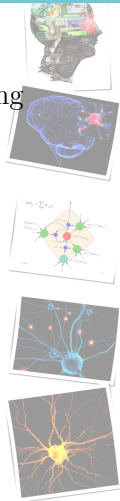


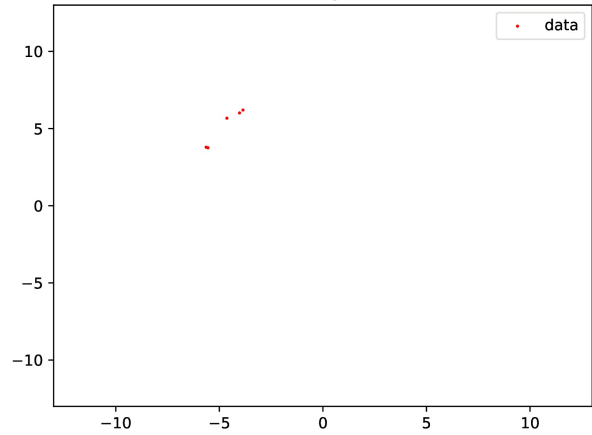
INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP	INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP
<p>A quick introduction to machine learning</p> <p>Spyros Samothrakis Senior Lecturer, IADS University of Essex MiSoC</p> <p>June 22, 2022</p> 						<h2>WELCOME/COURSE CONTENTS</h2> <ul style="list-style-type: none"> <li>What will this course cover? <ul style="list-style-type: none"> <li>Day 1: An intro to machine learning (ML)</li> <li>Day 1: ML labs</li> <li>Day 2: An intro to causal inference</li> <li>Day 2: ML and causal inference labs</li> </ul> </li> <li>Textbooks? <ul style="list-style-type: none"> <li>Mitchell, T. M. (1997). Machine learning.<sup>1</sup></li> <li>Bishop, C. M. (2006). Pattern recognition and machine learning. springer.<sup>2</sup></li> <li>Wasserman, L. (2013). All of statistics: a concise course in statistical inference. Springer Science &amp; Business Media.<sup>3</sup></li> </ul> </li> </ul> <p><sup>1</sup><a href="http://www.cs.cmu.edu/~tom/mlbook.html">http://www.cs.cmu.edu/~tom/mlbook.html</a>  <sup>2</sup><a href="https://www.microsoft.com/en-us/research/publication/pattern-recognition-machine-learning/">https://www.microsoft.com/en-us/research/publication/pattern-recognition-machine-learning/</a>  <sup>3</sup><a href="http://www.stat.cmu.edu/~larry/all-of-statistics/index.html">http://www.stat.cmu.edu/~larry/all-of-statistics/index.html</a></p>					
1 / 56						2 / 56					
INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP	INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP
<h2>BETTER SCIENCE THROUGH DATA</h2> <p>Hey, Tony, Stewart Tansley, and Kristin M. Tolle. “Jim Gray on eScience: a transformed scientific method.” (2009).<sup>4</sup></p> <ul style="list-style-type: none"> <li>Thousand years ago: empirical branch <ul style="list-style-type: none"> <li>You observed stuff and you wrote down about it</li> </ul> </li> <li>Last few hundred years: theoretical branch <ul style="list-style-type: none"> <li>Equations of gravity, equations of electromagnetism</li> </ul> </li> <li>Last few decades: computational branch <ul style="list-style-type: none"> <li>Modelling at the micro level, observing at the macro level</li> </ul> </li> <li>Today: data exploration <ul style="list-style-type: none"> <li>Let machines create models using vast amounts of data</li> </ul> </li> </ul> <p><sup>4</sup><a href="http://languagelog.ldc.upenn.edu/myl/JimGrayOnE-Science.pdf">http://languagelog.ldc.upenn.edu/myl/JimGrayOnE-Science.pdf</a></p>						<h2>BETTER BUSINESS THROUGH DATA</h2> <ul style="list-style-type: none"> <li>There was a report by McKinsey</li> </ul> <p>Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., &amp; Hung Byers, A. (2011). Big data: The next frontier for innovation, competition, and productivity. McKinsey Global Institute.<sup>5</sup></p> <ul style="list-style-type: none"> <li>Urges everyone to monetise “Big Data”</li> <li>Use the data provided within your organisation to gain insights</li> <li>Has some numbers as to how much this is worth</li> <li>Proposes a number of methods, most of them associated with machine learning and databases</li> </ul> <p><sup>5</sup><a href="http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/big-data-the-next-frontier-for-innovation">http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/big-data-the-next-frontier-for-innovation</a></p>					
3 / 56						4 / 56					
INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP	INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP
<h2>WHY IS IT POPULAR NOW?</h2> <ul style="list-style-type: none"> <li><b>Algorithms + data + tools</b></li> <li>Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). Statistical science, 16(3), 199-231.<sup>6</sup></li> <li>Anderson, P. W. (1972). More is different. Science, 177(4047), 393-396.<sup>7</sup></li> <li>Pedregosa, et.al. (2011). Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, 2825-2830.<sup>8</sup></li> </ul> <p><sup>6</sup><a href="http://projecteuclid.org/download/pdf_1/euclid.ss/1009213726%20">http://projecteuclid.org/download/pdf_1/euclid.ss/1009213726%20</a>  <sup>7</sup><a href="https://www.tkm.kit.edu/downloads/TKM1_2011_more_is_different_PWA.pdf">https://www.tkm.kit.edu/downloads/TKM1_2011_more_is_different_PWA.pdf</a>  <sup>8</sup><a href="https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf">https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf</a></p>						<h2>WHAT WILL WE COVER?</h2> <ul style="list-style-type: none"> <li>ML background <ul style="list-style-type: none"> <li><i>Supervised learning</i> <ul style="list-style-type: none"> <li>Regression</li> <li>Classification</li> </ul> </li> <li>Understanding basic modelling</li> <li>Confirming your model is sane</li> <li>Tuning your model</li> <li><b>All within a very applied setting</b></li> </ul> </li> <li>Tools <ul style="list-style-type: none"> <li>Numpy</li> <li>Scikit-learn</li> </ul> </li> </ul>					
5 / 56						6 / 56					

## WHAT IS SUPERVISED LEARNING?

- Imagine someone gives you data from a group of smokers
  - What is their life expectancy?
- We are given inputs  $x_0, x_1 \dots x_n$  and we are looking to predict  $y$
- The problem alludes to certain statistical concepts
- Let's plot some imaginary data

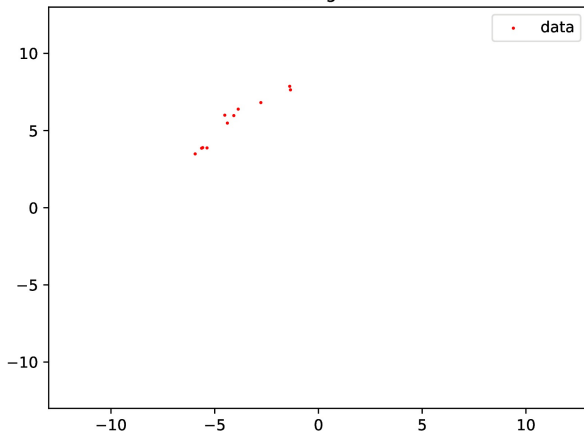
7 / 56

## REGRESSION - LINK THE DOTS (1) training set



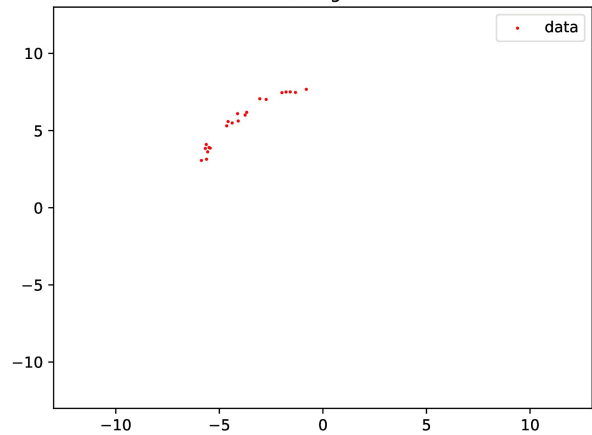
8 / 56

## REGRESSION - LINK THE DOTS (2) training set



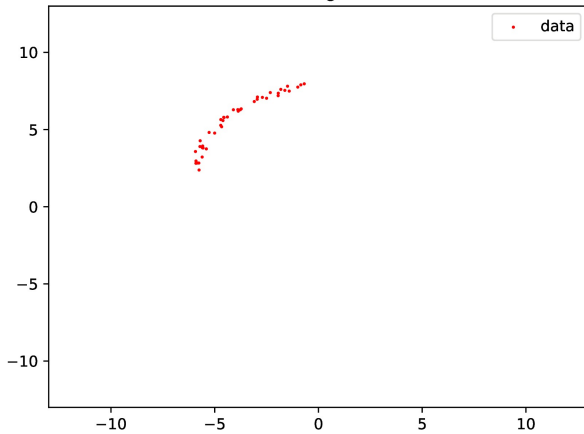
9 / 56

## REGRESSION - LINK THE DOTS (3) training set



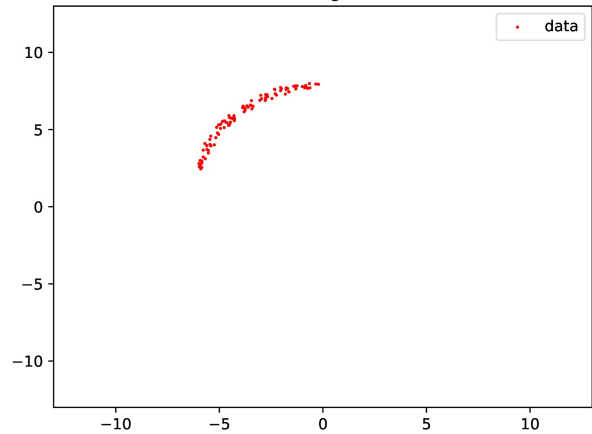
10 / 56

## REGRESSION - LINK THE DOTS (4) training set

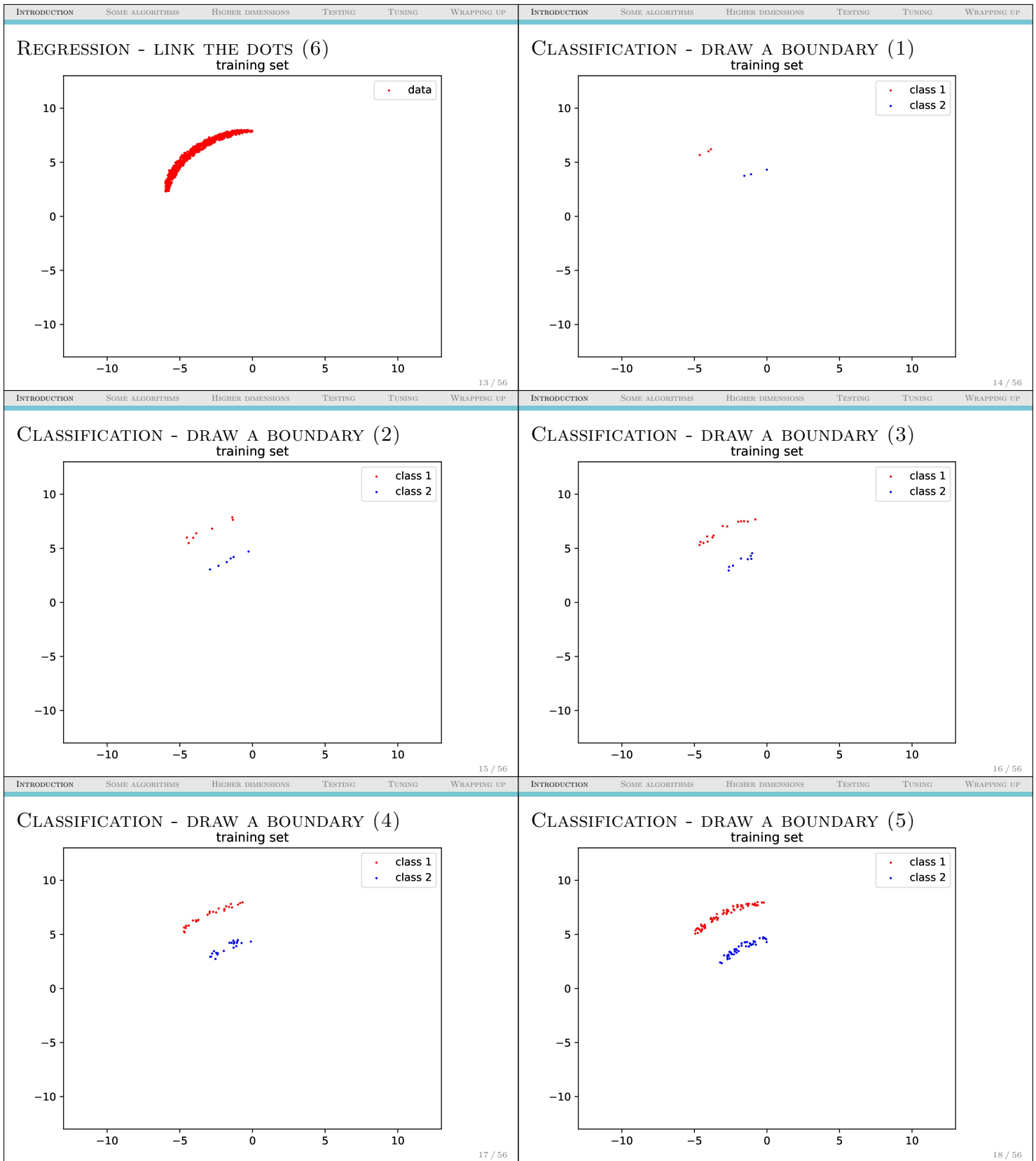


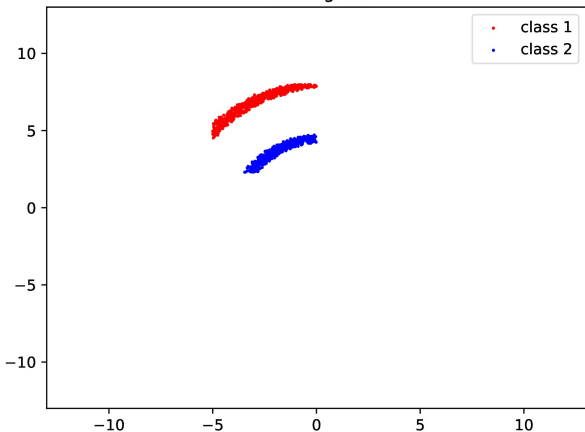
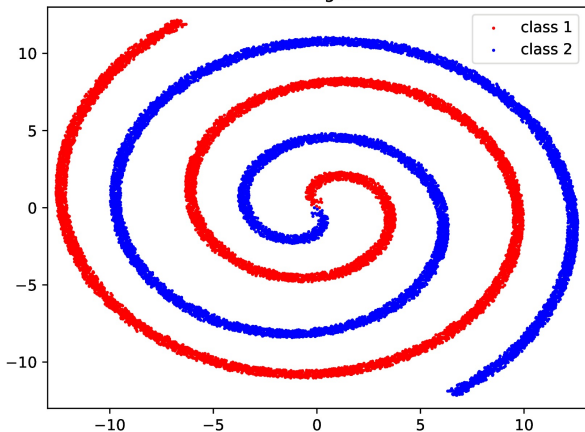
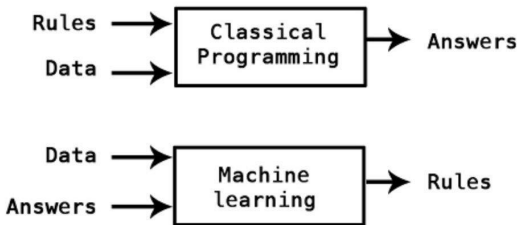
11 / 56

## REGRESSION - LINK THE DOTS (5) training set

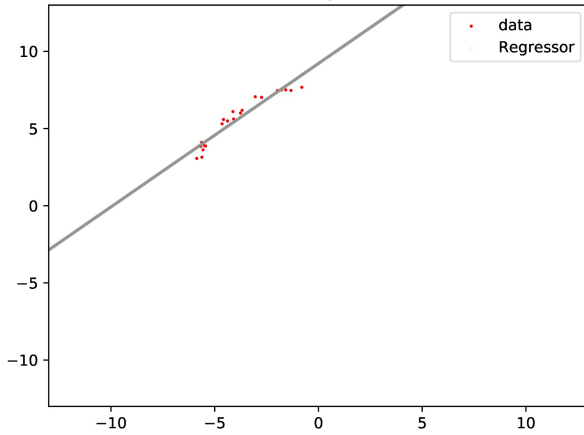


12 / 56



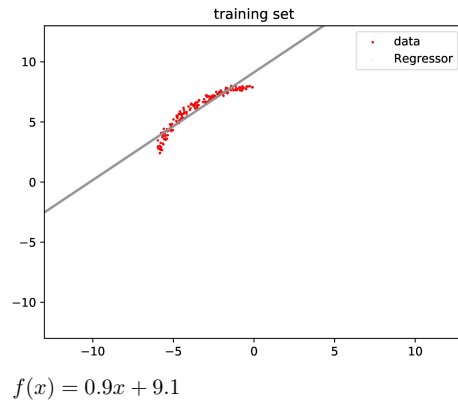
INTRODUCTION SOME ALGORITHMS HIGHER DIMENSIONS TESTING TUNING WRAPPING UP	INTRODUCTION SOME ALGORITHMS HIGHER DIMENSIONS TESTING TUNING WRAPPING UP
<div data-bbox="99 275 643 323"> <h3>CLASSIFICATION - DRAW A BOUNDARY (6)</h3> <p>training set</p> </div> <div data-bbox="120 323 701 751">  </div> <div data-bbox="753 753 797 770"> 19 / 56 </div>	<div data-bbox="829 275 1218 323"> <h3>FULL DATA</h3> <p>training set</p> </div> <div data-bbox="850 323 1432 751">  </div> <div data-bbox="1479 753 1523 770"> 20 / 56 </div>
<div data-bbox="99 821 282 852"> <h3>INTUITION (1)</h3> </div> <div data-bbox="125 894 735 947"> <p>Chollet, F. (2018). Deep learning with Python (Vol. 361). New York: Manning.<sup>9</sup></p> </div> <div data-bbox="152 974 665 1197">  </div> <div data-bbox="147 1276 675 1304"> <p><sup>9</sup><a href="https://www.manning.com/books/deep-learning-with-python">https://www.manning.com/books/deep-learning-with-python</a></p> </div> <div data-bbox="753 1302 797 1318"> 21 / 56 </div>	<div data-bbox="829 821 1013 852"> <h3>INTUITION (2)</h3> </div> <div data-bbox="875 963 1498 1142"> <ul style="list-style-type: none"> <li>▶ That's it - we are given data, and we need to come up with an algorithm to join it up – but in high dimensions <ul style="list-style-type: none"> <li>▶ Can be binary, categorical, real-valued - more on this later</li> </ul> </li> <li>▶ How well a function joins the data is called the “loss”</li> <li>▶ Multiple solutions exist, so a loss function must take into account concepts other than pure fit</li> </ul> </div> <div data-bbox="1479 1302 1523 1318"> 22 / 56 </div>
<div data-bbox="99 1367 292 1398"> <h3>Vs CAUSALITY</h3> </div> <div data-bbox="144 1505 768 1705"> <ul style="list-style-type: none"> <li>▶ Imagine someone gives you data from a group of smokers <ul style="list-style-type: none"> <li>▶ What is their life expectancy?</li> <li>▶ <b>Is smoking bad for you?</b></li> </ul> </li> <li>▶ You could potentially just do predictions using correlations <ul style="list-style-type: none"> <li>▶ What if there was a gene that caused early death and also made you like smoking?</li> <li>▶ More on this tomorrow</li> </ul> </li> </ul> </div> <div data-bbox="753 1848 797 1864"> 23 / 56 </div>	<div data-bbox="829 1367 1091 1398"> <h3>LINEAR REGRESSION</h3> </div> <div data-bbox="875 1533 1266 1659"> <ul style="list-style-type: none"> <li>▶ Linear and logistic regression <ul style="list-style-type: none"> <li>▶ Logistic regression does classification</li> </ul> </li> <li>▶ You just assume everything is a line</li> <li>▶ <math>f(x) = wx + b</math></li> </ul> </div> <div data-bbox="1479 1848 1523 1864"> 24 / 56 </div>

EXAMPLE (LINEAR REGRESSION)  
training set



25 / 56

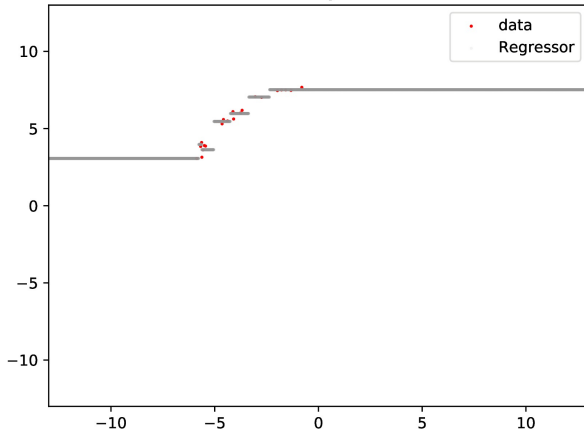
EXAMPLE (LINEAR REGRESSION)



$$f(x) = 0.9x + 9.1$$

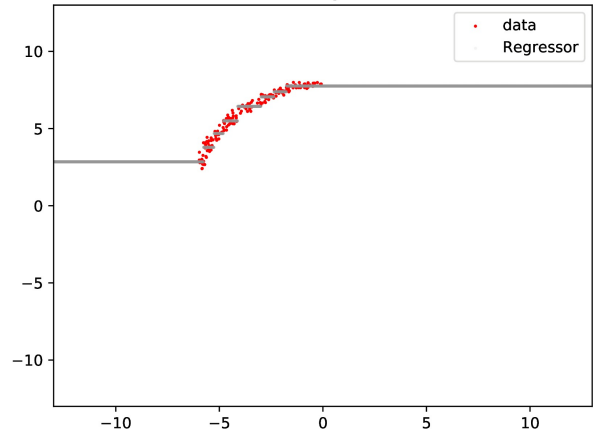
26 / 56

EXAMPLE (DECISION TREE)  
training set



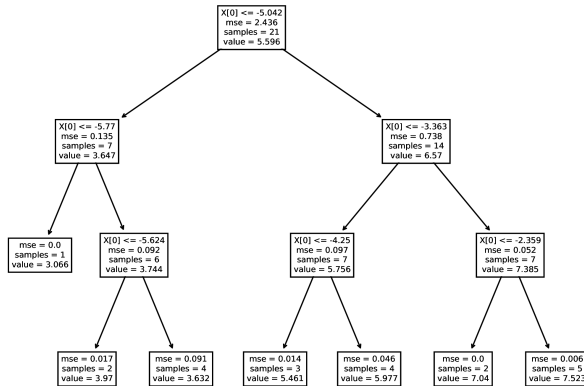
27 / 56

EXAMPLE (DECISION TREE)  
training set



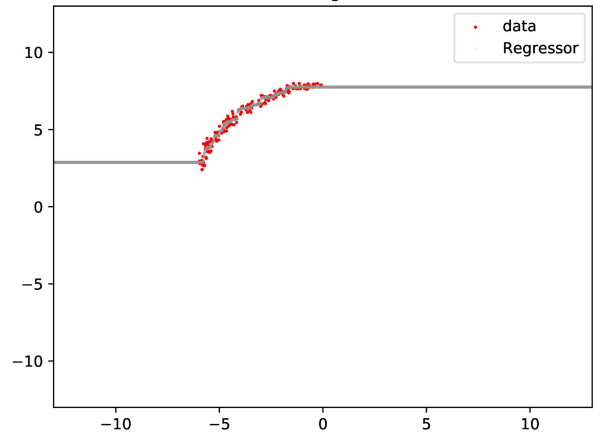
28 / 56

EXAMPLE (DECISION TREE — INTERNAL)

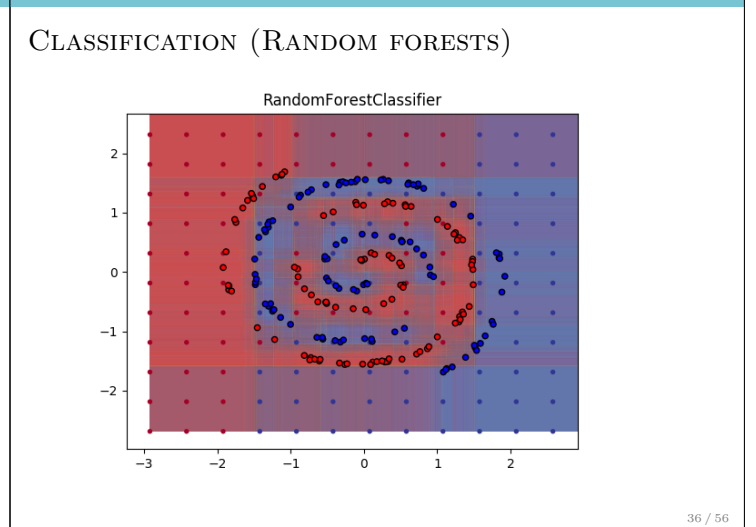
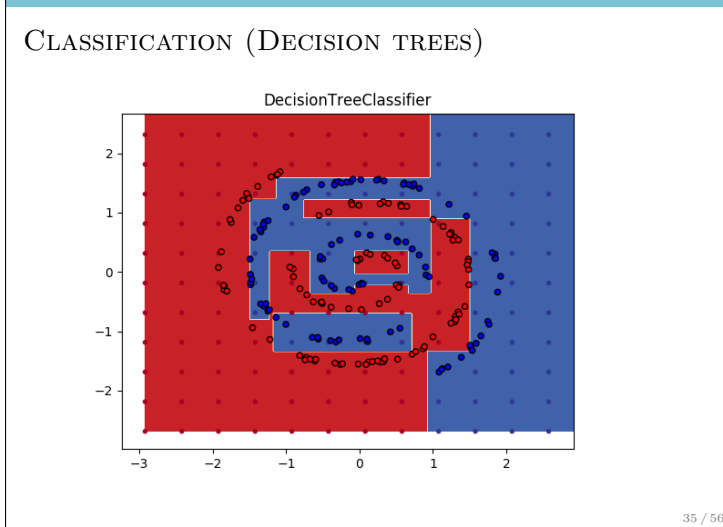
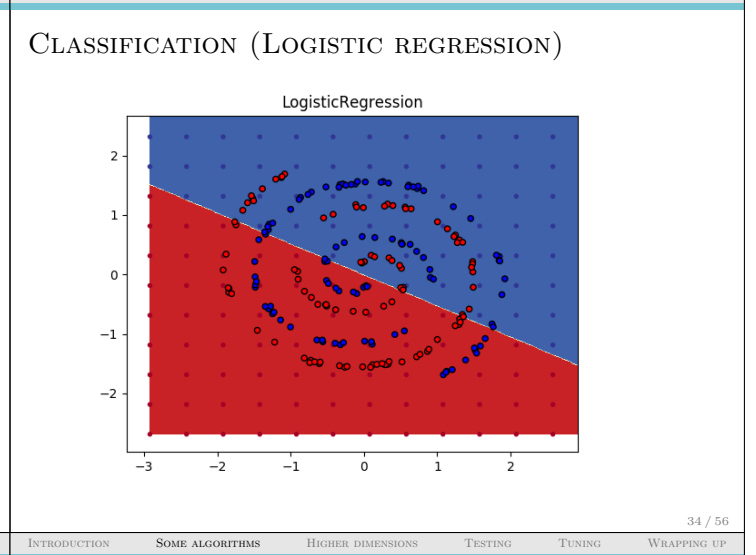
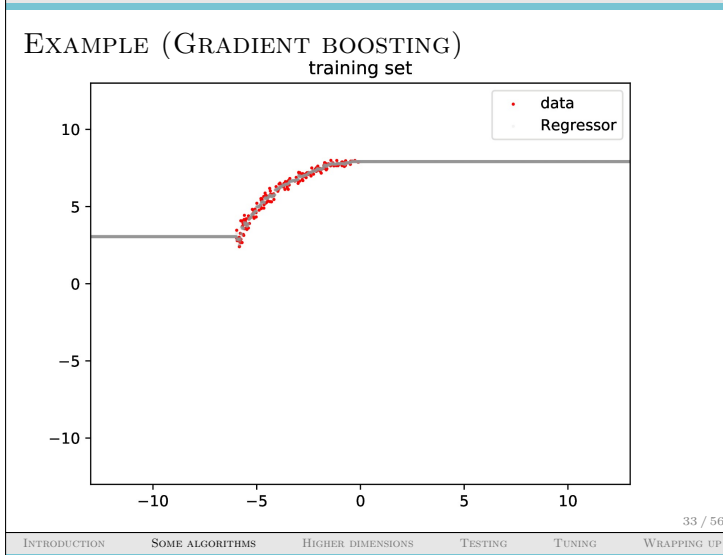
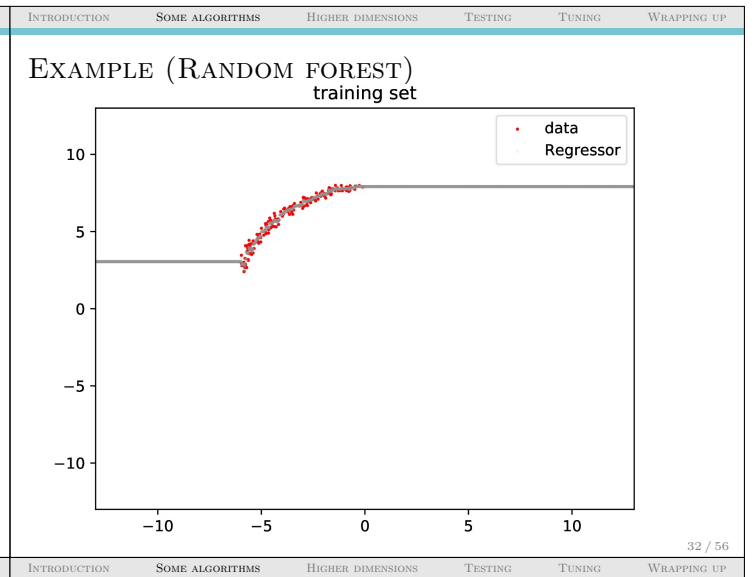
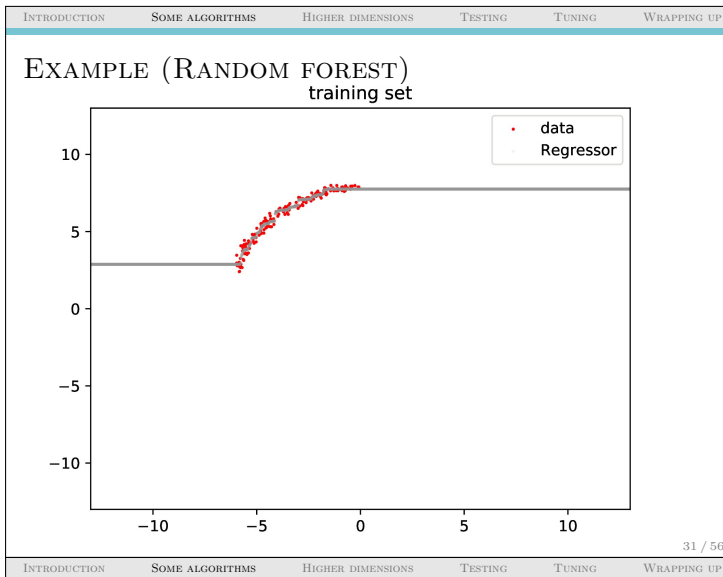


29 / 56

EXAMPLE (RANDOM FOREST)  
training set



30 / 56



## DATA DIMENSIONALITY

- Until now we have seen input data of 1 (for regression) or two (for classification) dimensions
- How about higher dimensional data?
  - Some times data can have millions of features
- Let's examine more high dimensional dataset
- Visualisation becomes harder

37 / 56

## DIABETES CLASSIFICATION

Feature	Description
$X_0$	Pregnancies: Number of times pregnant
$X_1$	Glucose: Plasma glucose concentration
$X_2$	BloodPressure: Diastolic blood pressure (mm Hg)
$X_3$	SkinThickness: Tripe skin fold thickness (mm)
$X_4$	Insulin: 2-Hour serum insulin (mU/ml)
$X_5$	BMI: Body mass index (weight in kg/(height in m) <sup>2</sup> )
$X_6$	DiabetesPedigreeFunction: Diabetes pedigree function
$X_7$	Age: Age (years)
$y$	Outcome: Has diabetes (0 or 1)

<https://www.kaggle.com/mathchi/diabetes-data-set>

38 / 56

## HOW DOES THE DATA LOOK LIKE?

Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DPF	Age	
0	6	148	72	35	0	33.60	0.63	50
1	1	85	66	29	0	26.60	0.35	31
2	8	183	64	0	0	23.30	0.67	32
3	1	89	66	23	94	28.10	0.17	21
4	0	137	40	35	168	43.10	2.29	33

y

0 1

1 0

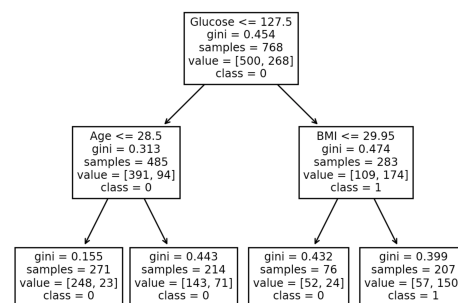
2 1

3 0

4 1

39 / 56

## DECISION TREE



40 / 56

## DIABETES REGRESSION

Efron, B., Hastie, T., Johnstone, I., & Tibshirani, R. (2004). Least angle regression. *Annals of statistics*, 32(2), 407-499.<sup>10</sup>

Feature	Description
$X_0$	age in years
$X_1$	sex
$X_2$	bmi body mass index
$X_3$	bp average blood pressure
$X_4$	s1 tc, total serum cholesterol
$X_5$	s2 ldl, low-density lipoproteins
$X_6$	s3 hdl, high-density lipoproteins
$X_7$	s4 tc, total cholesterol / HDL
$X_8$	s5 lg, possibly log of serum triglycerides level
$X_9$	s6 glu, blood sugar level
$y$	disease progression one year after baseline

<sup>10</sup>[https://scikit-learn.org/stable/datasets/toy\\_dataset.html#diabetes-dataset](https://scikit-learn.org/stable/datasets/toy_dataset.html#diabetes-dataset)

41 / 56

## LET'S SEE THE REAL DATA VALUES

	age	sex	bmi	bp	s1	s2	s3	s4	s5	s6
0	0.04	0.05	0.06	0.02	-0.04	-0.03	-0.04	-0.00	0.02	-0.02
1	-0.00	-0.04	-0.05	-0.03	-0.01	-0.02	0.07	-0.04	-0.07	-0.09
2	0.09	0.05	0.04	-0.01	-0.05	-0.03	-0.03	-0.00	0.00	-0.03
3	-0.09	-0.04	-0.01	-0.04	0.01	0.02	-0.04	0.03	0.02	-0.01
4	0.01	-0.04	-0.04	0.02	0.00	0.02	0.01	-0.00	-0.03	-0.05

“Note: Each of these 10 feature variables have been mean centered and scaled by the standard deviation times `n_samples` (i.e. the sum of squares of each column totals 1).”

	y
0	151.00
1	75.00
2	141.00
3	206.00
4	135.00

42 / 56

INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP	INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP
<h2>LINEAR REGRESSION</h2> $y = -210x_0 - 5036x_1 + 10916x_2 + 6812x_3 - 16635x_4 + 10011x_5 + 2121x_6 + 3718x_7 + 15776x_8 + 1420x_9 + 152$						<h2>QUALITY ASSESSMENT</h2> <ul style="list-style-type: none"> <li>▶ In lower dimensions, the visualisations we did provided some insights to the quality of our methods <ul style="list-style-type: none"> <li>▶ This is impossible in higher dimensions</li> </ul> </li> <li>▶ We need to measure some kind of metric that denotes quality of fit</li> </ul>					
43 / 56						44 / 56					
INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP	INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP
<h2>METRICS</h2> <ul style="list-style-type: none"> <li>▶ For regression, <ul style="list-style-type: none"> <li>▶ Mean Squared Error</li> <li>▶ Mean Absolute Error</li> </ul> </li> <li>▶ For classification <ul style="list-style-type: none"> <li>▶ Accuracy</li> <li>▶ Mean Squared Error</li> <li>▶ Cross-entropy loss</li> <li>▶ AUC</li> </ul> </li> <li>▶ Each one has different benefits, e.g. absolute errors tend to be more robust to outliers</li> </ul>						<h2>ACCURACY</h2> <ul style="list-style-type: none"> <li>▶ Our model is <math>\hat{f}(x)</math>, <math>x</math> are examples, <math>y</math> is outcome</li> <li>▶ Accuracy is the obvious one <ul style="list-style-type: none"> <li>▶ <math>accuracy = \frac{1}{N} \sum_{i=0}^{N-1} (y_i = \hat{f}(x))</math></li> <li>▶ The higher the accuracy the better</li> </ul> </li> <li>▶ What if the dataset is unbalanced - how informative is accuracy then?</li> <li>▶ There are multiple score functions <ul style="list-style-type: none"> <li>▶ Use the one appropriate for your problem</li> </ul> </li> </ul>					
45 / 56						46 / 56					
INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP	INTRODUCTION	SOME ALGORITHMS	HIGHER DIMENSIONS	TESTING	TUNING	WRAPPING UP
<h2>MEAN SQUARED ERROR (MSE)</h2> <ul style="list-style-type: none"> <li>▶ Our model is <math>\hat{f}(x)</math>, <math>x</math> are examples, <math>y</math> is outcome <ul style="list-style-type: none"> <li>▶ <math>MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{f}(x_i))^2</math></li> </ul> </li> </ul>						<h2>TRAIN/VALIDATION/TEST SPLIT</h2> <ul style="list-style-type: none"> <li>▶ Basic idea: split your data into three portions</li> <li>▶ 1. train, you used that to train your classifier/regressor</li> <li>▶ 2. validation, you use that to assess the quality of your method, retraining as you see fit</li> <li>▶ 3. test, you report results on this</li> <li>▶ Common split is 60%/20%/20%</li> </ul>					
47 / 56						48 / 56					



INTRODUCTION

SOME ALGORITHMS

HIGHER DIMENSIONS

TESTING

TUNING

WRAPPING UP

CROSS VALIDATION

► How about we split our data into multiple validation sets and find the mean?

► Instead of having just one split train/test split, we can have multiple

► Colloquially goes by names like 5-fold CV, 10-fold CV

► There are multiple ways of doing the sampling to create training/validation sets, we will focus on only one

49 / 56

INTRODUCTION

SOME ALGORITHMS

HIGHER DIMENSIONS

TESTING

TUNING

WRAPPING UP

PICTORIAL DEPICTION OF 5-FOLD CV

Copied from SKlearns website<sup>11</sup>

<sup>11</sup>[https://scikit-learn.org/stable/\\_images/grid\\_search\\_cross\\_validation.png](https://scikit-learn.org/stable/_images/grid_search_cross_validation.png)

50 / 56

INTRODUCTION

SOME ALGORITHMS

HIGHER DIMENSIONS

TESTING

TUNING

WRAPPING UP

WHY TUNE?

► Your data has peculiarities

► You need to “calibrate” your algorithm with these peculiarities

► Tuning properly will have a significant effect on cross-validation scores

► ... and hence the quality of learning

51 / 56

INTRODUCTION

SOME ALGORITHMS

HIGHER DIMENSIONS

TESTING

TUNING

WRAPPING UP

HYPERPARAMETERS

► Called hyperparameters (vs parameters) as they influence how the modelling is done (vs the direct modeling)

► How many trees?

► Tree depth?

► Maximum tree size

► l2 regularisation?

► vs parameters (e.g. weights in linear regression)

52 / 56

INTRODUCTION

SOME ALGORITHMS

HIGHER DIMENSIONS

TESTING

TUNING

WRAPPING UP

WE NEED TO LOOK FOR OPTIMAL PARAMETERS

► Computationally expensive

► We can do this either by searching both the classifier/regressor space and their parameters

► Grid search

► More than one parameter, we exhaustively search

53 / 56

INTRODUCTION

SOME ALGORITHMS

HIGHER DIMENSIONS

TESTING

TUNING

WRAPPING UP

EXAMPLE USING LINEAR REGRESSION

alpha	scores	mean	std
0.0001	[2782, 3032, 3226, 3003, 2917]	2992.1772	145.5645
0.0001	[2783, 3032, 3223, 3002, 2920]	2992.0154	143.9139
0.0002	[2785, 3032, 3218, 3001, 2923]	2991.8400	141.7267
0.0007	[2812, 3042, 3186, 3002, 2945]	2997.5634	122.1458
0.0009	[2818, 3042, 3179, 2992, 2946]	2995.3784	117.9862
0.0012	[2827, 3043, 3178, 2978, 2947]	2994.6426	115.5067
0.0037	[2884, 3060, 3190, 2895, 2968]	2999.3816	114.1540
0.0049	[2918, 3079, 3201, 2869, 2985]	3010.3321	118.4097
0.0065	[2938, 3111, 3215, 2856, 3017]	3027.3294	126.2295
0.0085	[2966, 3152, 3219, 2859, 3057]	3050.5713	128.2733
0.0113	[3014, 3212, 3236, 2872, 3113]	3089.2555	134.1712
0.0149	[3028, 3292, 3279, 2918, 3201]	3143.7112	146.9126
0.0196	[3040, 3366, 3358, 2970, 3289]	3204.6848	166.7447
0.0259	[3082, 3493, 3484, 3074, 3435]	3313.4750	193.2530
0.0342	[3206, 3706, 3681, 3237, 3678]	3501.7398	229.0676
0.0452	[3434, 4030, 3972, 3448, 4037]	3784.1217	281.4318
0.0597	[3801, 4573, 4447, 3745, 4545]	4222.0278	369.6680
0.0788	[4401, 5460, 5212, 4299, 5425]	4959.4742	505.7819
0.1040	[5211, 6521, 6262, 5200, 6486]	5935.8770	603.2078
0.1374	[5353, 6521, 6262, 5290, 6486]	5982.4134	547.2524

54 / 56

## WHAT DO YOU OBSERVE?

- ▶ Properly tuning your model can have a huge impact!

## WRAPPING UP

- ▶ You get data from somewhere
- ▶ ML will help you predict certain targets
- ▶ Data can be noisy
- ▶ You might need to pre-process it
- ▶ The more data the better
- ▶ Choosing the right classifier/regressor is important
  - ▶ Cross-validate and test