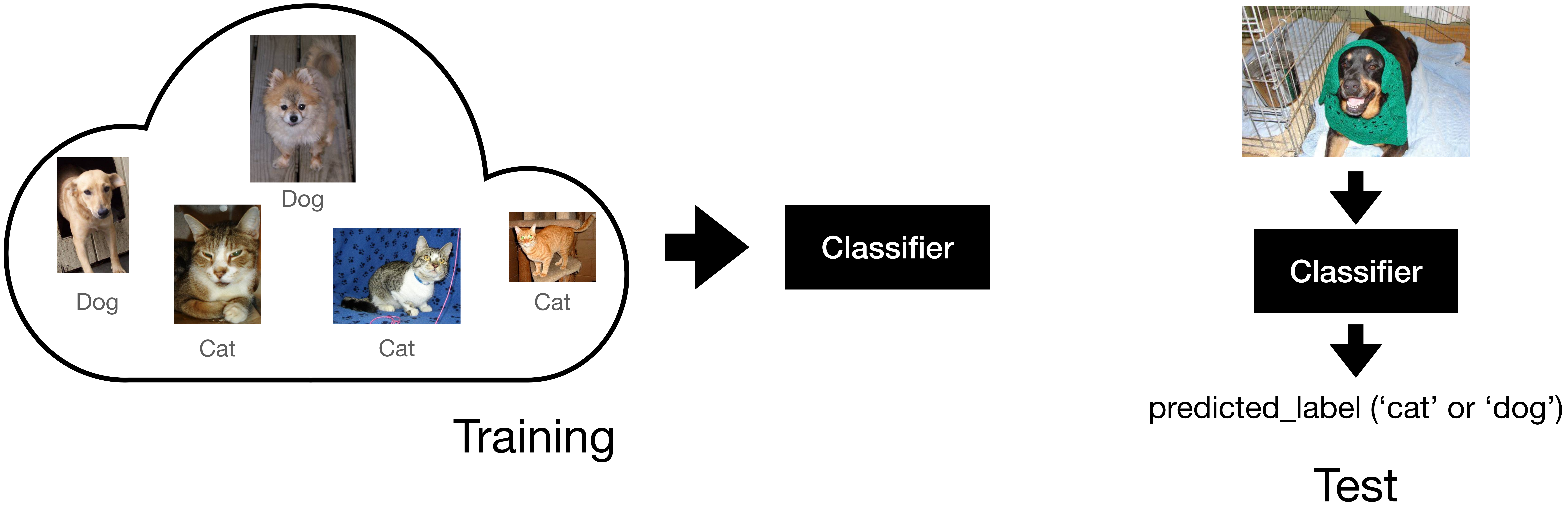
Output

predicted_label (‘cat’ or ‘dog’)

Case study: classification of cat and dog images

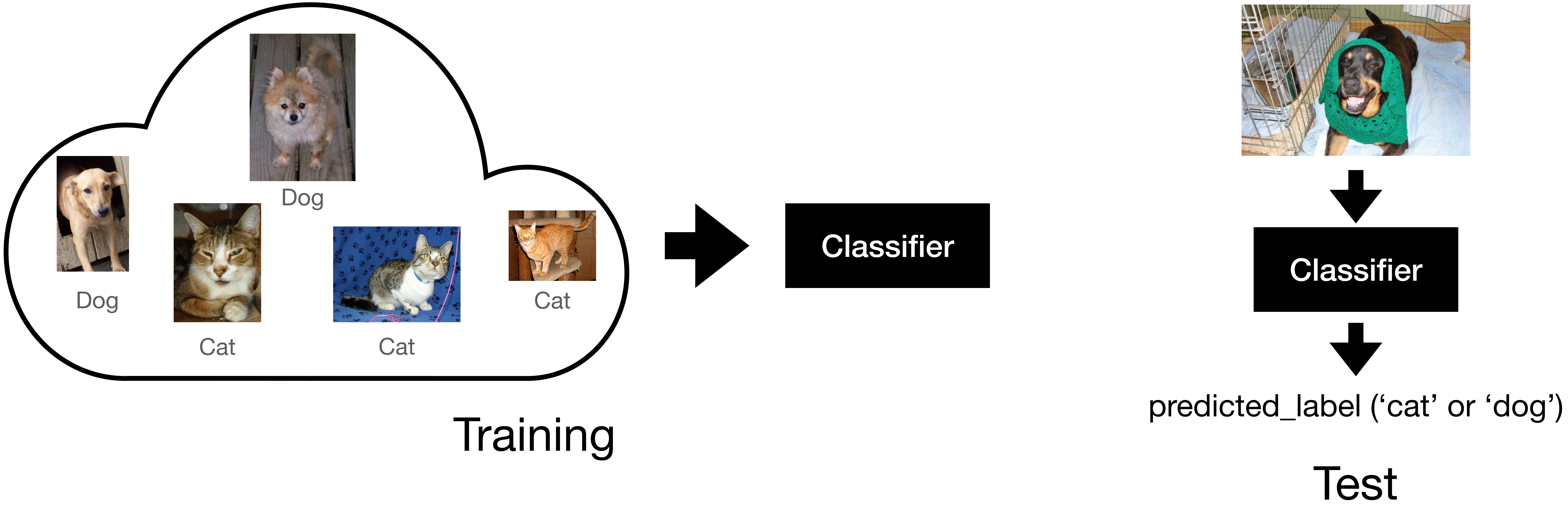


Case study: classification of cat and dog images



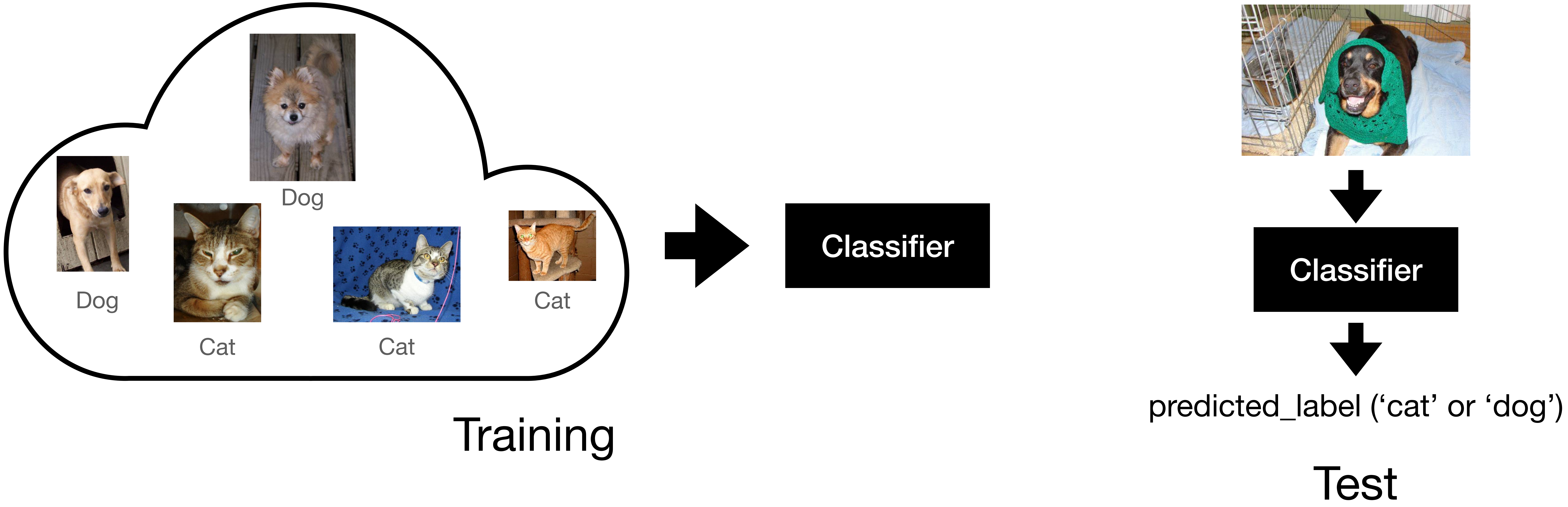
Central concepts in ML: **generalisation** and **algorithms**

Case study: classification of cat and dog images



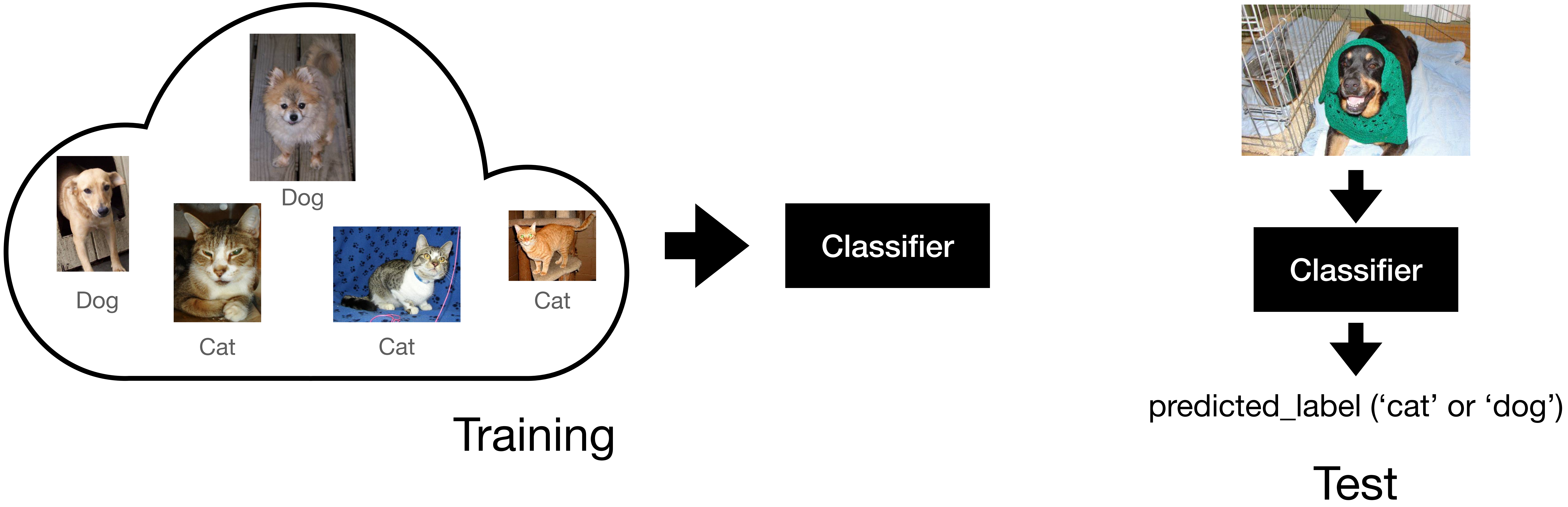
Central concepts in ML: <u>generalisation</u> and **algorithms**

Case study: classification of cat and dog images



Central concepts in ML: **generalisation** and **algorithms**

Case study: classification of cat and dog images

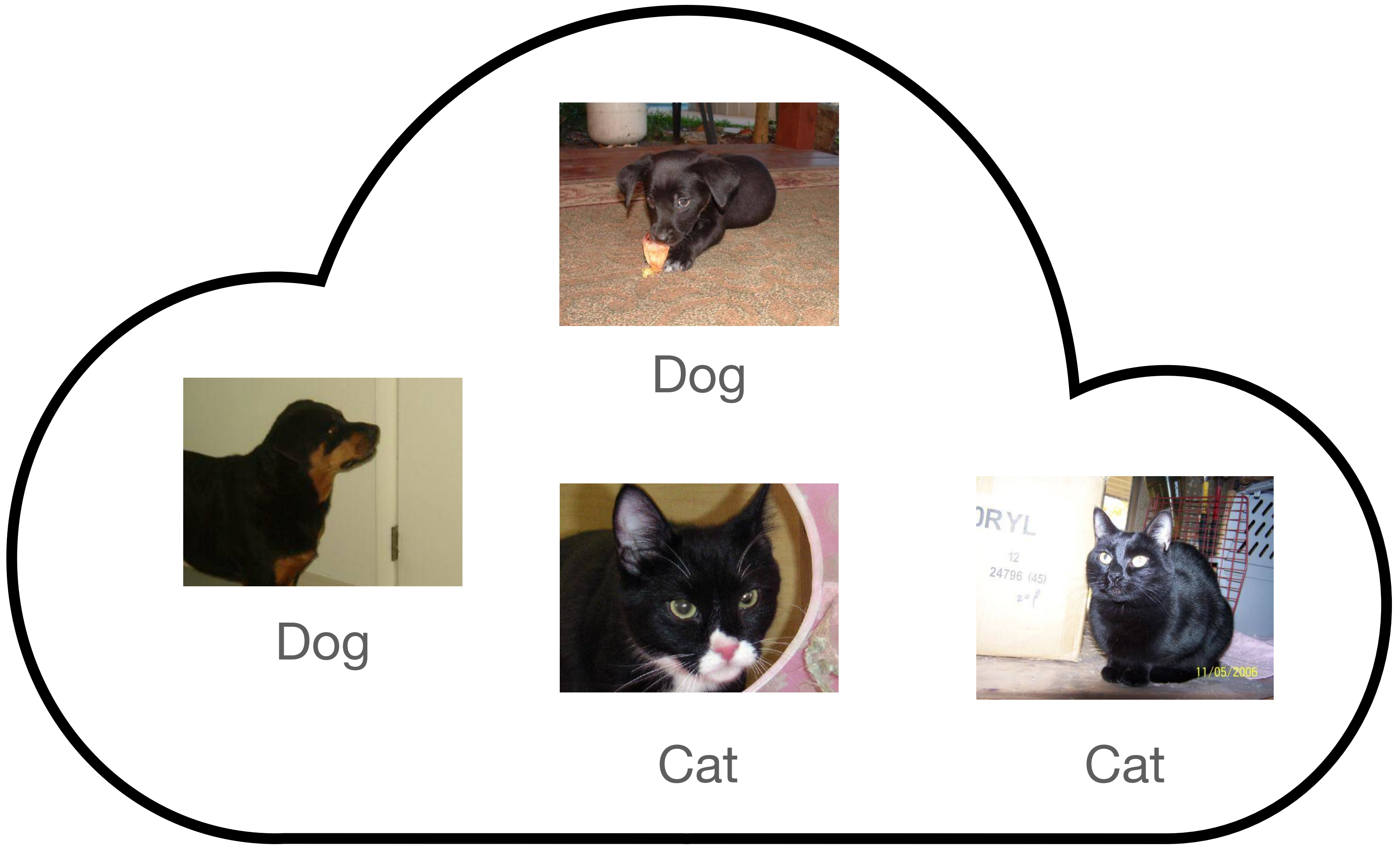


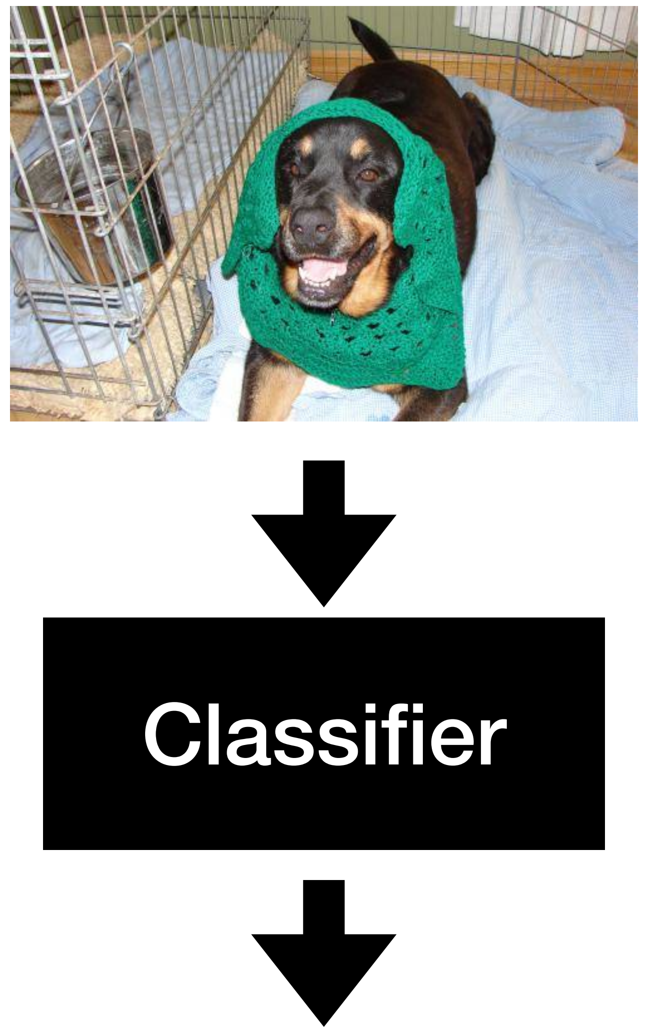
Central objectives of ML: finding **statistically** and **computationally** **efficient algorithms** to solve **generalisation** problems

Case study: classification of cat and dog images

Central objectives of ML: finding **statistically** and **computationally** **efficient** **algorithms** to solve **generalisation** problems

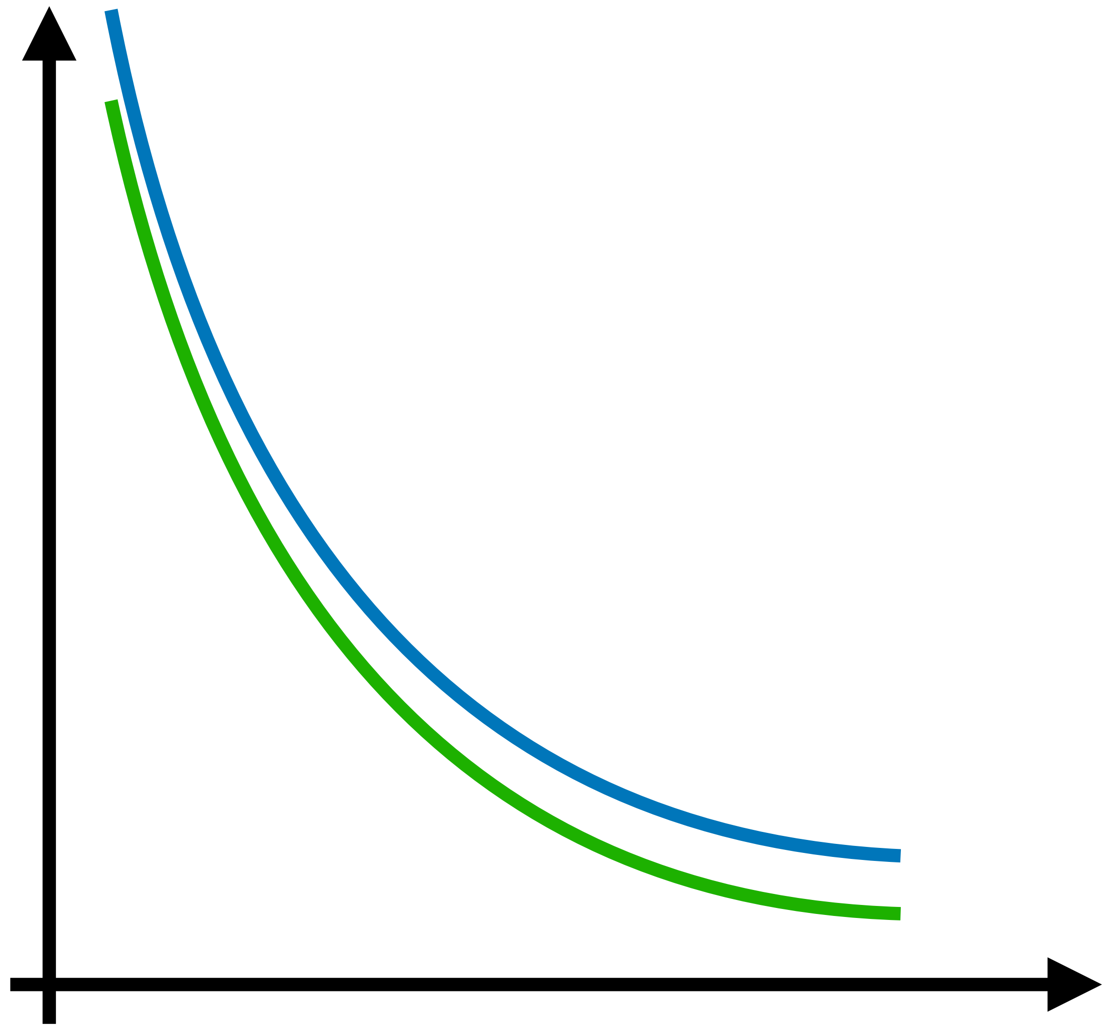
Case study: classification of cat and dog images





predicted_label (‘cat’ or ‘dog’)

Average classification error on test images



Number of images in training set

Held out test set

Test

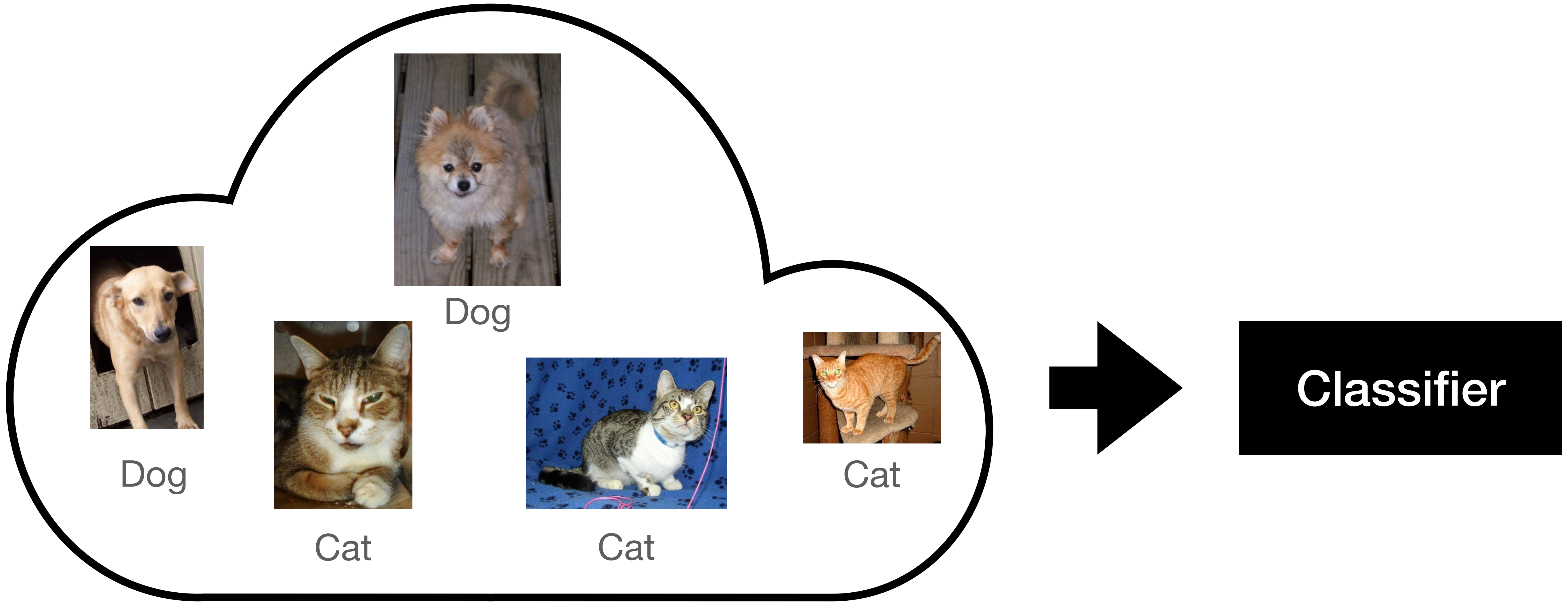
Results

Central objectives of ML: finding **statistically** and **computationally** **efficient** **algorithms** to solve **generalisation** problems

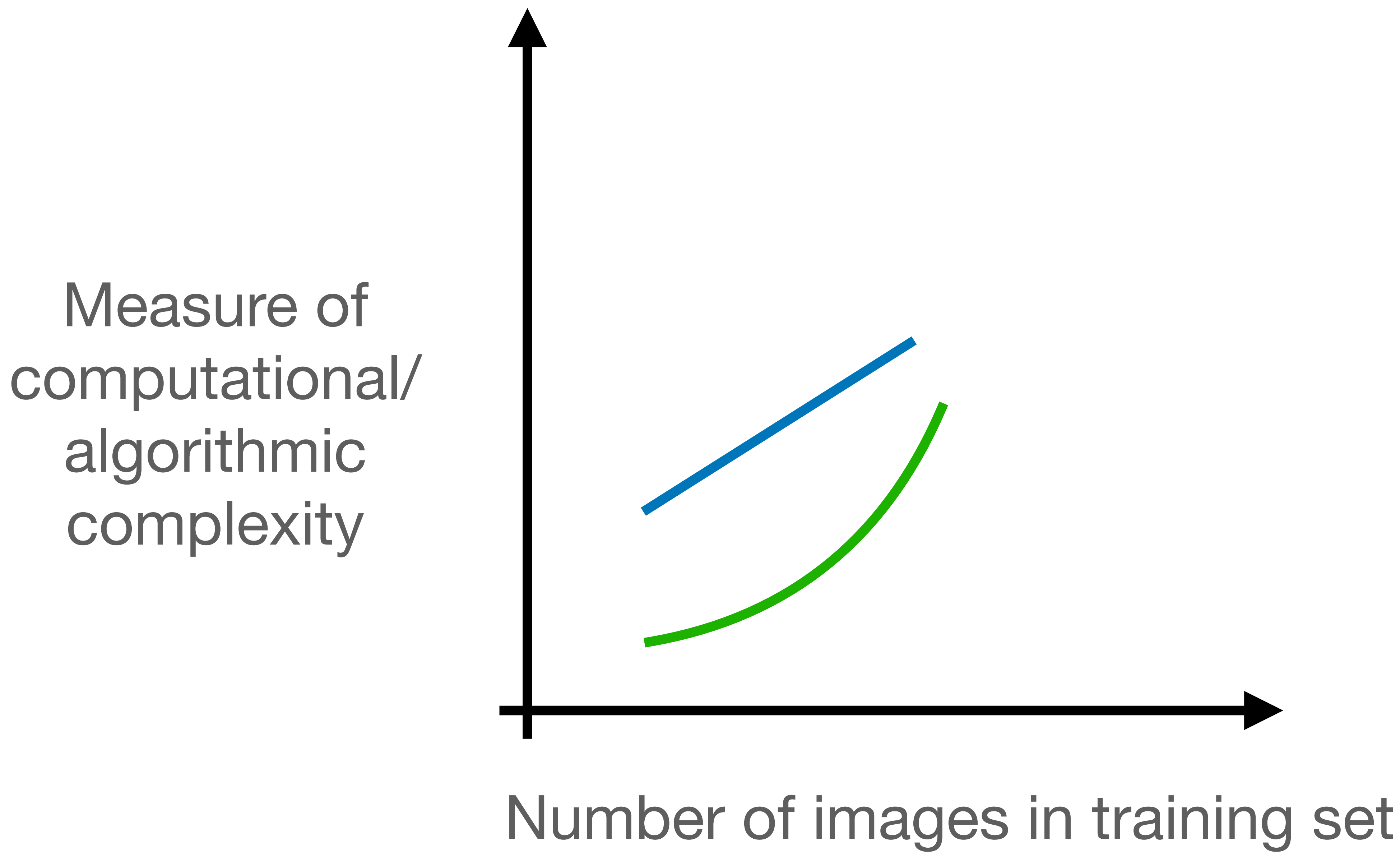
Case study: classification of cat and dog images

Central objectives of ML: finding **statistically** and **computationally** **efficient** **algorithms** to solve **generalisation** problems

Case study: classification of cat and dog images

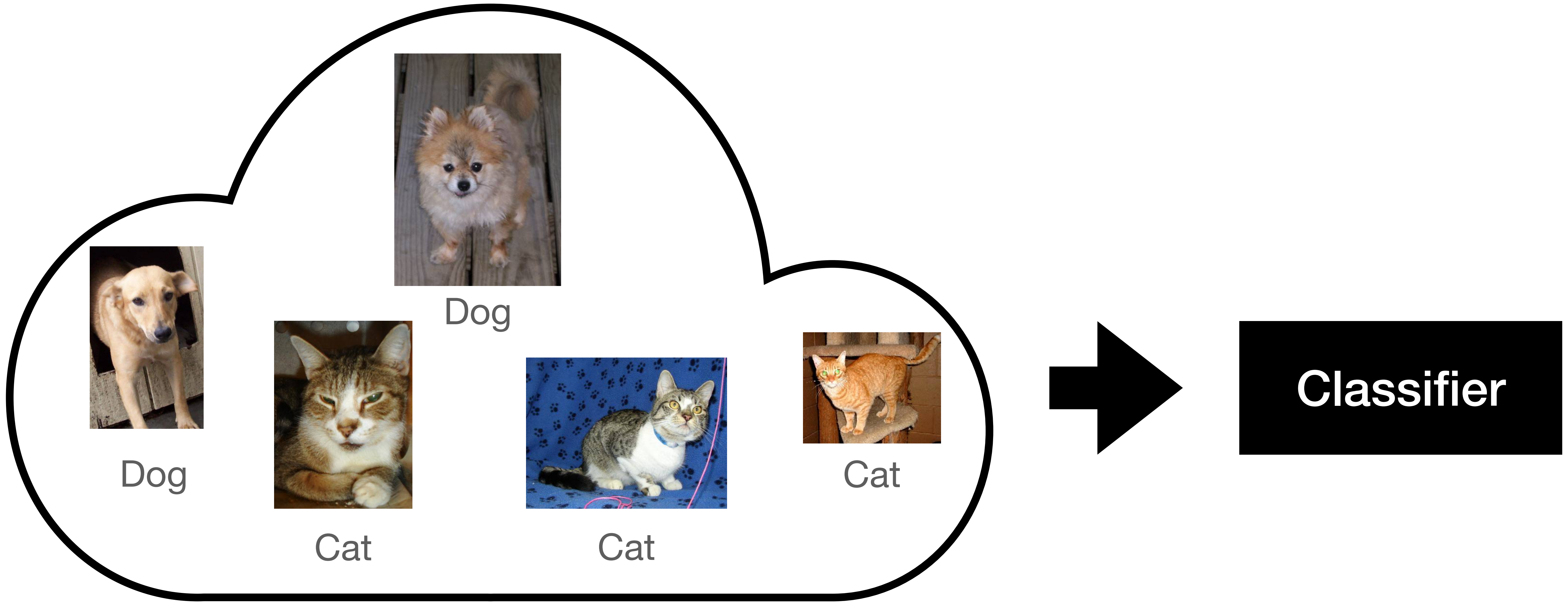


Training

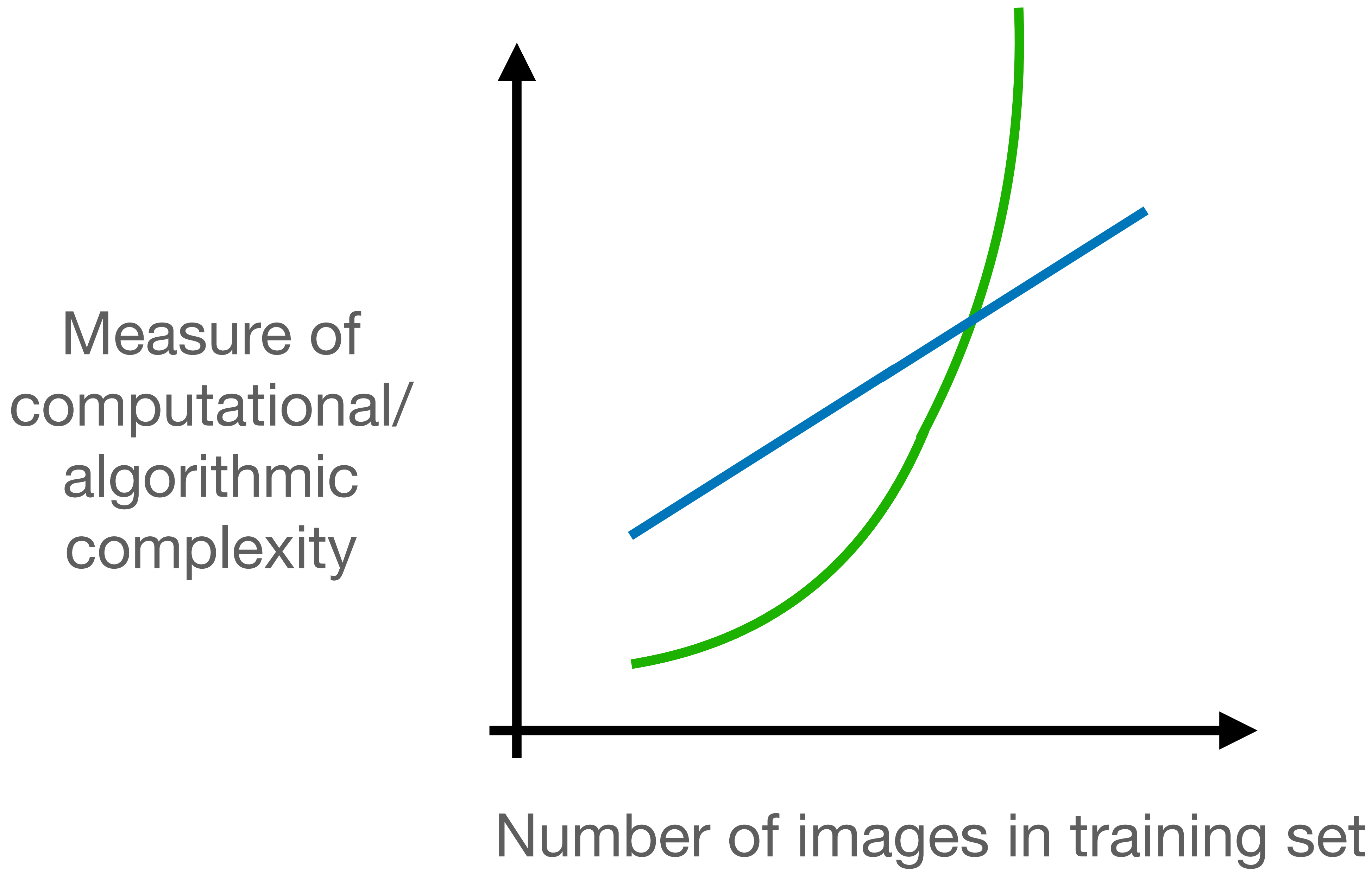


Central objectives of ML: finding **statistically** and **computationally** **efficient** **algorithms** to solve **generalisation** problems

Case study: classification of cat and dog images

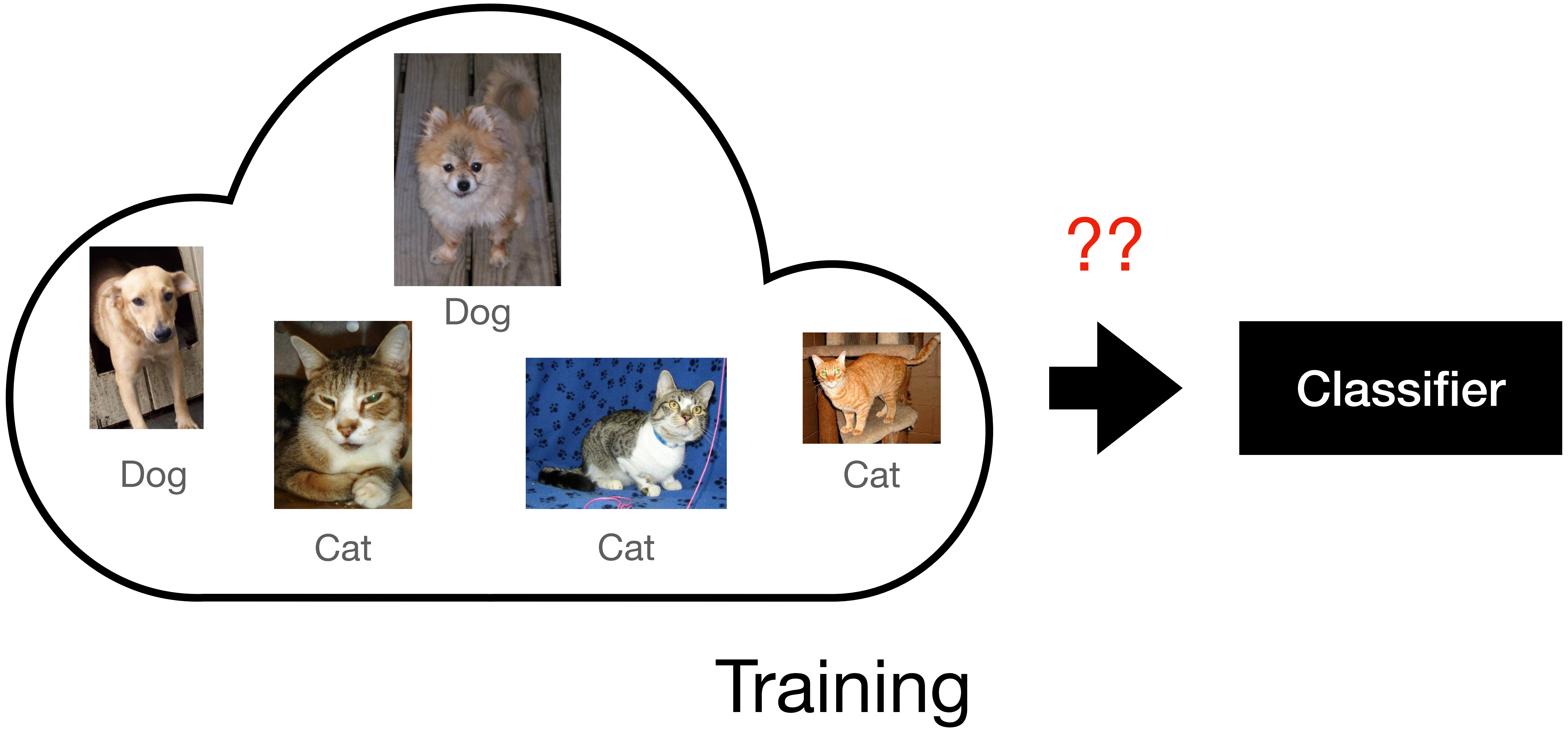


Training

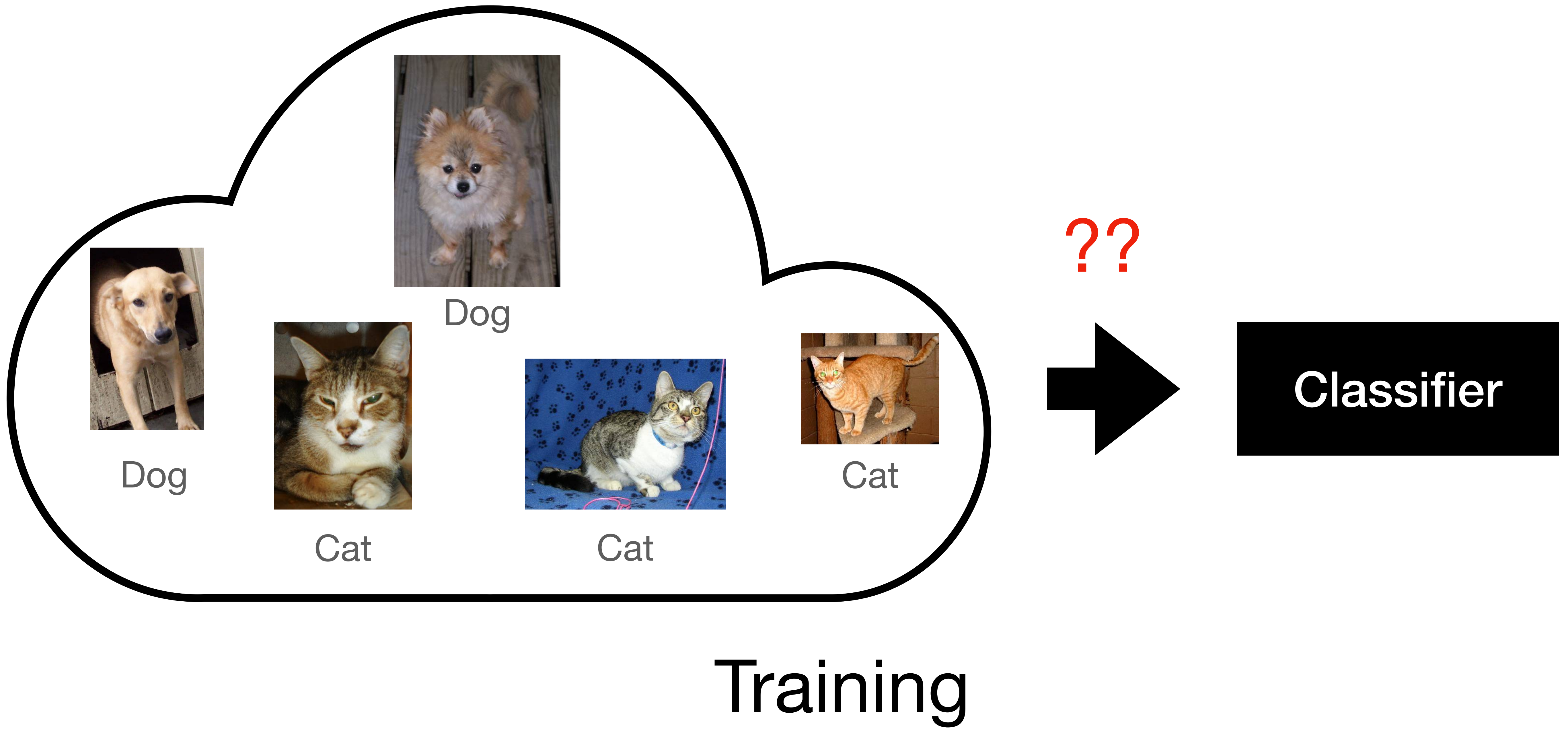


Central objectives of ML: finding **statistically** and **computationally** **efficient** **algorithms** to solve **generalisation** problems

Case study: classification of cat and dog images



Case study: classification of cat and dog images



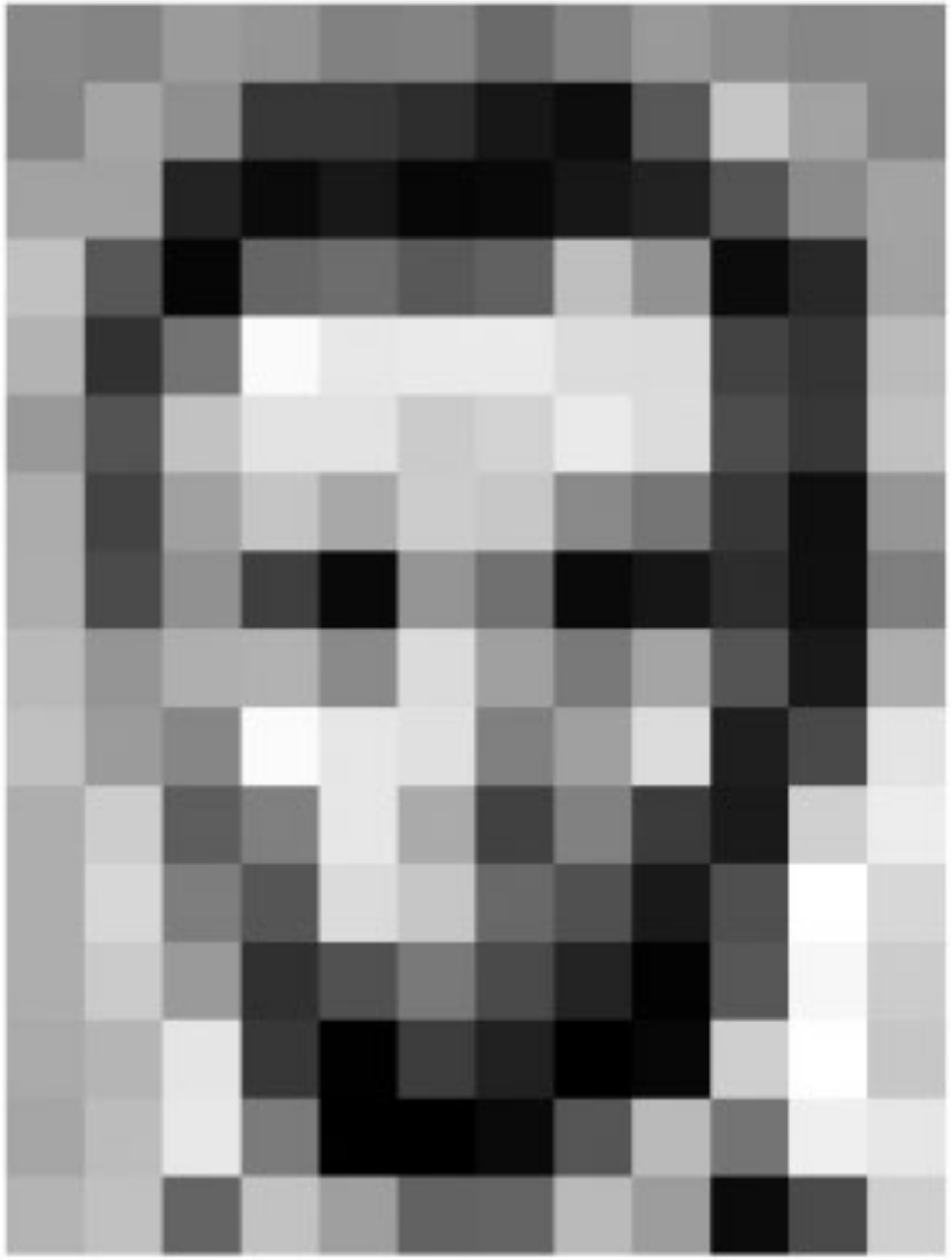
Case study: classification of cat and dog images

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

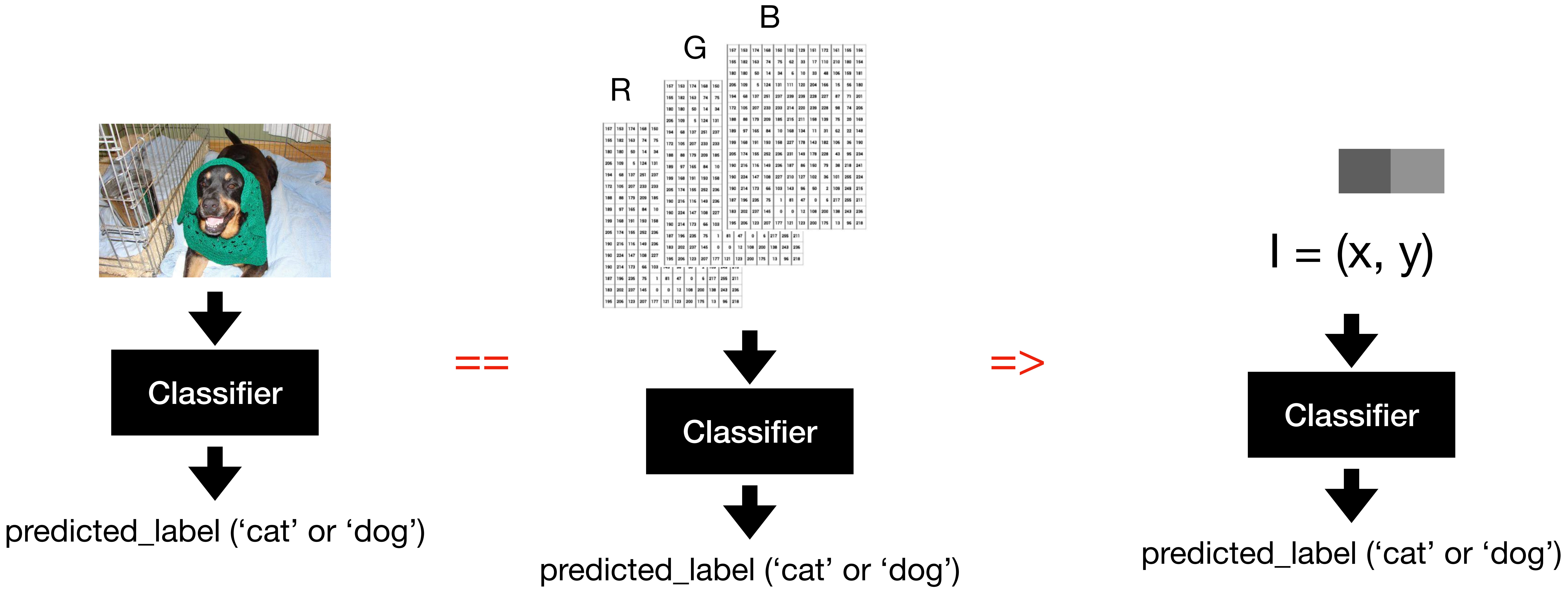
What is this?

Case study: classification of cat and dog images

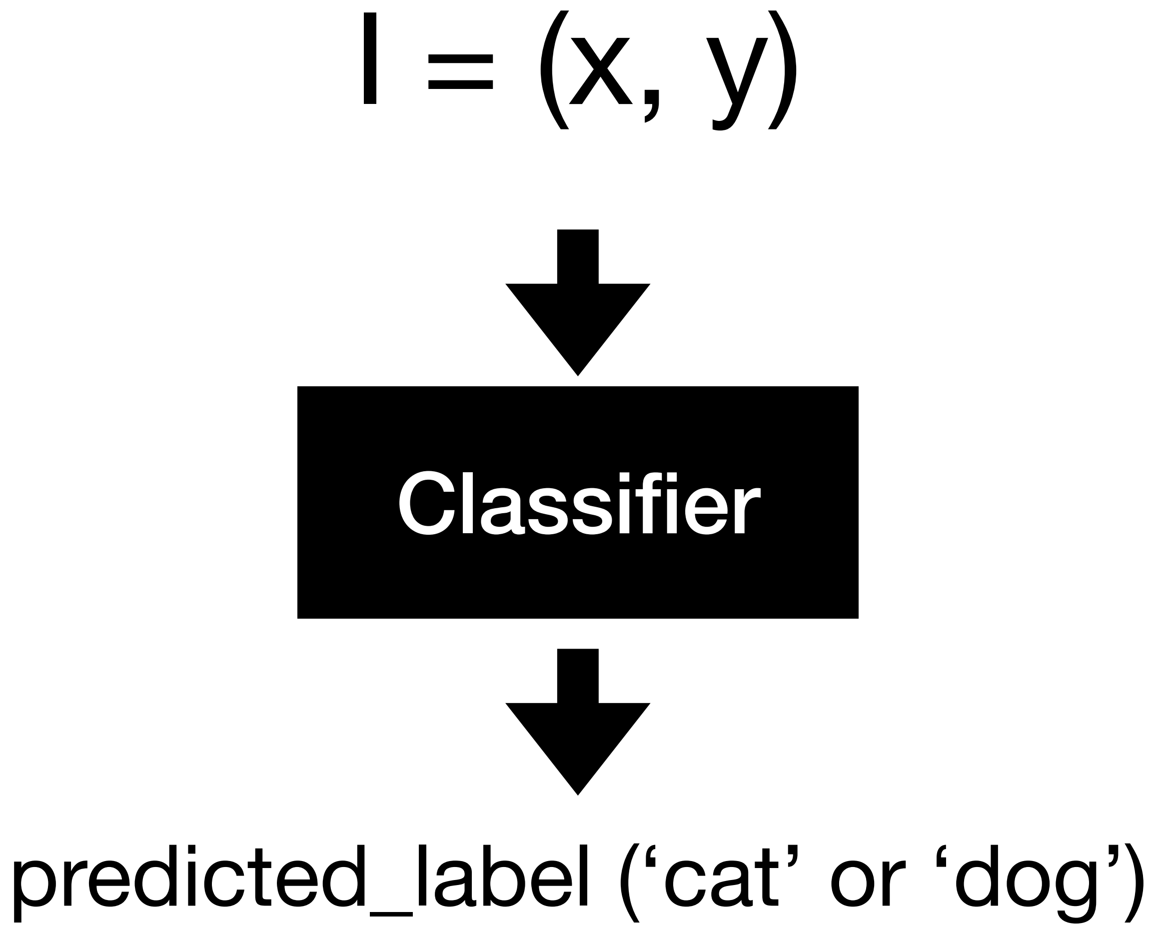
157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

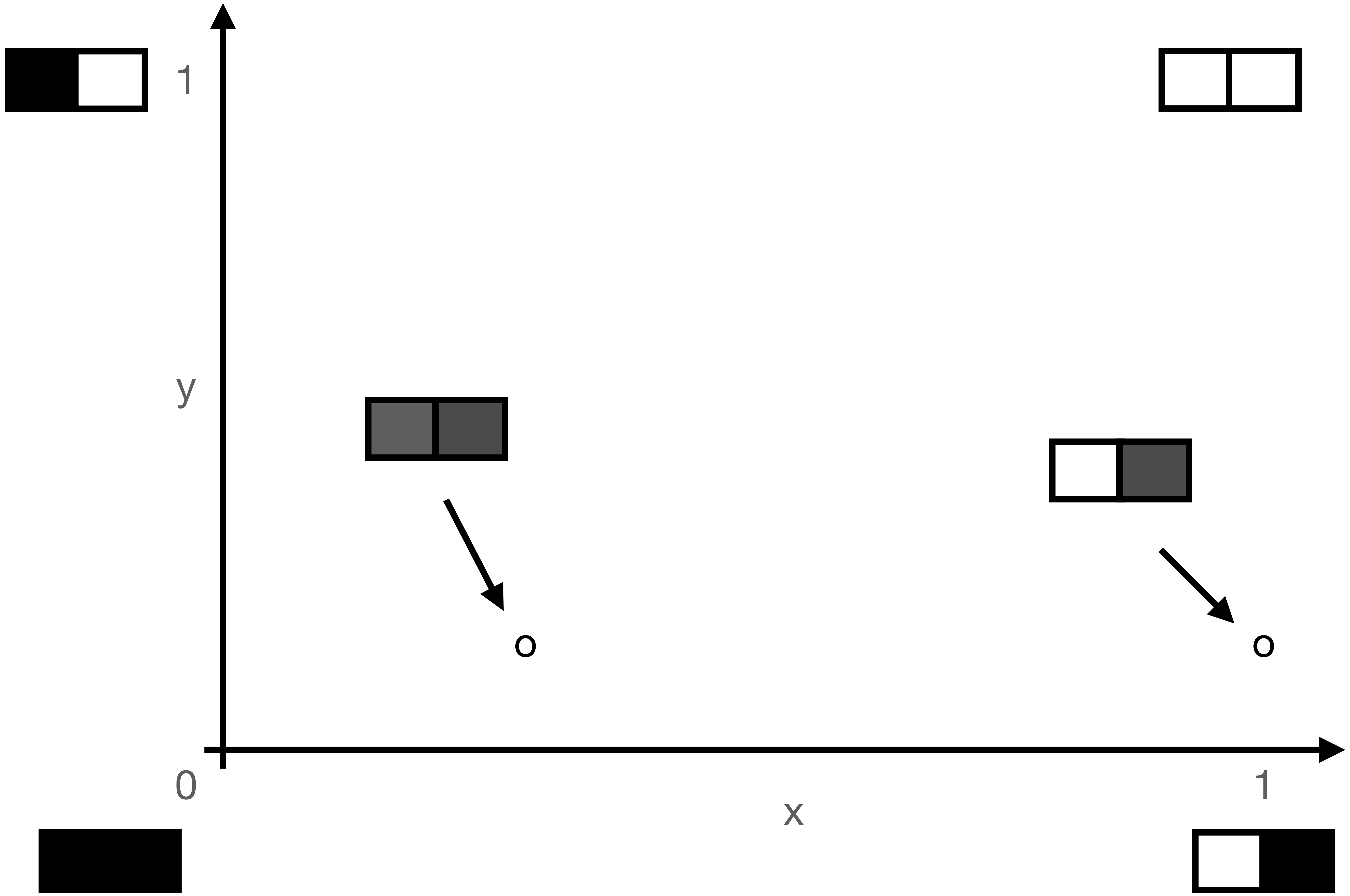


Case study: classification of cat and dog images



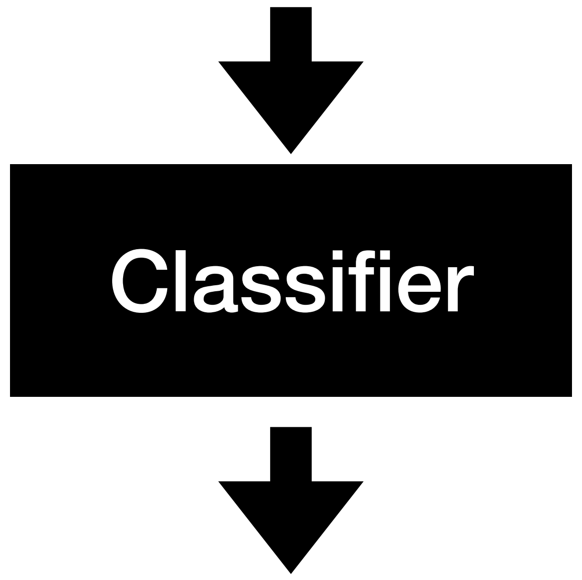
Case study: classification of cat and dog images



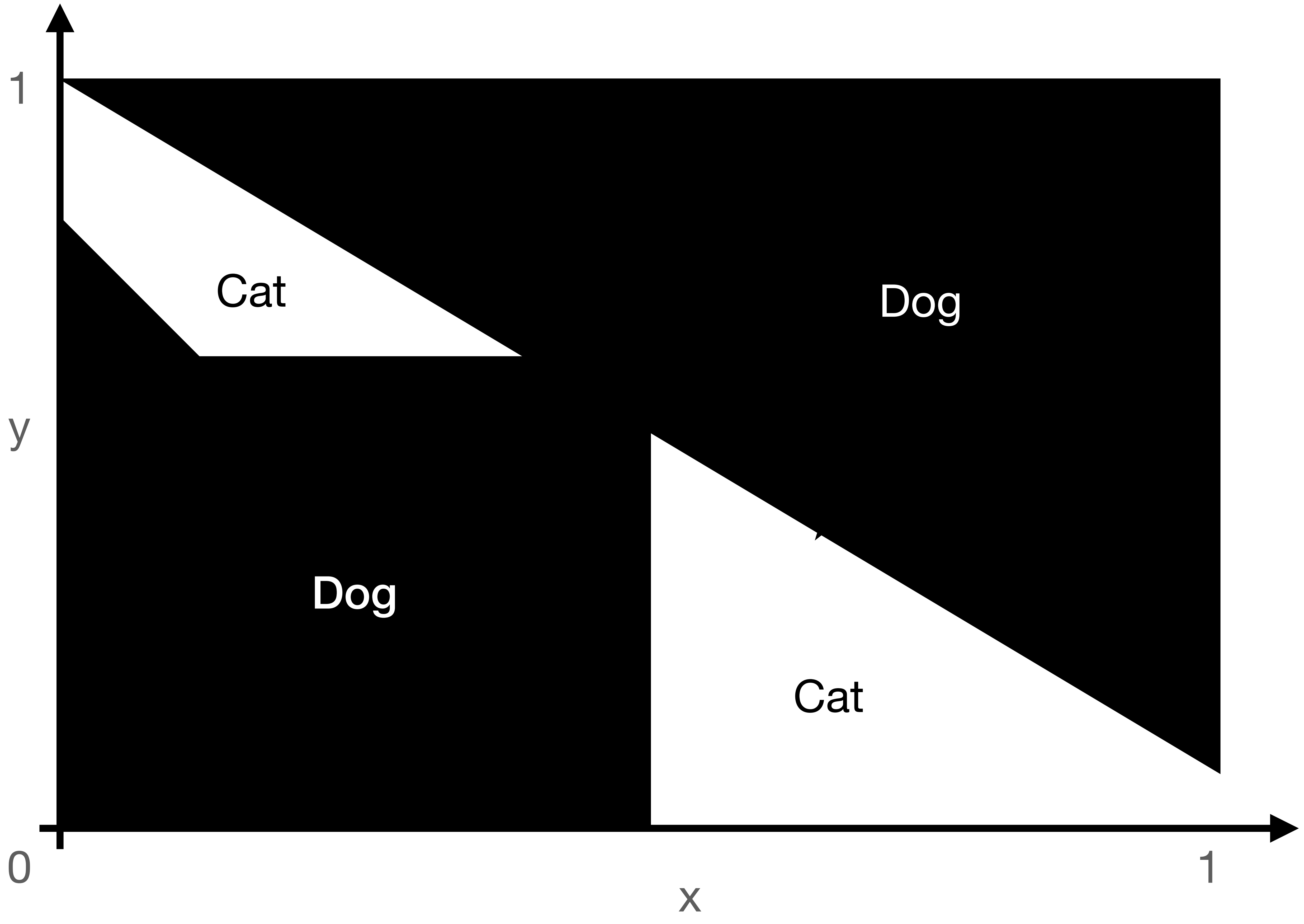


Case study: classification of cat and dog images

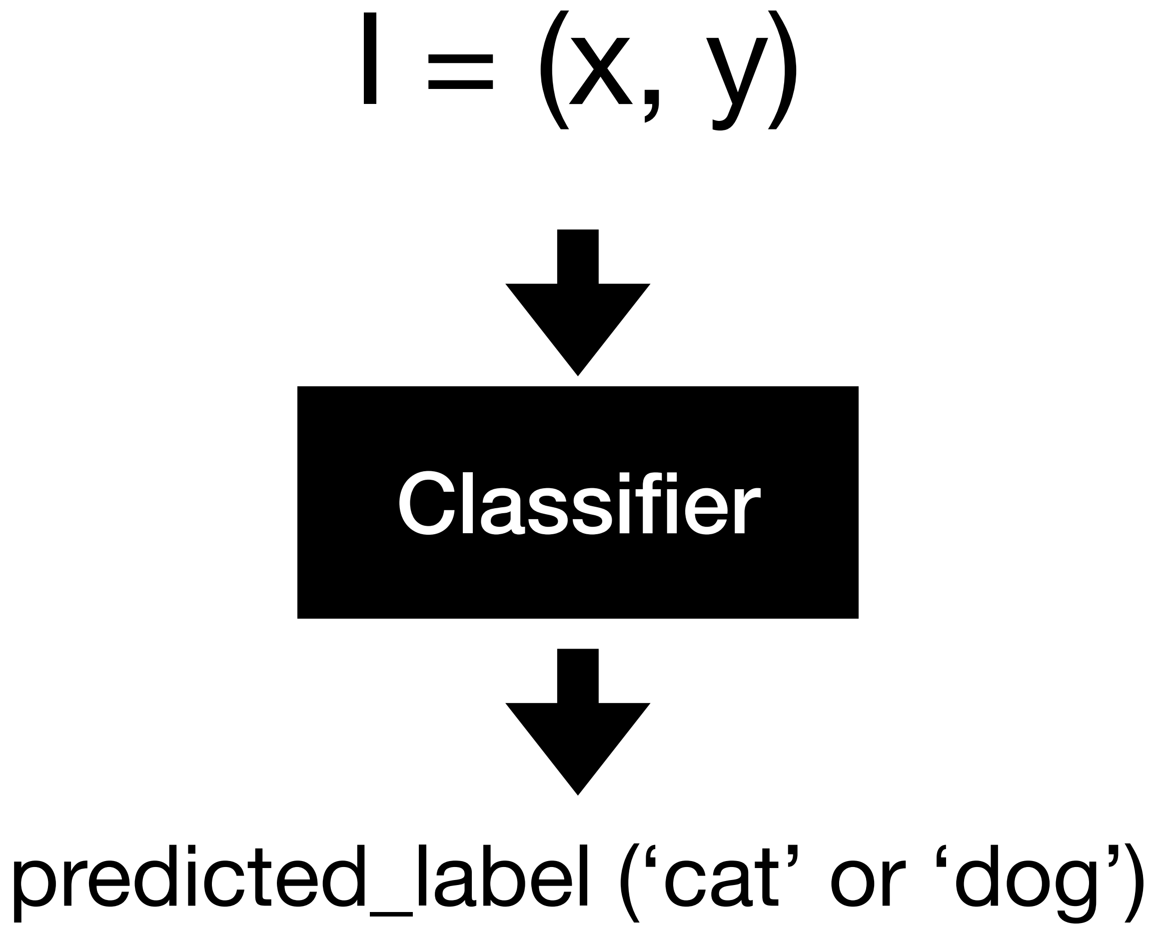
I = (x, y)

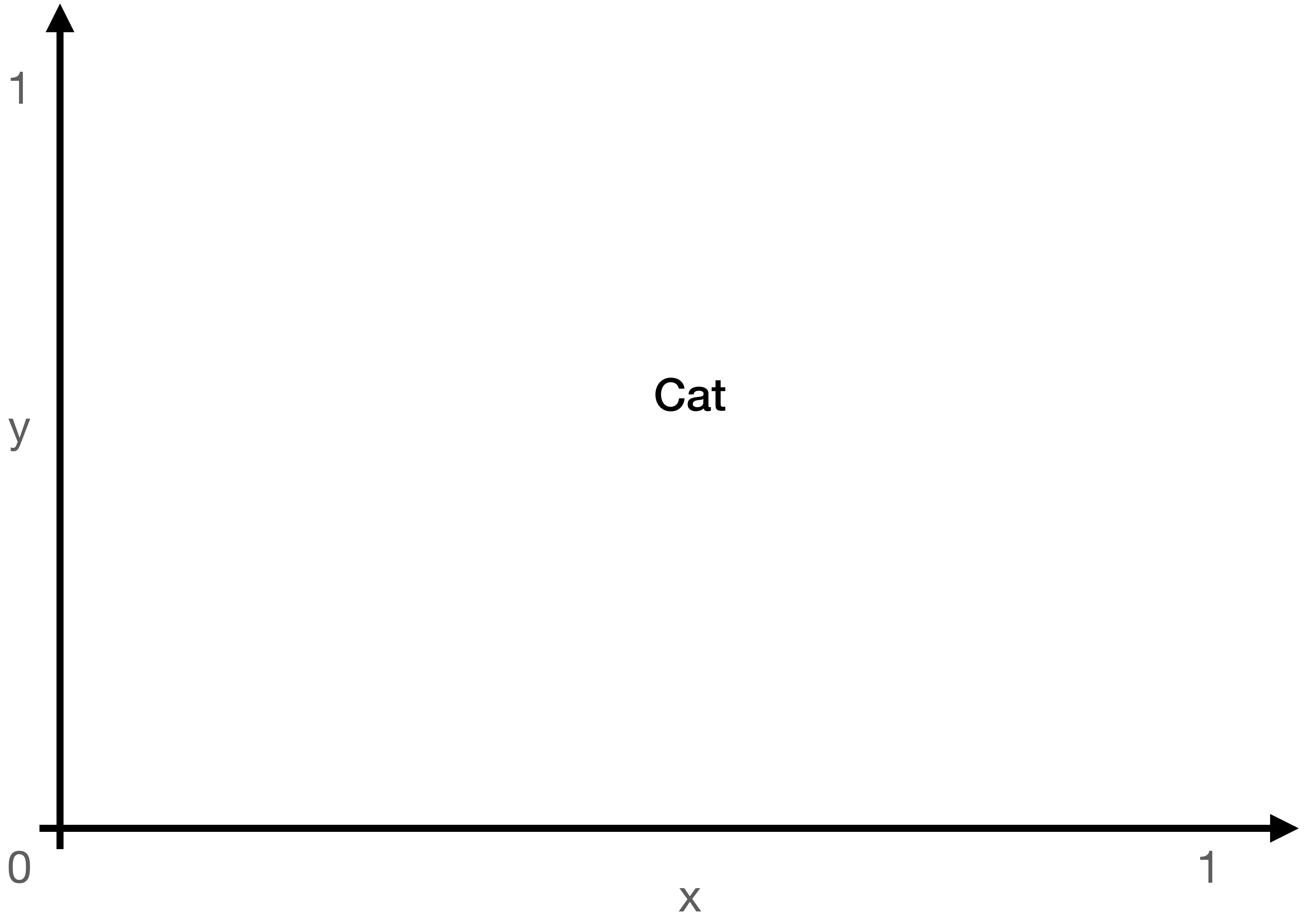


predicted_label (‘cat’ or ‘dog’)

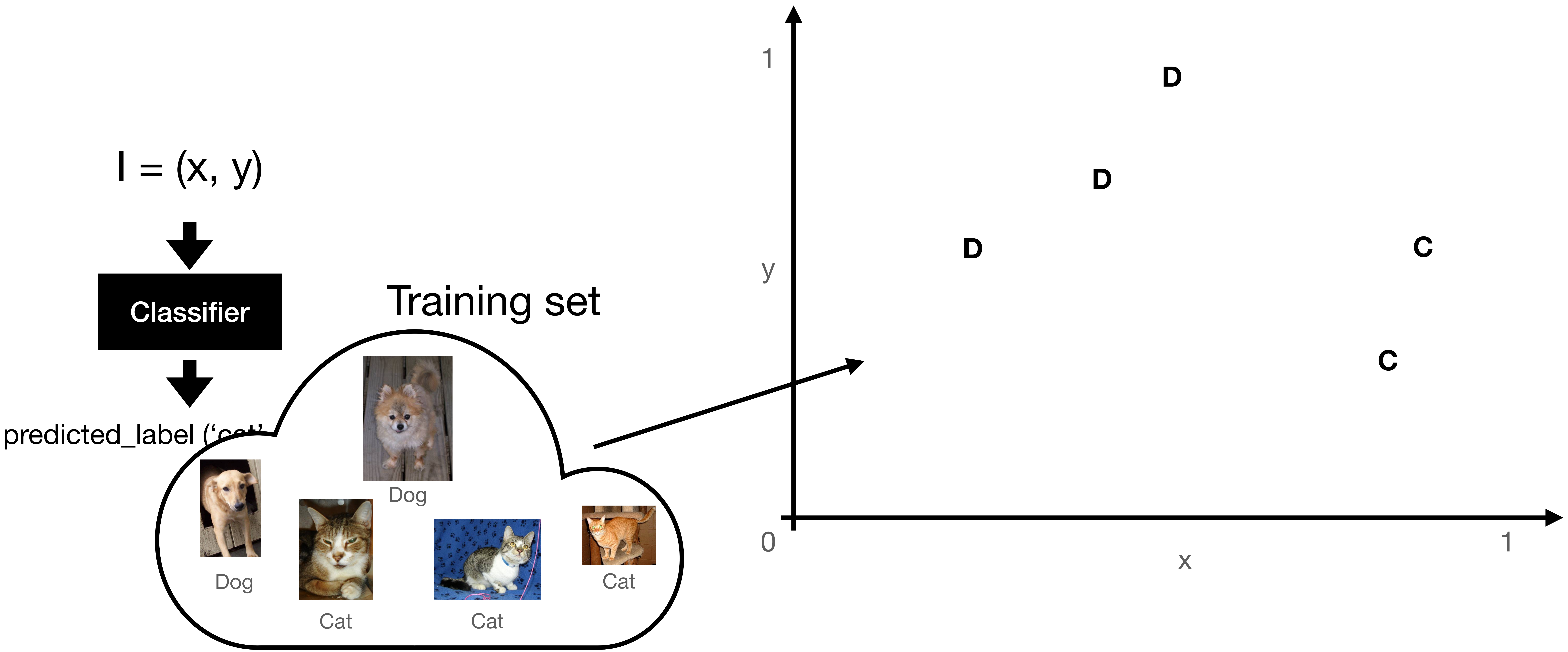


Case study: classification of cat and dog images

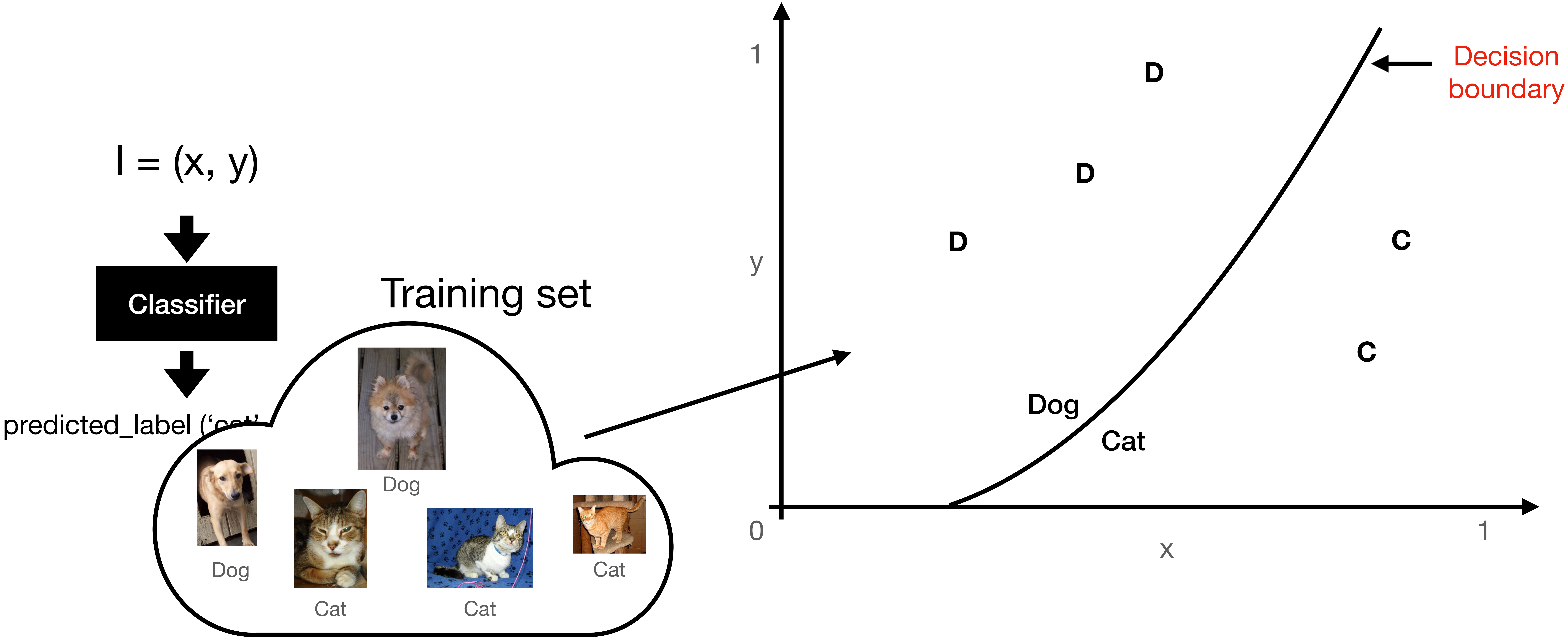




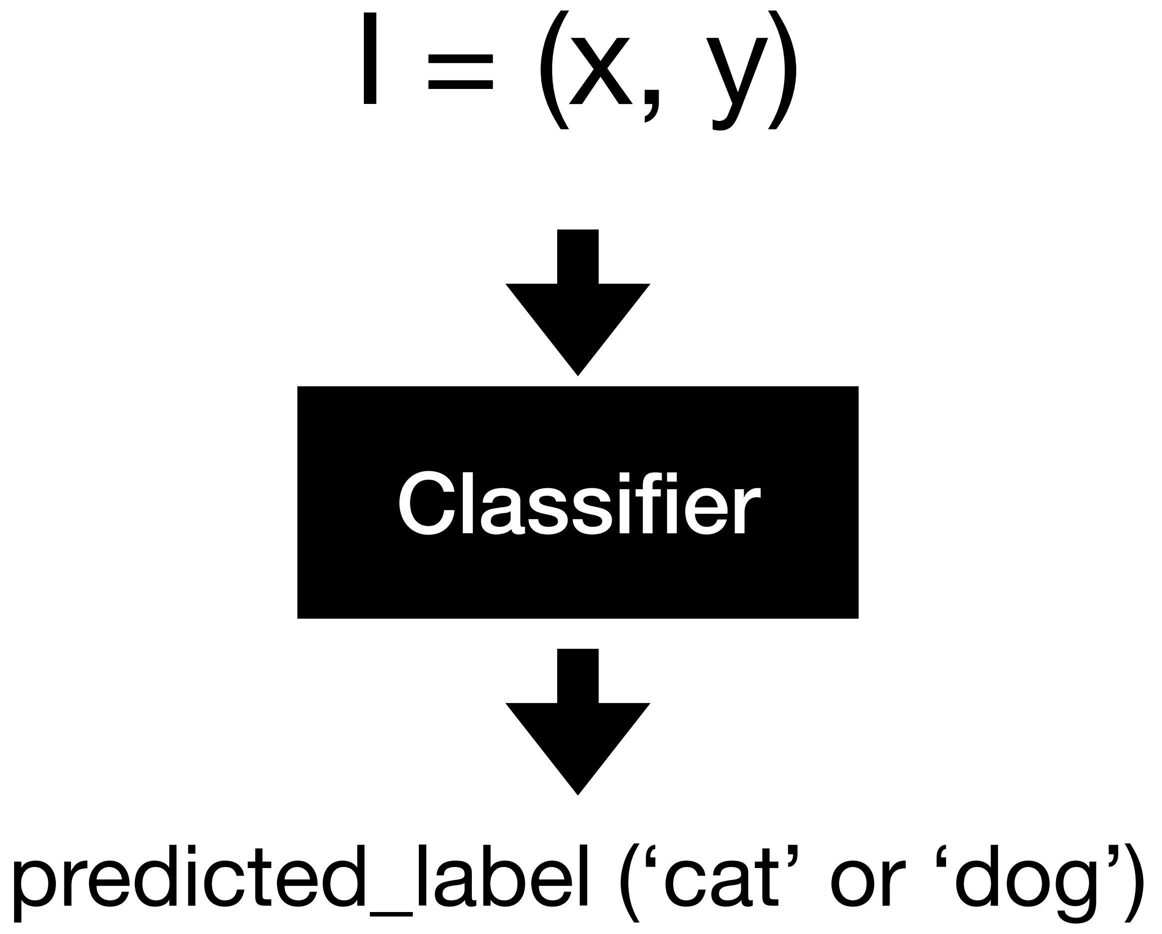
Case study: classification of cat and dog images

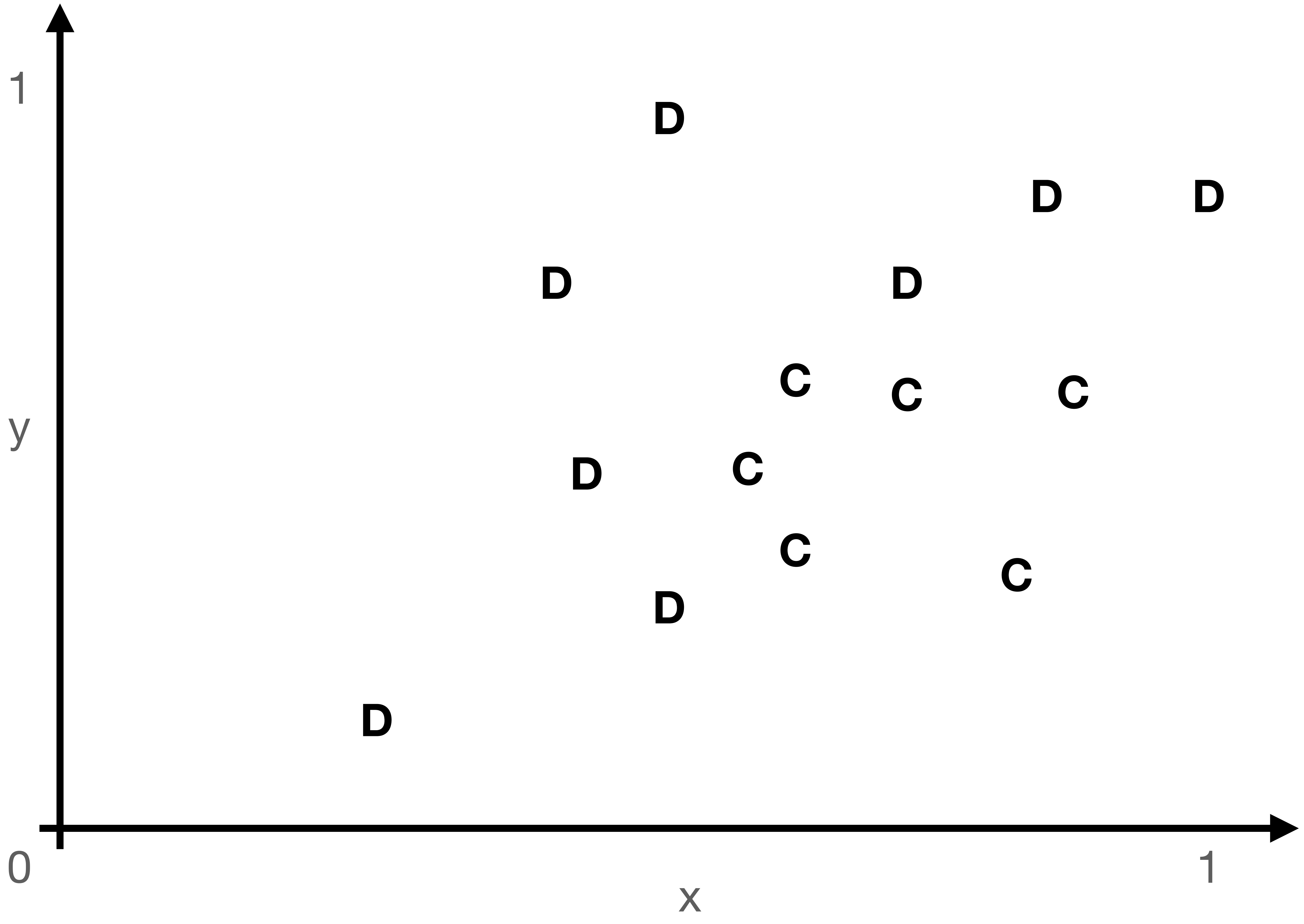


Case study: classification of cat and dog images

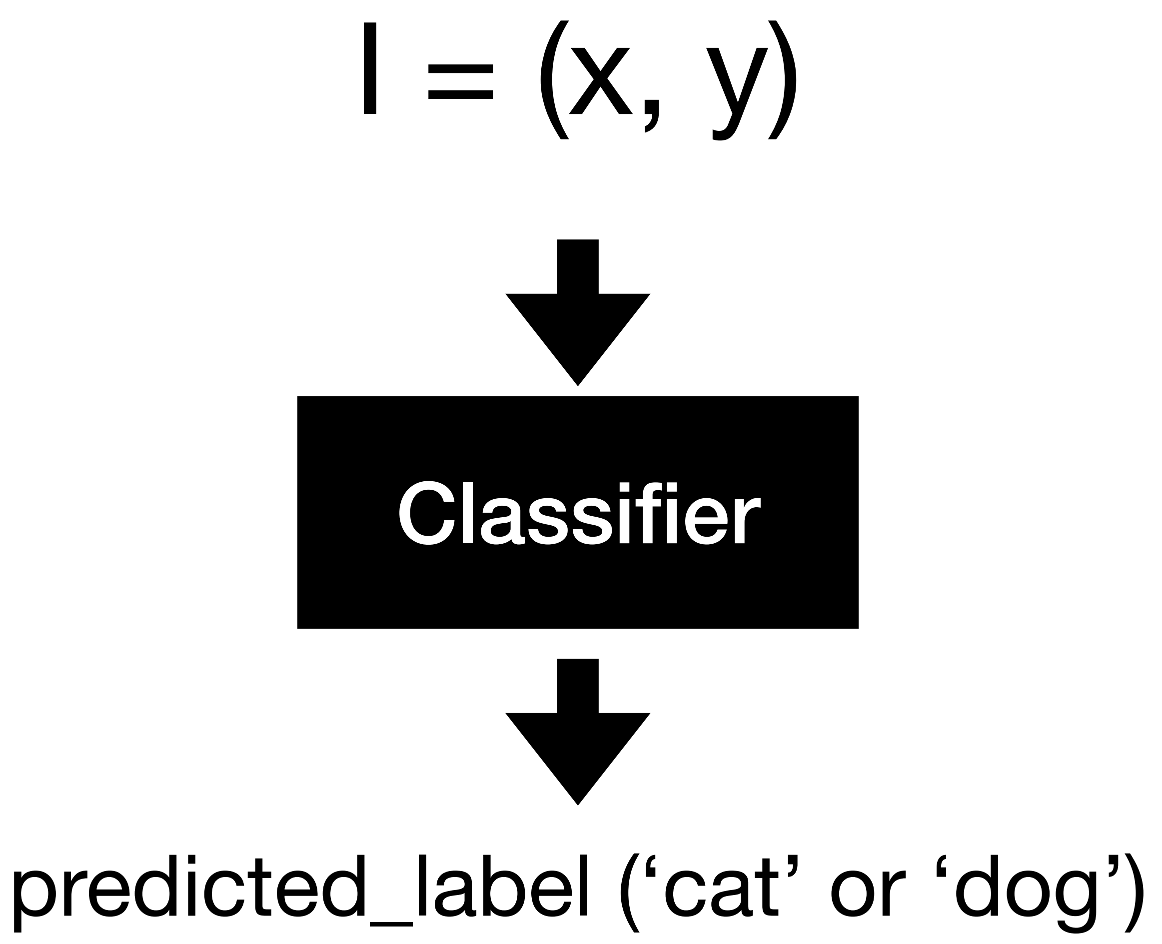


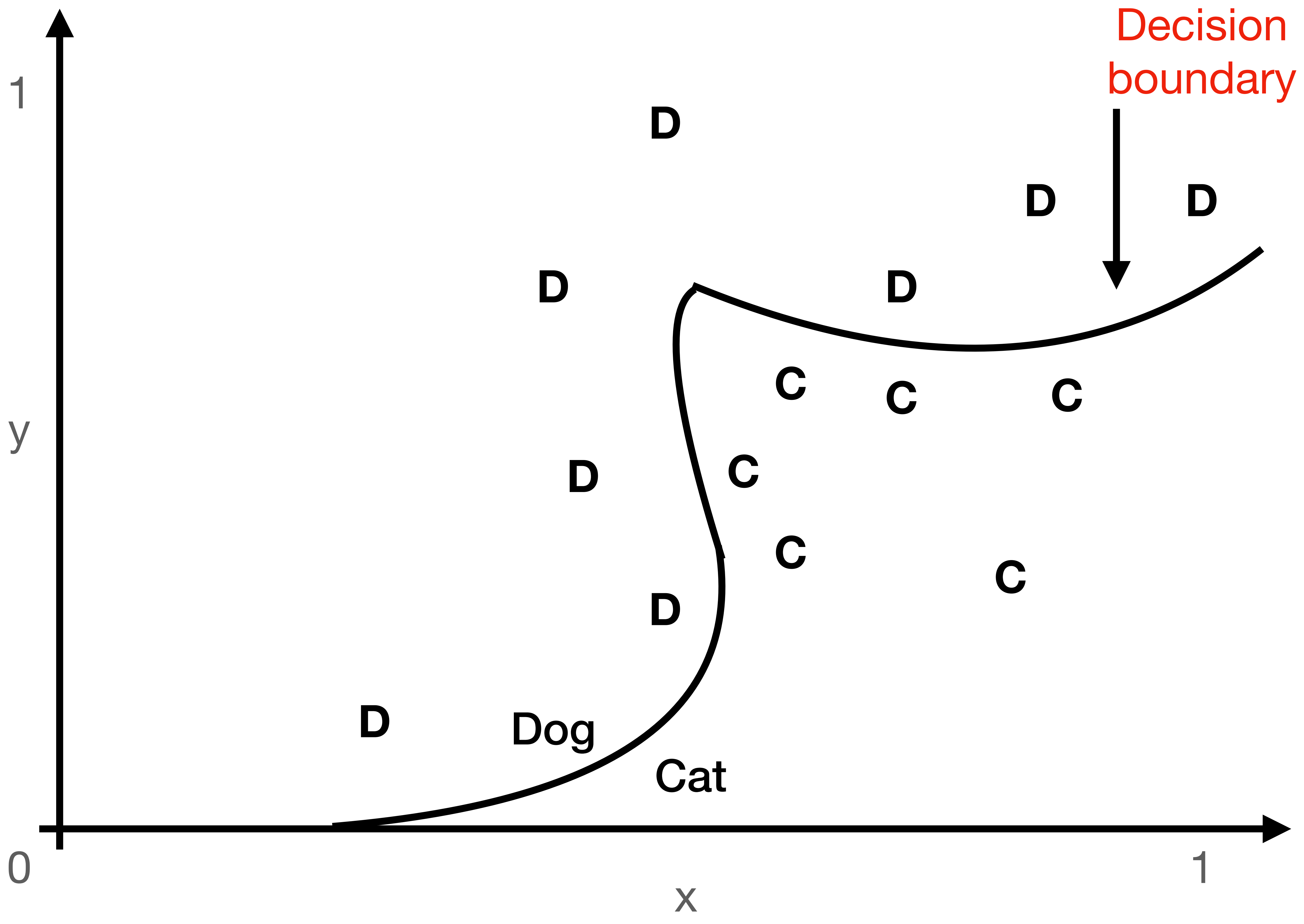
Case study: classification of cat and dog images



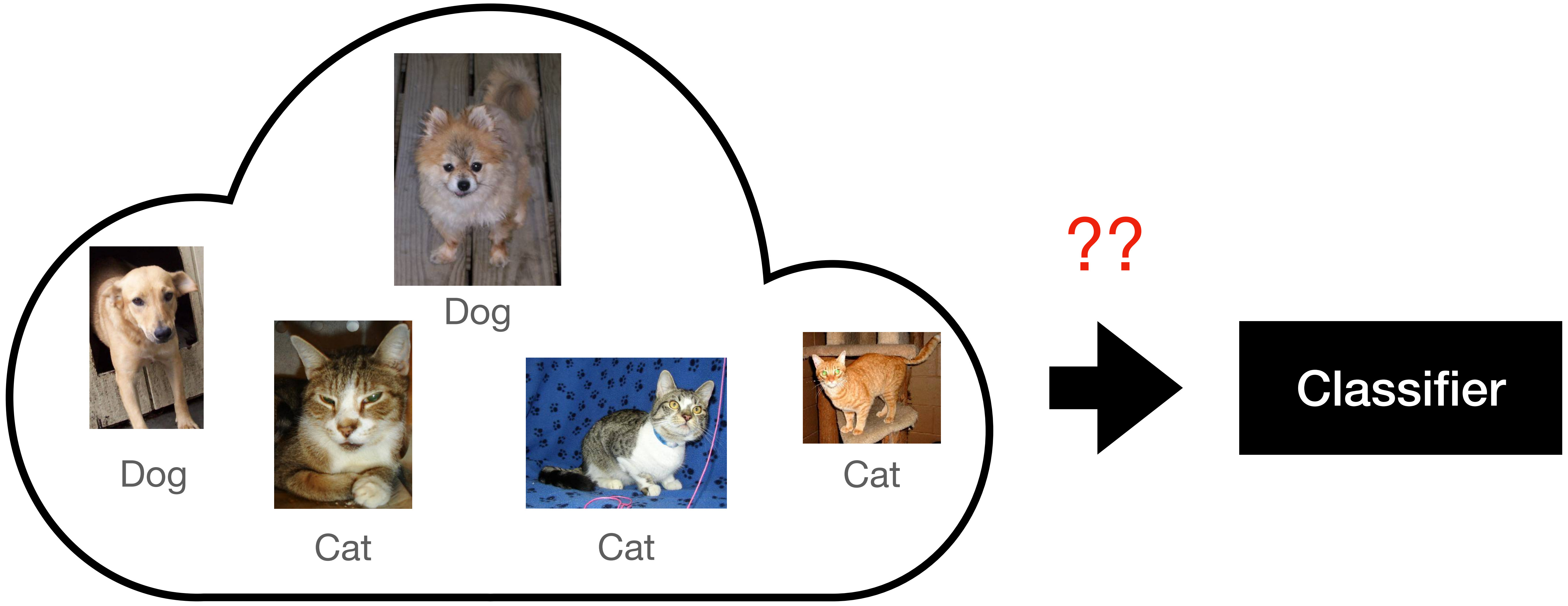


Case study: classification of cat and dog images

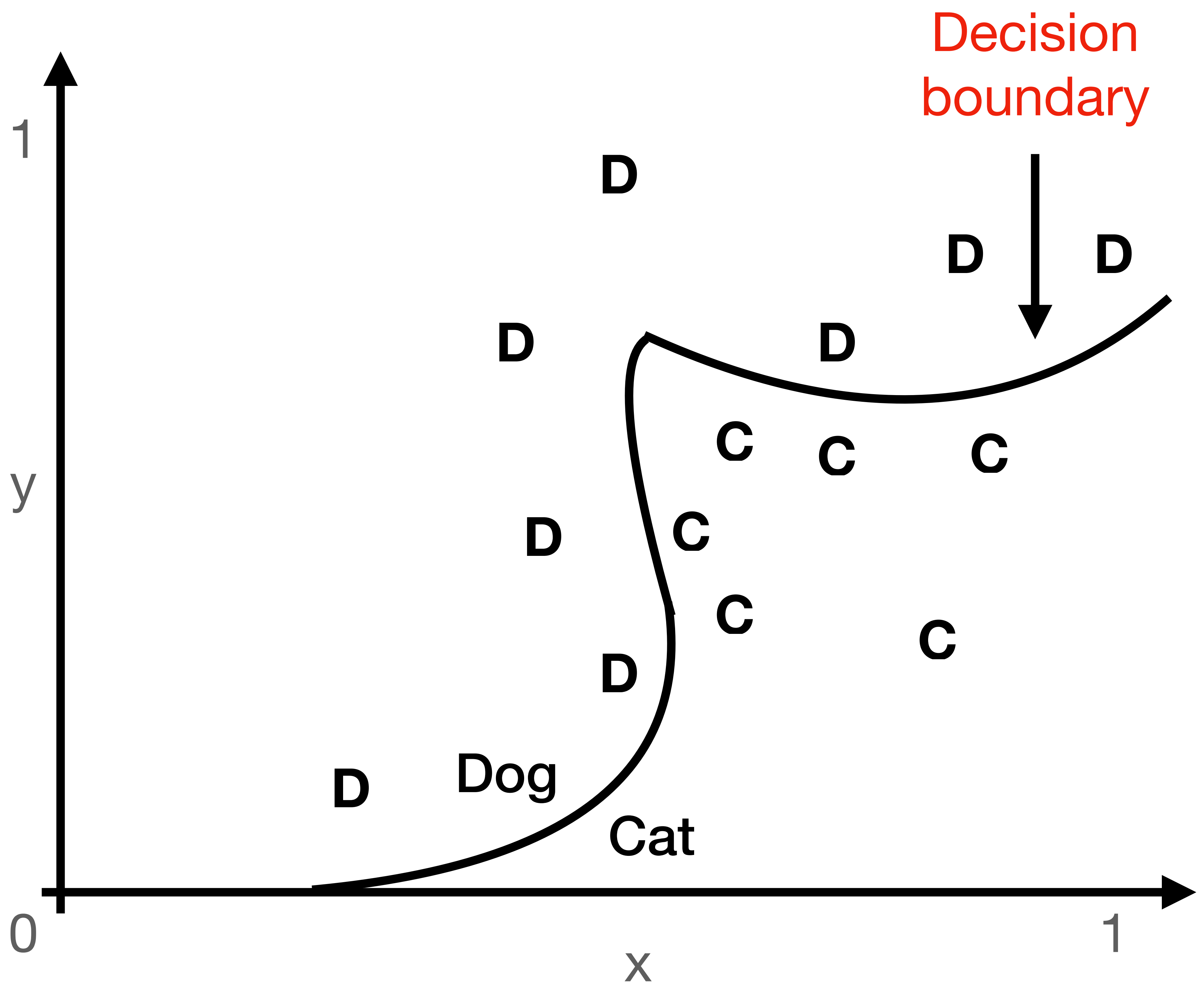




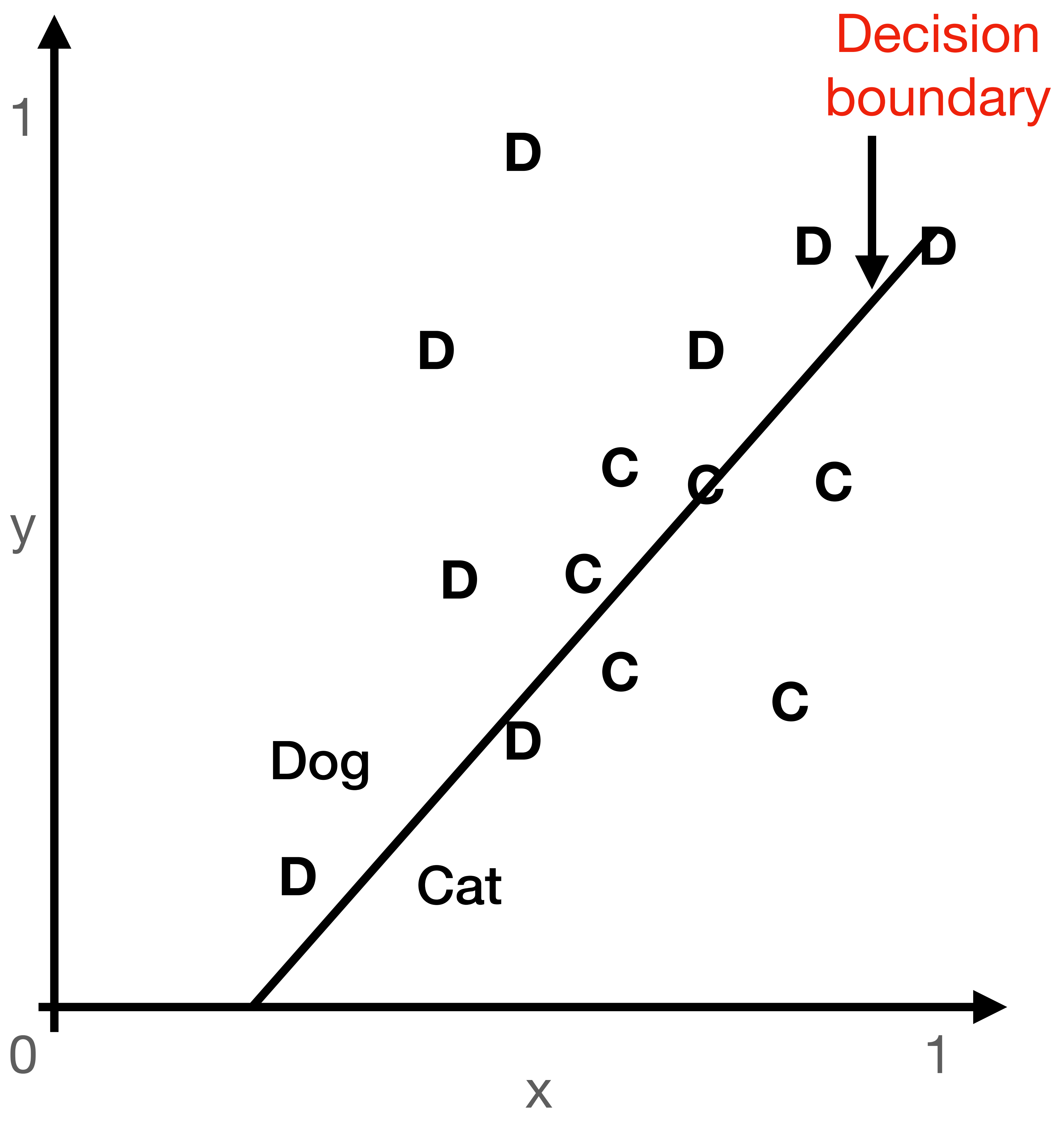
Case study: classification of cat and dog images



Training procedure

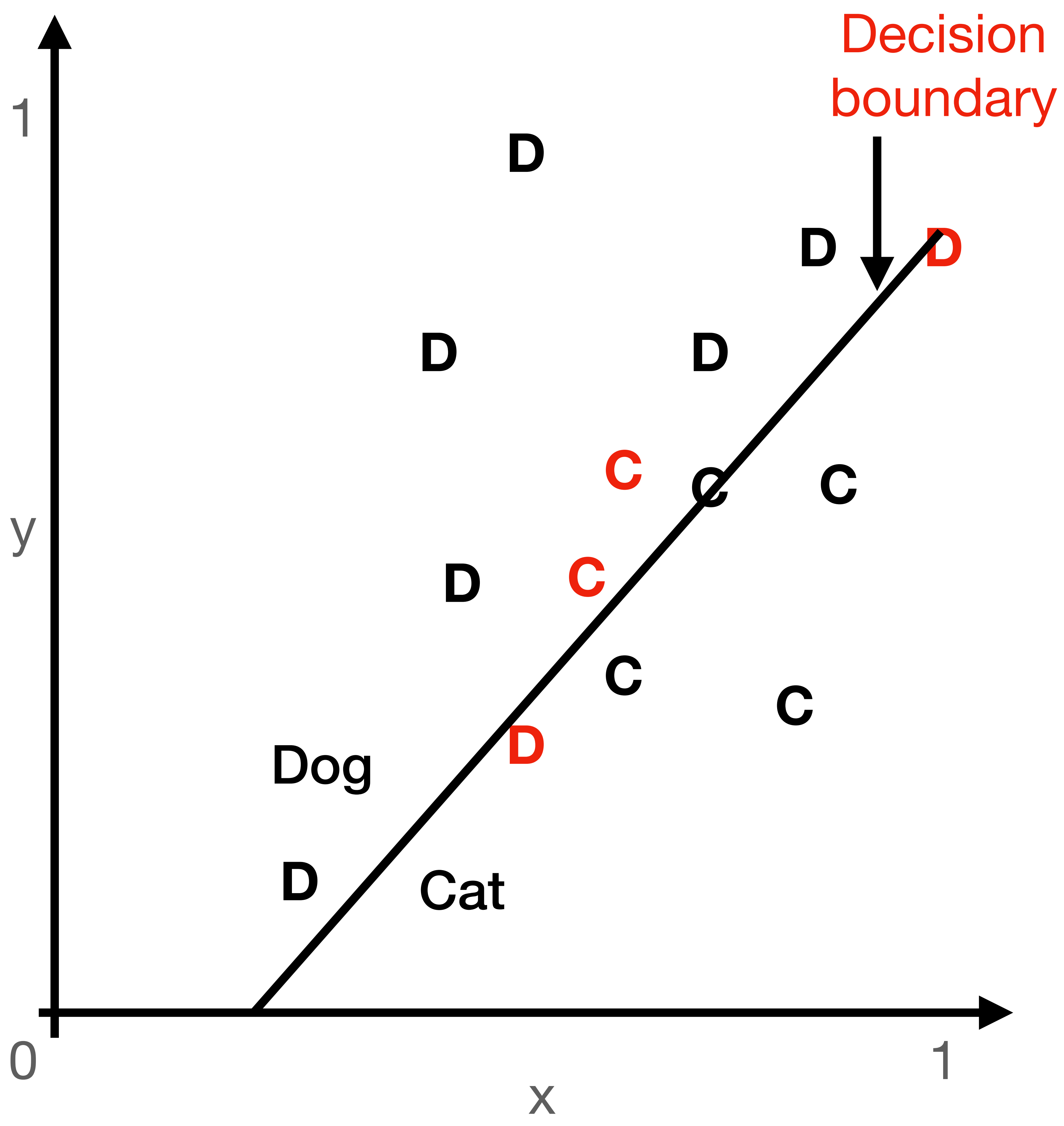


Case study: classification of cat and dog images



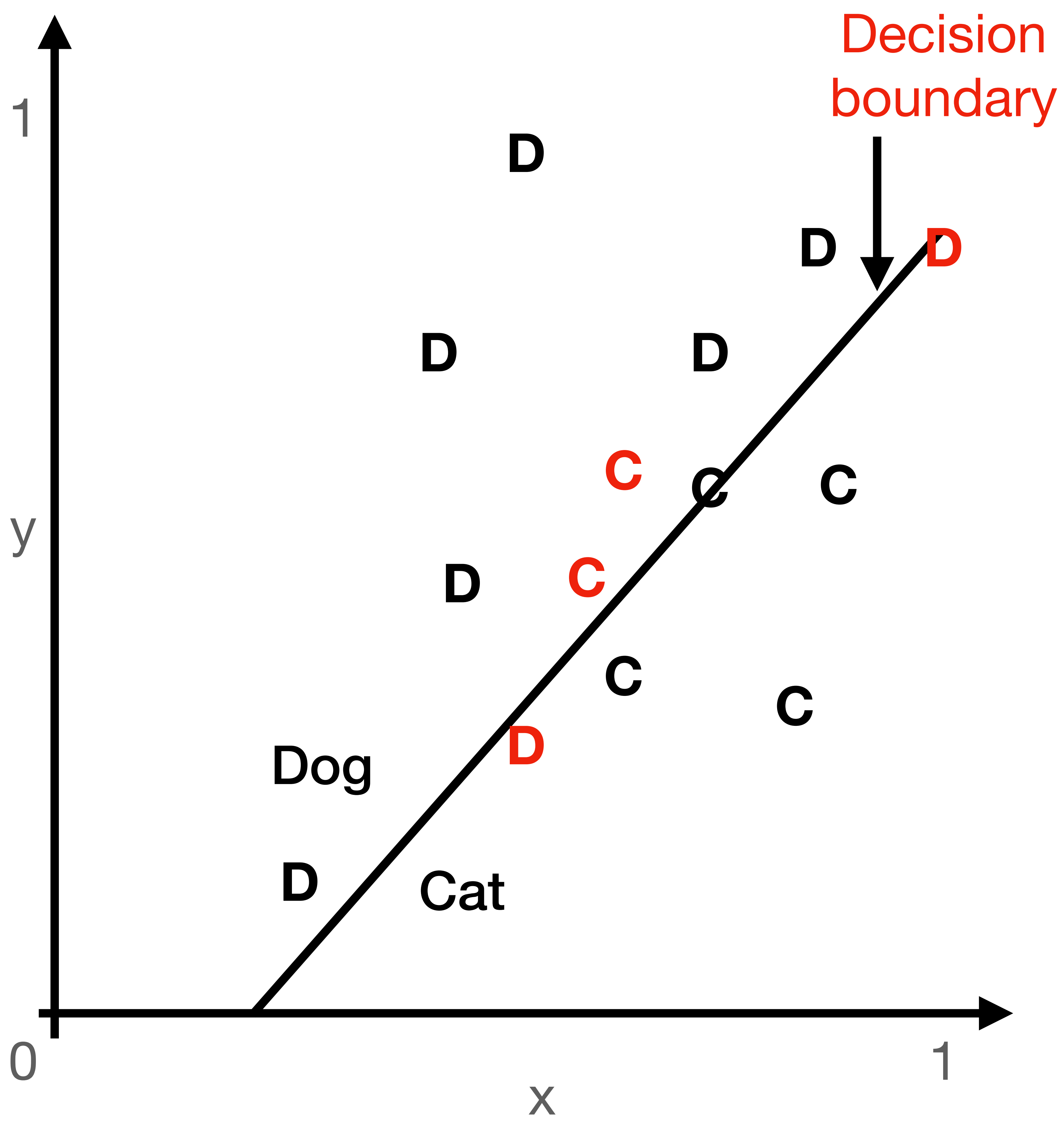
Linear classification

Case study: classification of cat and dog images



Linear classification

Case study: classification of cat and dog images

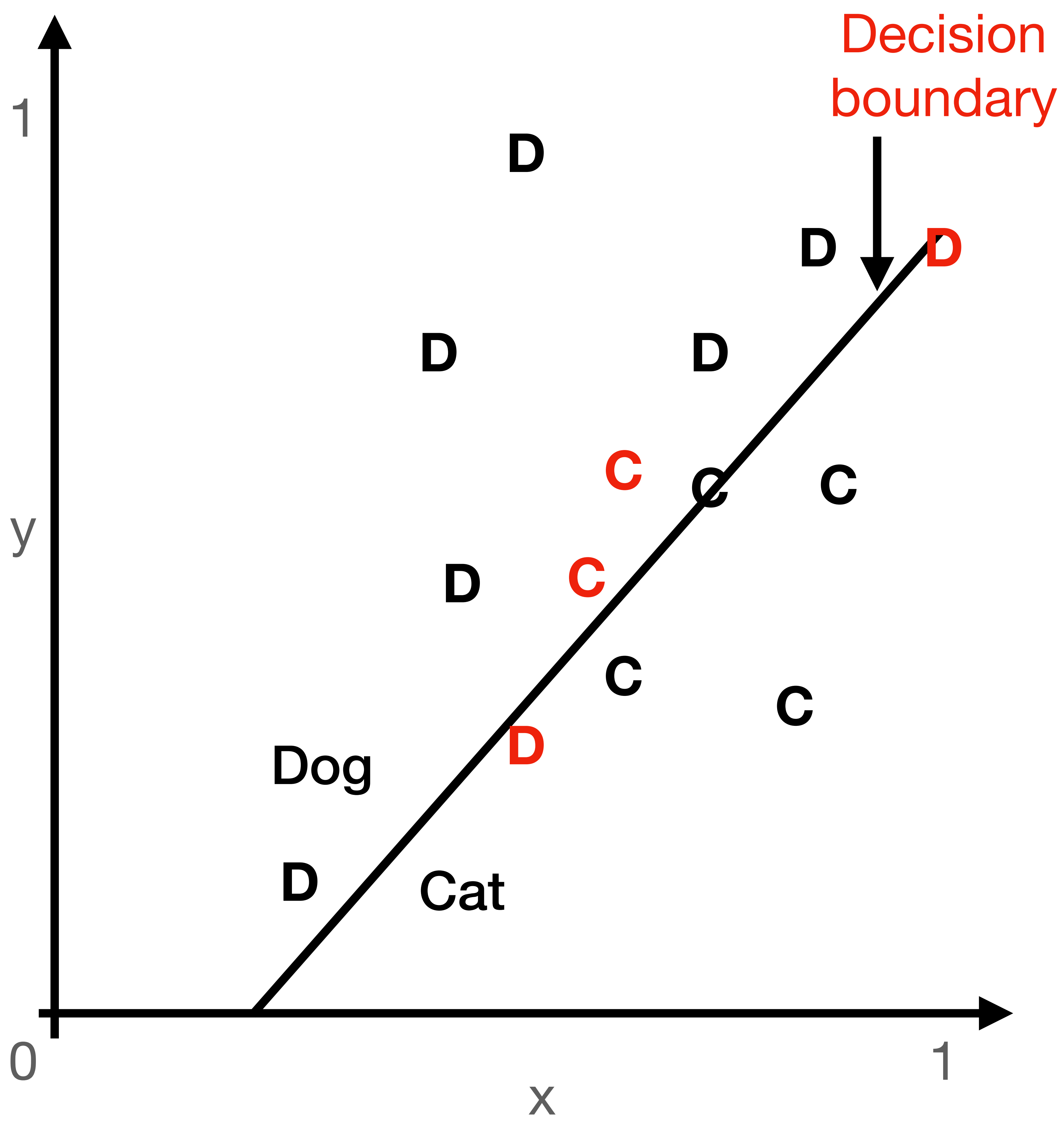


Which line/hyperplane?

"Empirical risk minimisation" idea:
pick the line that minimises classification error on training set

Linear classification

Case study: classification of cat and dog images



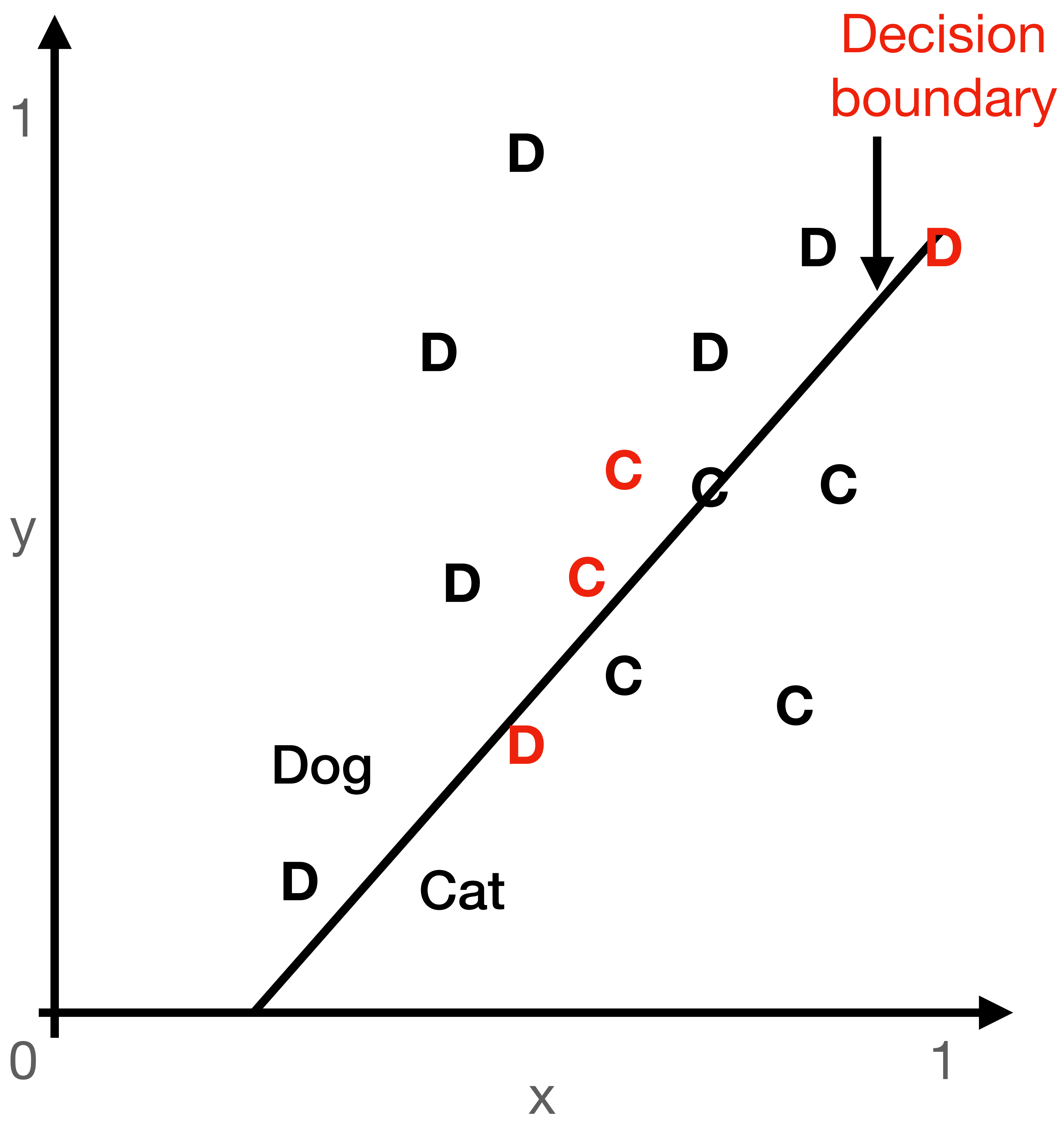
Which line/hyperplane?

“Empirical risk minimisation” idea:
pick the line that minimises
classification error on training set

Problem reduced to an optimisation problem

Linear classification

Case study: classification of cat and dog images

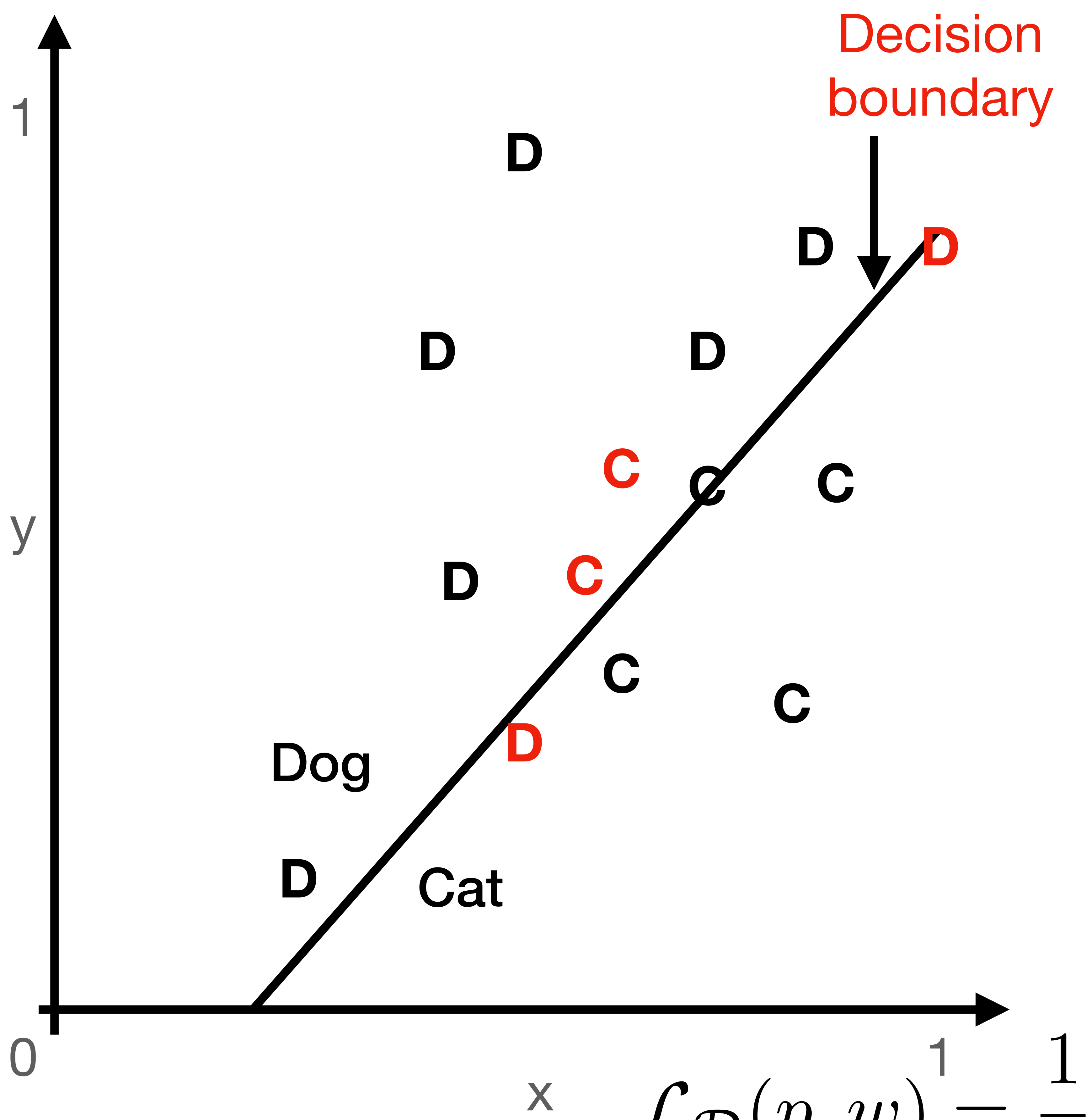


Formalisation of the optimisation problem (d=2)

$$h^* \in \arg\min_{h \in \mathcal{H}} \mathcal{L}_{\mathcal{D}}(h)$$

Linear classification

Case study: classification of cat and dog images



Formalisation of the optimisation problem (d=2)

$$h^* \in \arg\min_{h \in \mathcal{H}} \mathcal{L}_{\mathcal{D}}(h)$$

Training data

$$\mathcal{D} = (v_i, l_i)_{i=1}^n \in (\mathbb{R}^2 \times \{cat, dog\})^n$$

Parametrisation of a line

$$L_{p,w} = \{v \in \mathbb{R}^2 \mid (v-p)^T w = 0\}$$

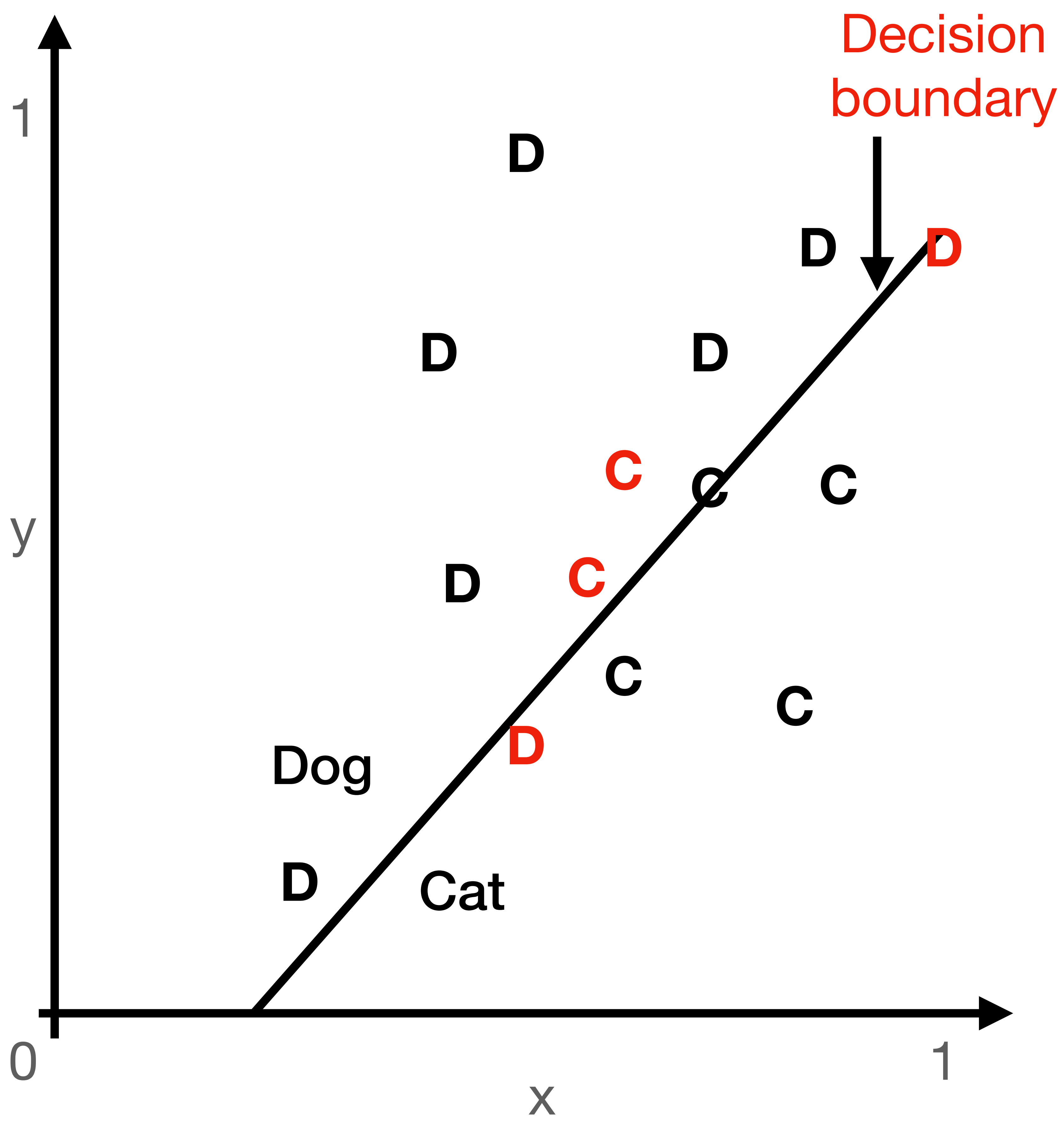
Hypothesis space

$$\mathcal{H} = \{(p, w) \in \mathbb{R}^2 \times (\mathbb{R}^2 \setminus \{(0,0)\})\}$$

Loss function

$$\mathcal{L}_{\mathcal{D}}(p, w) = \frac{1}{n} \sum_{i=1}^n \mathbf{1}_{l_i = \text{cat}} \mathbf{1}_{(v_i - p)^T w > 0} + \mathbf{1}_{l_i = \text{dog}} \mathbf{1}_{(v_i - p)^T w < 0} + \frac{1}{2} \mathbf{1}_{(v_i - p)^T w = 0}$$

Case study: classification of cat and dog images



Search procedure?

- analytic?
- random?
- grid search?

- gradient descent: local search by starting from some point in parameter space and following slope (gradient) of loss function

Linear classification

Case study: classification of cat and dog images

Gradient descent

Gradient

$$\nabla f : x, y \mapsto \left(\frac{\partial f}{\partial x}(x, y), \frac{\partial f}{\partial y}(x, y) \right)$$

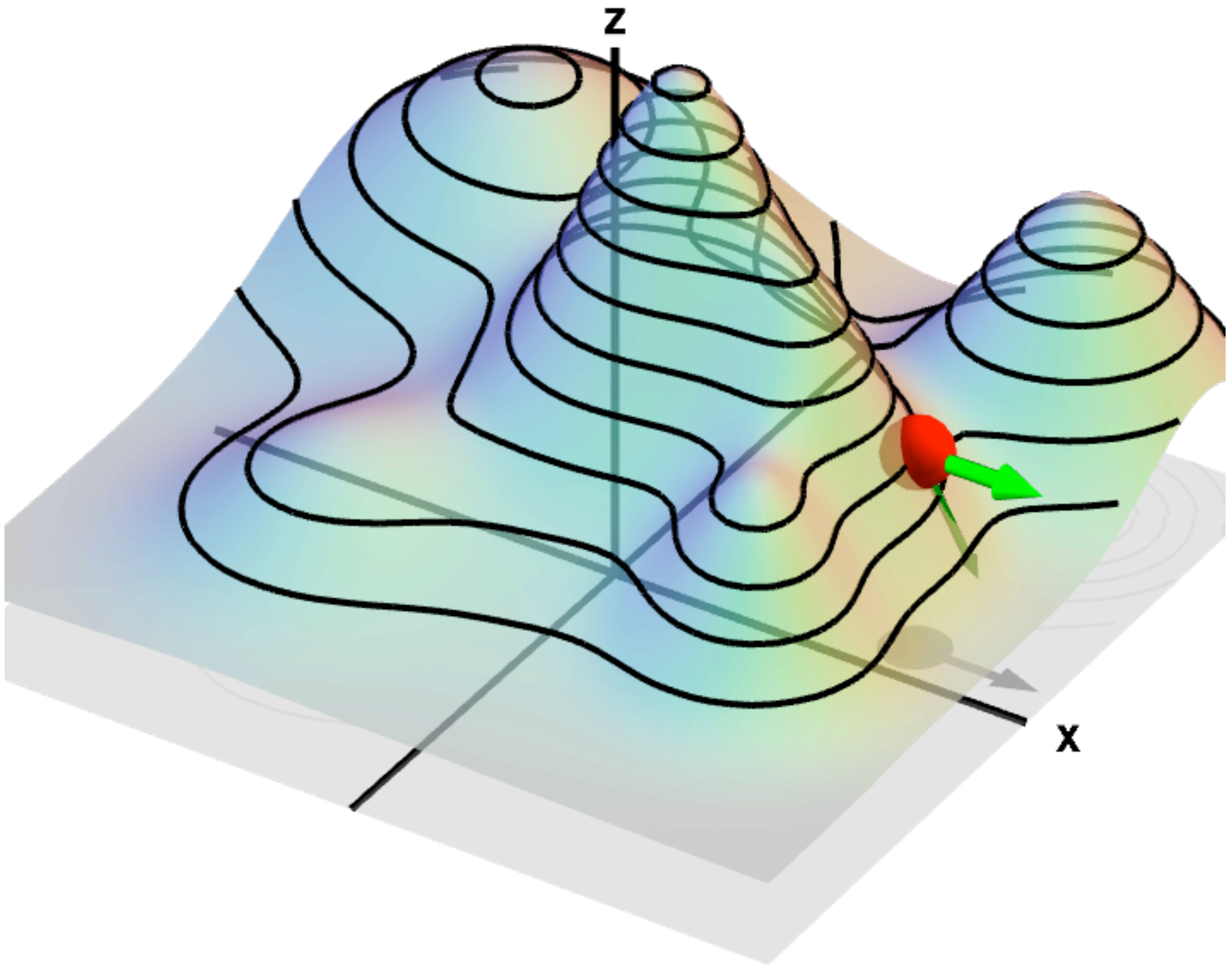


Figure: https://mathinsight.org/directional_derivative_gradient_introduction

Case study: classification of cat and dog images

Loss function

$$\mathcal{L}_{\mathcal{D}}(p,w) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{l_i=\text{cat}}\mathbf{1}_{(v_i-p)^T w>0} + \mathbf{1}_{l_i=\text{dog}}\mathbf{1}_{(v_i-p)^T w<0} + \frac{1}{2}\mathbf{1}_{(v_i-p)^T w=0}$$

$$\Phi(x) = -\log\left(\sigma(x)\right)$$

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

$$\lambda_{\mathcal{D}}(p,w) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{l_i=\text{cat}}\Phi\left(-(v_i-p)^T w\right) + \mathbf{1}_{l_i=\text{dog}}\Phi\left((v_i-p)^T w\right)$$

$$\nabla_w \lambda_{\mathcal{D}}(p,w) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{l_i=\text{cat}}\sigma\left((v_i-p)^T w\right)(v_i-p) - \mathbf{1}_{l_i=\text{dog}}\sigma\left(-(v_i-p)^T w\right)(v_i-p)$$

$$\nabla_p \lambda_{\mathcal{D}}(p,w) = \frac{1}{n}\sum_{i=1}^{n} -\mathbf{1}_{l_i=\text{cat}}\sigma\left((v_i-p)^T w\right)w + \mathbf{1}_{l_i=\text{dog}}\sigma\left(-(v_i-p)^T w\right)w$$

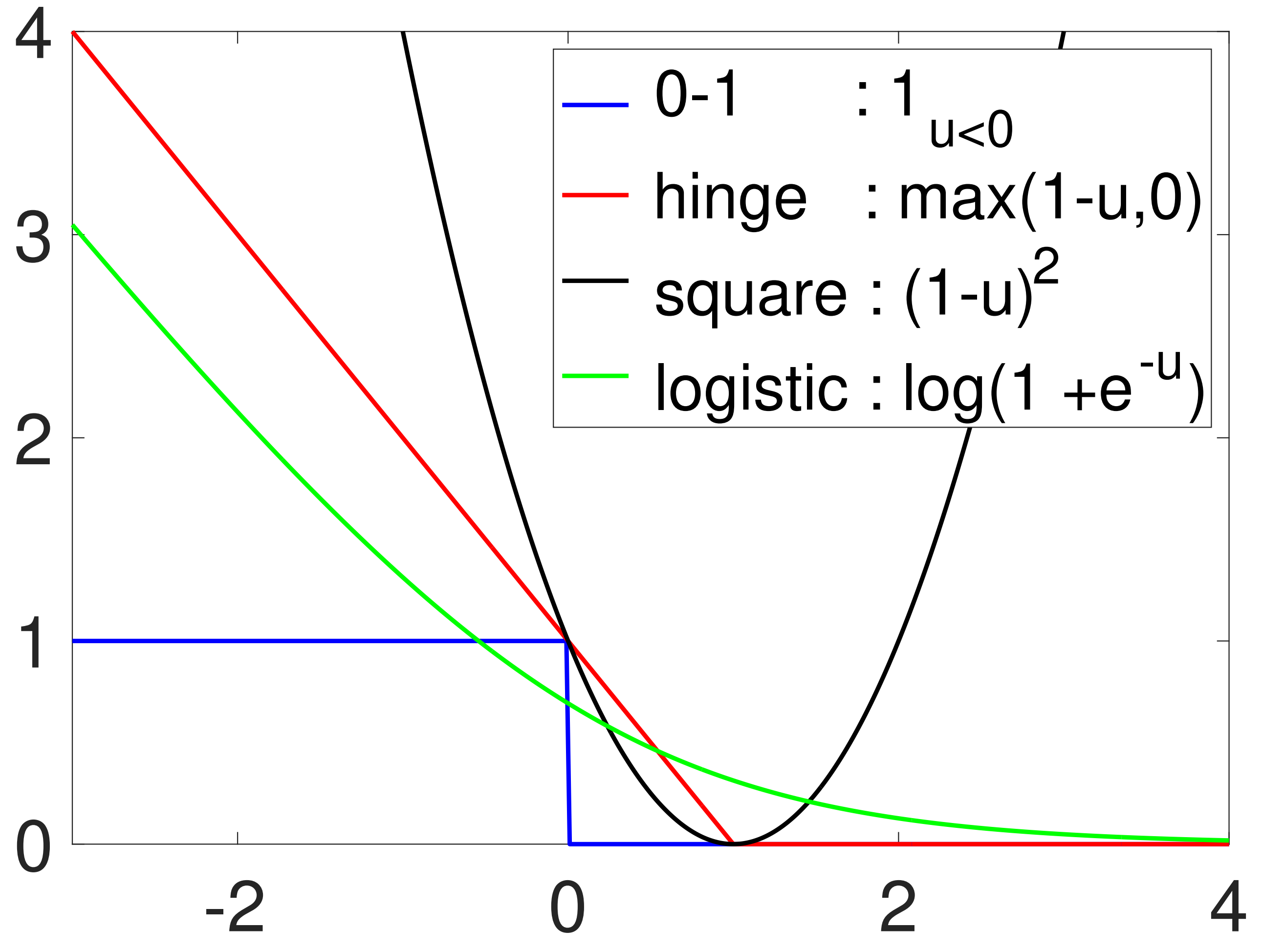


Figure: Learning Theory from First Principles, Bach (forthcoming)

Summary

- Why machine learning may be relevant to cognitive scientists
  - Automatisation of time-consuming tasks
  - Analysis of experimental data
  - Development of models of cognitive and neural processes
- What is machine learning about?
  - Designing and implementing computationally efficient statistical procedures
- Classification of cat and dog images
  - Two simple algorithms for (image) classification: nearest-neighbor and linear classification
  - A simple empirical approach to measuring the statistical and computational efficiency of a classification algorithm

Tomorrow

- Python programming basics to prepare Wednesday's implementation of concepts developed in today's class
- You will need
  - A laptop
  - Access to a Google account to connect to google colab (https://colab.research.google.com/)

Thank you for your attention

Course material will be available after the class at https://thomas.schatz.cogserver.net/teaching/