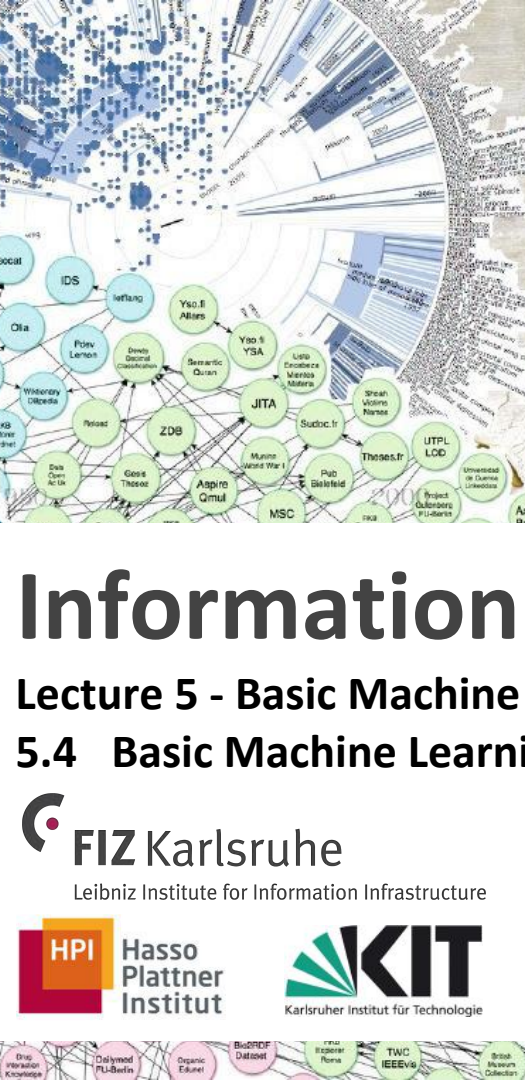




5.4 Basic Machine Learning Algorithms



HPI Hasso Plattner Institut

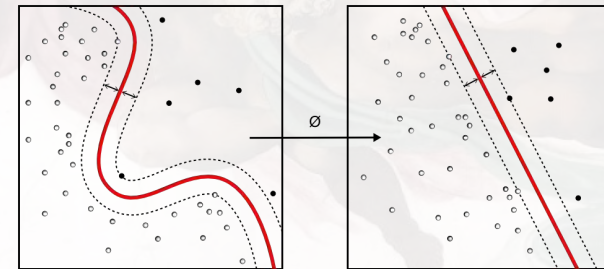


Leibniz Institute for Information Infrastructure
Karlsruhe Institute of Technology
Spring 2018

Information Service Engineering

Lecture 5: Basic Machine Learning

- 5.1 A Brief History of AI
- 5.2 Introduction to Machine Learning
- 5.3 Evaluation and Generalization Problems
- 5.4 Basic ML Algorithms**
- 5.5 Basic ML Algorithms II
- 5.6 Unsupervised Learning
- 5.7 Deep Learning



No Free Lunch Theorem

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

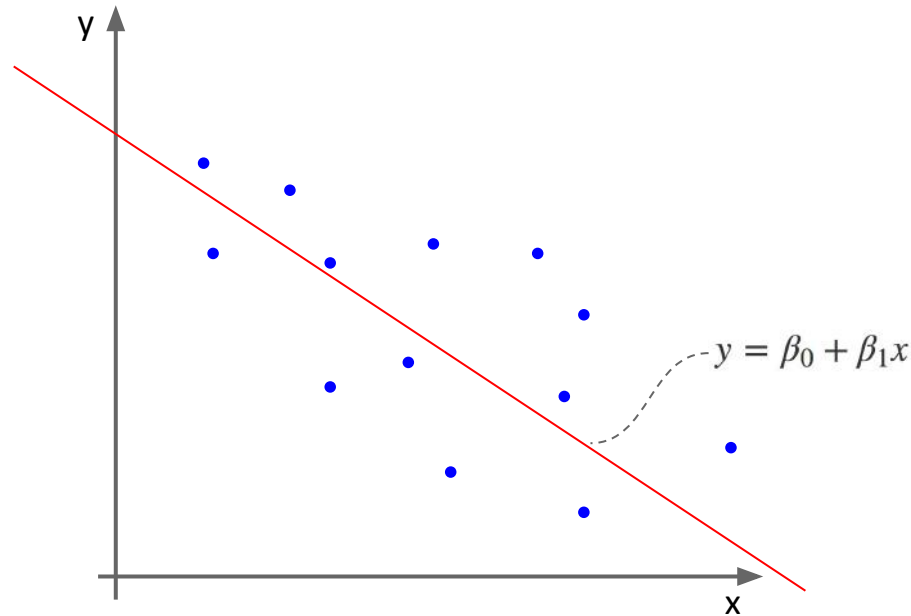
- A **model** is a simplified version of the **observations**
- **Simplifications** are meant to discard superfluous details that are unlikely to generalize
- To decide about discarding observations, we must make **assumptions**
- If you make no assumption about the data, then there is no reason to prefer one model over another
- **Therefore:** make reasonable assumptions about the data and restrict the evaluation to a few reasonable models
 - *e.g. for simple tasks evaluate linear models*



Linear Regression

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- With **Linear Regression** we aim to fit a line to a scattering of data



Linear Regression

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- Notation:

- Input vector $x \in \mathbb{R}^N$
- Output vector $y \in \mathbb{R}$
- Parameters $\beta = (\beta_0, \beta_1, \dots, \beta_N)^T \in \mathbb{R}$

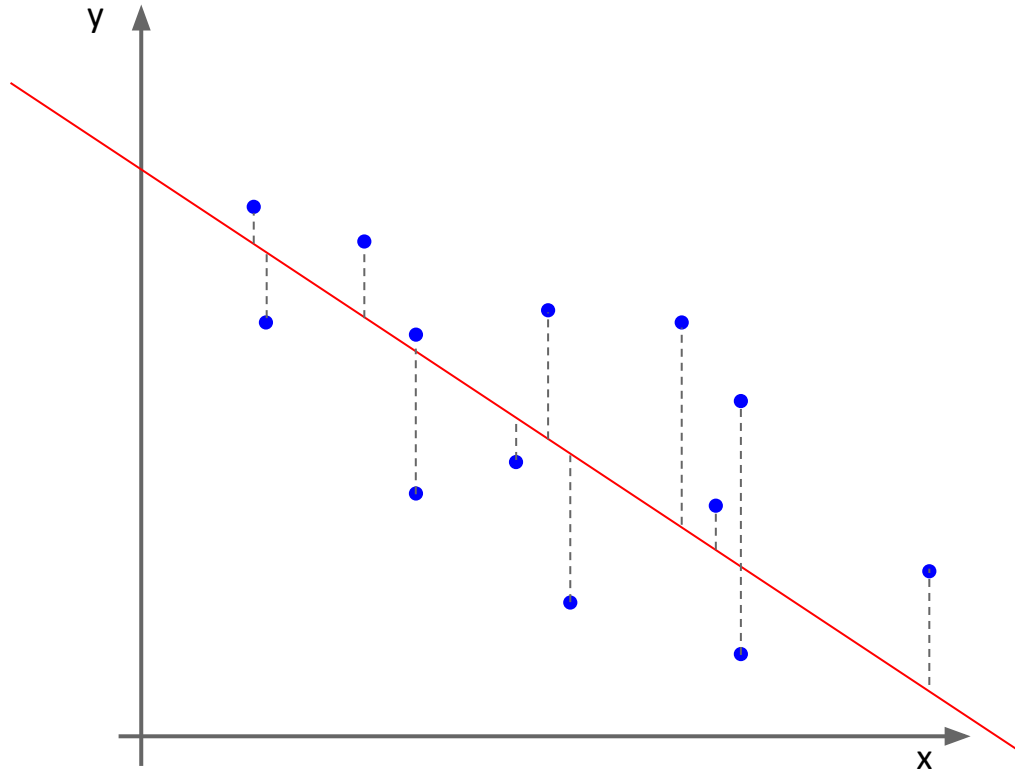
- Linear model:
$$f(x) = \beta_0 + \sum_{j=1}^N \beta_j x_j$$

- Given training data $D = \{(x_i, y_i)\}_{i=1}^P$

- we define the least squares costs (or “loss”)
$$L^{ls}(\beta) = \sum_{i=1}^P (y_i - f(x_i))^2$$

Least Square Costs

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms



- Try to find the line (hyperplane) that fits best to the given data by minimizing its distance between the data and the line
- The **Least Squares Cost ($L^{\text{ls}}(\beta)$)** framework aims at recovering the line (hyperplane) that minimized the total squared length of the error

$$L^{\text{ls}}(\beta) = \sum_{i=1}^P (y_i - f(x_i))^2$$

Minimizing the Least Square Costs

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- Find optimal parameters β

- Augment input vector with a 1 in front: $\mathbf{x} = (1, x) = (1, x_1, x_2, \dots, x_N)^\top \in \mathbb{R}^{N+1}$

$$\beta = (\beta_0, \beta_1, \dots, \beta_N)^\top \in \mathbb{R}^{N+1}$$

- Simplified linear model:

$$f(x) = \beta_0 + \sum_{j=1}^N \beta_j x_j = \mathbf{x}^\top \beta$$

- Rewrite LSC

$$L^{ls}(\beta) = \sum_{i=1}^P (y_i - \mathbf{x}_i^\top \beta)^2 = \|\mathbf{y} - \mathbf{X}\beta\|^2$$

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^\top \\ \vdots \\ \mathbf{x}_P^\top \end{pmatrix} = \begin{pmatrix} 1 & x_{1,1} & x_{1,2} & \dots & x_{1,N} \\ \vdots & & & & \vdots \\ 1 & x_{P,1} & x_{P,2} & \dots & x_{P,N} \end{pmatrix}, \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_P \end{pmatrix}$$

Minimizing the Least Square Costs

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- Find optimal parameters β

- Rewrite LSC

$$L^{ls}(\beta) = \sum_{i=1}^P (y_i - \mathbf{x}^\top \beta)^2 = \|\mathbf{y} - \mathbf{X}\beta\|^2$$

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^\top \\ \vdots \\ \mathbf{x}_P^\top \end{pmatrix} = \begin{pmatrix} 1 & x_{1,1} & x_{1,2} & \dots & x_{1,N} \\ \vdots & & & & \vdots \\ 1 & x_{P,1} & x_{P,2} & \dots & x_{P,N} \end{pmatrix}, \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_P \end{pmatrix}$$

- Compute optimum by setting the gradient to zero

$$0_P^\top = \frac{\partial L^{ls}(\beta)}{\partial \beta} = -2(\mathbf{y} - \mathbf{X}\beta)^\top \mathbf{X}$$

$$\Leftrightarrow 0_P = \mathbf{X}^\top \mathbf{X}\beta - \mathbf{X}^\top \mathbf{y}$$

$$\Leftrightarrow \hat{\beta}^{ls} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

Example - Diamonds

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

Diamonds Dataset

- **price** price in US dollars (\$326--\$18,823)
- **carat** weight of the diamond (0.2--5.01)
- **cut** quality of the cut (Fair, Good, Very Good, Premium, Ideal)
- **color** diamond colour, from J (worst) to D (best)
- **clarity** a measurement of how clear the diamond is (I1 (worst), SI2, SI1, VS2, VS1, VVS2, VVS1, IF (best))
- **x** length in mm (0--10.74)
- **y** width in mm (0--58.9)
- **z** depth in mm (0--31.8)
- **depth** total depth percentage = $z / \text{mean}(x, y) = 2 * z / (x + y)$ (43--79)
- **table** width of top of diamond relative to widest point (43--95)



Diamonds



<http://bit.ly/2FOnLVC>

File Edit View Insert Format Data Tools Add-ons Help All changes saved in Drive



100%

\$ % .0 .00 123

Arial

10

B I S A

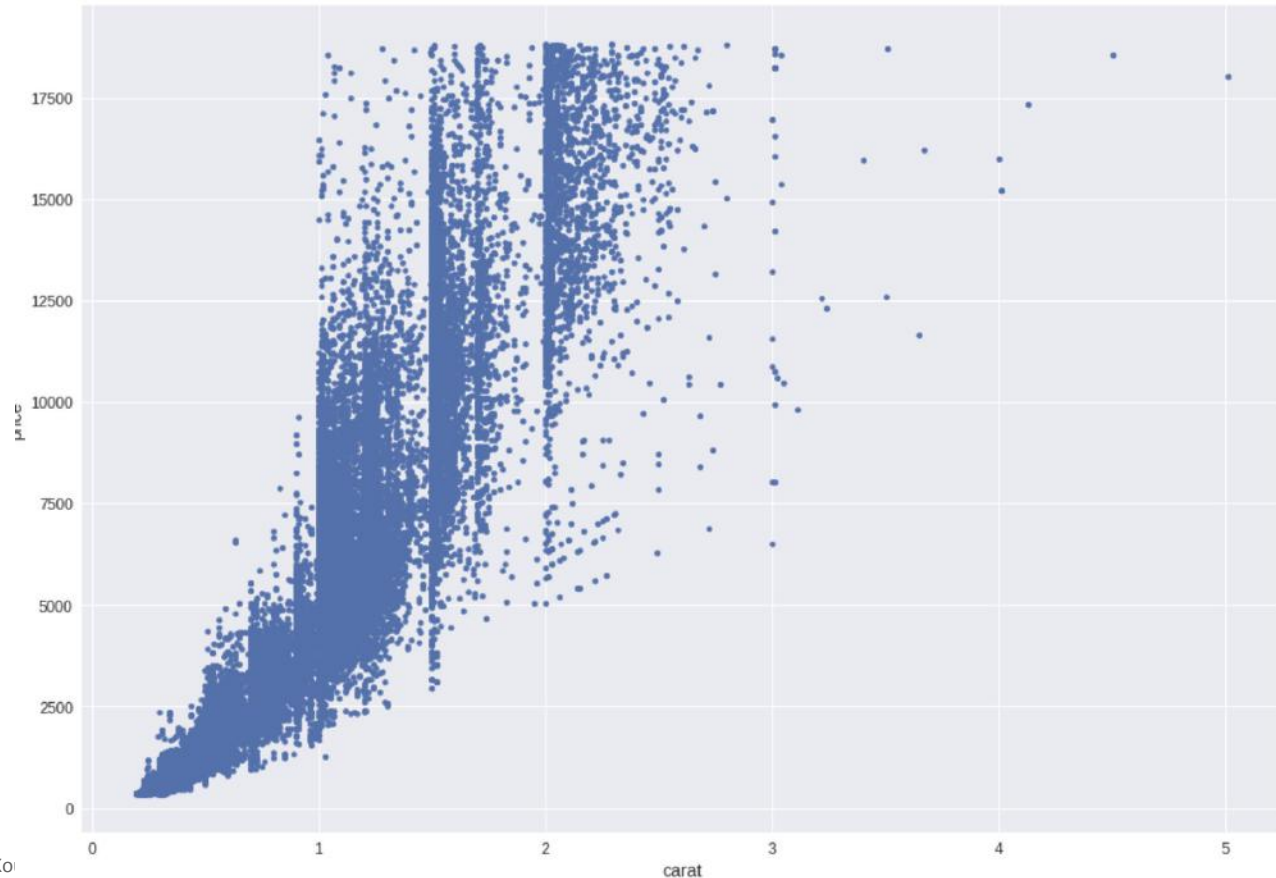


	A	B	C	D	E	F	G	H	I	J
1	carat	cut	color	clarity	depth	table	price	x	y	z
2	0.23	Ideal	E	SI2	61.5	55	326	3.95	3.98	2.43
3	0.21	Premium	E	SI1	59.8	61	326	3.89	3.84	2.31
4	0.23	Good	E	VS1	56.9	65	327	4.05	4.07	2.31
5	0.29	Premium	I	VS2	62.4	58	334	4.2	4.23	2.63
6	0.31	Good	J	SI2	63.3	58	335	4.34	4.35	2.75
7	0.24	Very Good	J	VVS2	62.8	57	336	3.94	3.96	2.48
8	0.24	Very Good	I	VVS1	62.3	57	336	3.95	3.98	2.47
9	0.26	Very Good	H	SI1	61.9	55	337	4.07	4.11	2.53
10	0.22	Fair	E	VS2	65.1	61	337	3.87	3.78	2.49
11	0.23	Very Good	H	VS1	59.4	61	338	4	4.05	2.39
12	0.3	Good	J	SI1	64	55	339	4.25	4.28	2.73
13	0.23	Ideal	J	VS1	62.8	56	340	3.93	3.9	2.46
14	0.22	Premium	F	SI1	60.4	61	342	3.88	3.84	2.33
15	0.31	Ideal	J	SI2	62.2	54	344	4.35	4.37	2.71
16	0.2	Premium	E	SI2	60.2	62	345	3.79	3.75	2.27
17	0.32	Premium	E	I1	60.9	58	345	4.38	4.42	2.68
18	0.3	Ideal	I	SI2	62	54	348	4.31	4.34	2.68
19	0.3	Good	J	SI1	63.4	54	351	4.23	4.29	2.7
20	0.3	Good	J	SI1	63.8	56	351	4.23	4.26	2.71
21	0.3	Very Good	J	SI1	62.7	59	351	4.21	4.27	2.66
22	0.3	Good	I	SI2	63.3	56	351	4.26	4.3	2.71
23	0.23	Very Good	E	VS2	63.8	55	352	3.85	3.92	2.48

Example - Diamonds

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- There might be a correlation between the **weight** of a diamond and its **price**

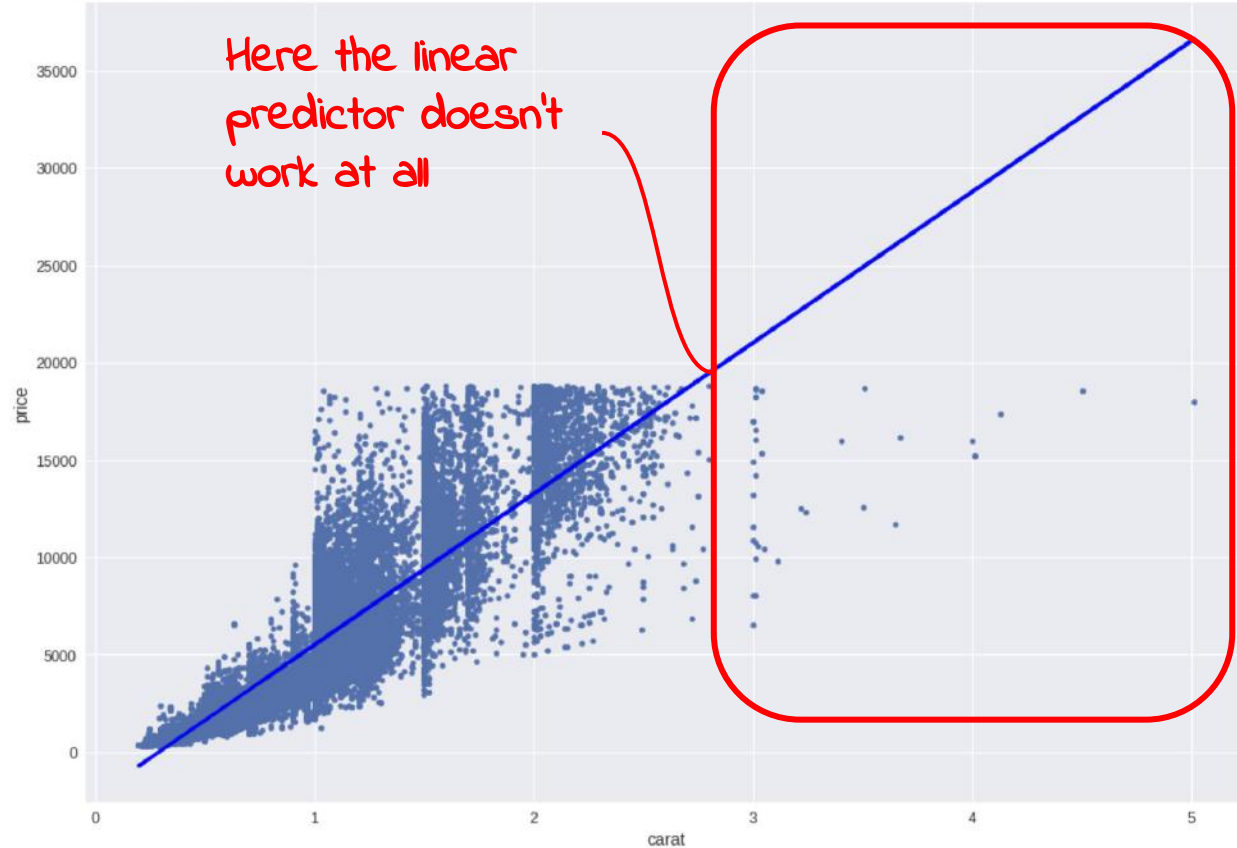


Example - Diamonds

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- The linear regression prediction underfits the data
- There might be a **non linear correlation**
- The **price** might also be dependent on other factors, such as e.g. **cut**, **clarity**, **colour**, etc.

Mean squared error: 28974532.09



Root Mean Square Error - RMSE

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- A popular measure for the achieved quality of the prediction is the **Root Mean Square Error RMSE**:

$$RMSE(\mathbf{X}, h) = \sqrt{\frac{1}{m} \sum_{i=1}^m (h(\mathbf{x}_i^T) - y_i)^2}$$

with m ... number of dataset instances
 $\mathbf{x}_i^T$... feature vector for i th instance
 y_i ... label (desired output) of i th instance
 h ... hypothesis (prediction function)

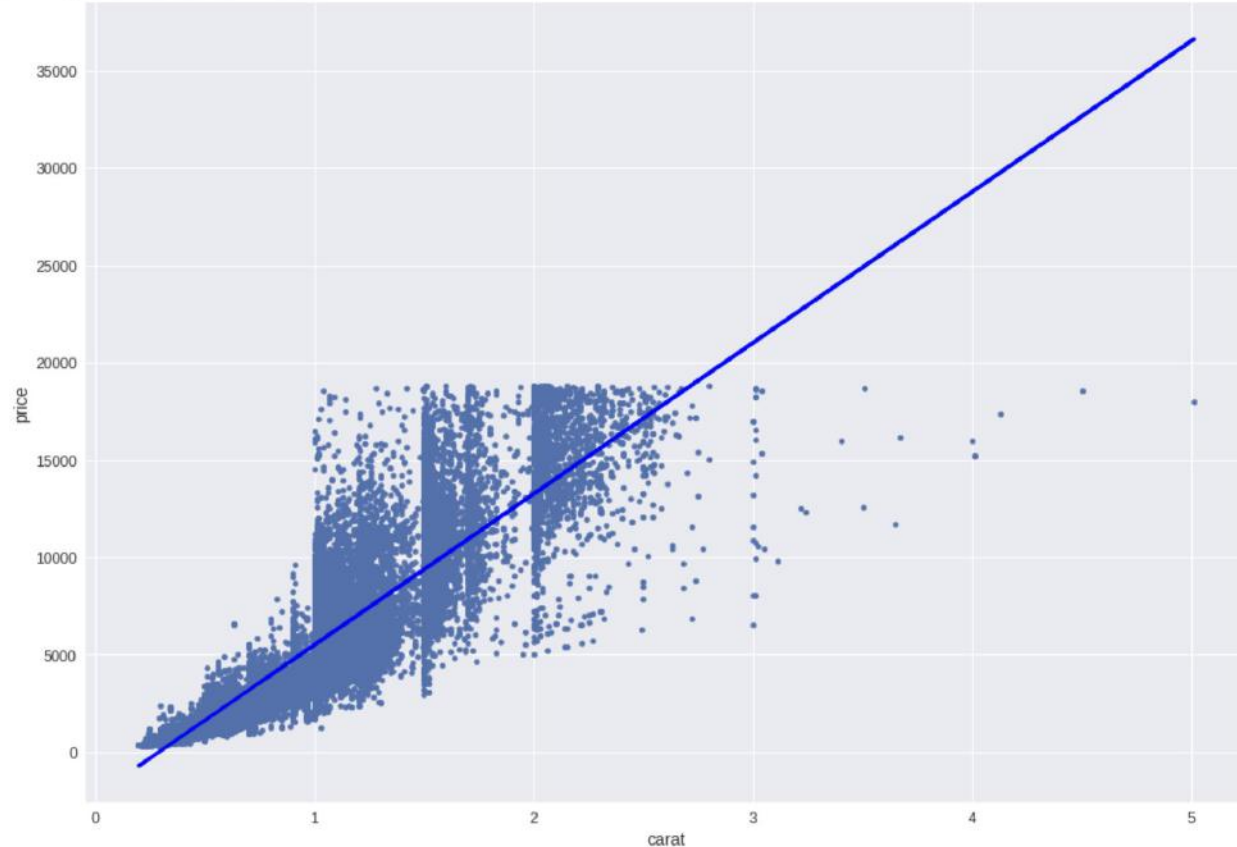
$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_p^T \end{pmatrix}$$

Example - Diamonds

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- The linear regression prediction underfits the data
- There might be a **non linear correlation**
- The **price** might also be dependent on other factors, such as e.g. **cut**, **clarity**, **colour**, etc.
- RMSE = **1548.53**

Mean squared error: 28974532.09





Diamonds



<http://bit.ly/2FOnLVC>

File Edit View Insert Format Data Tools Add-ons Help All changes saved in Drive



100%

\$

%

.0

.00

123

Arial

10

B

I

S

A

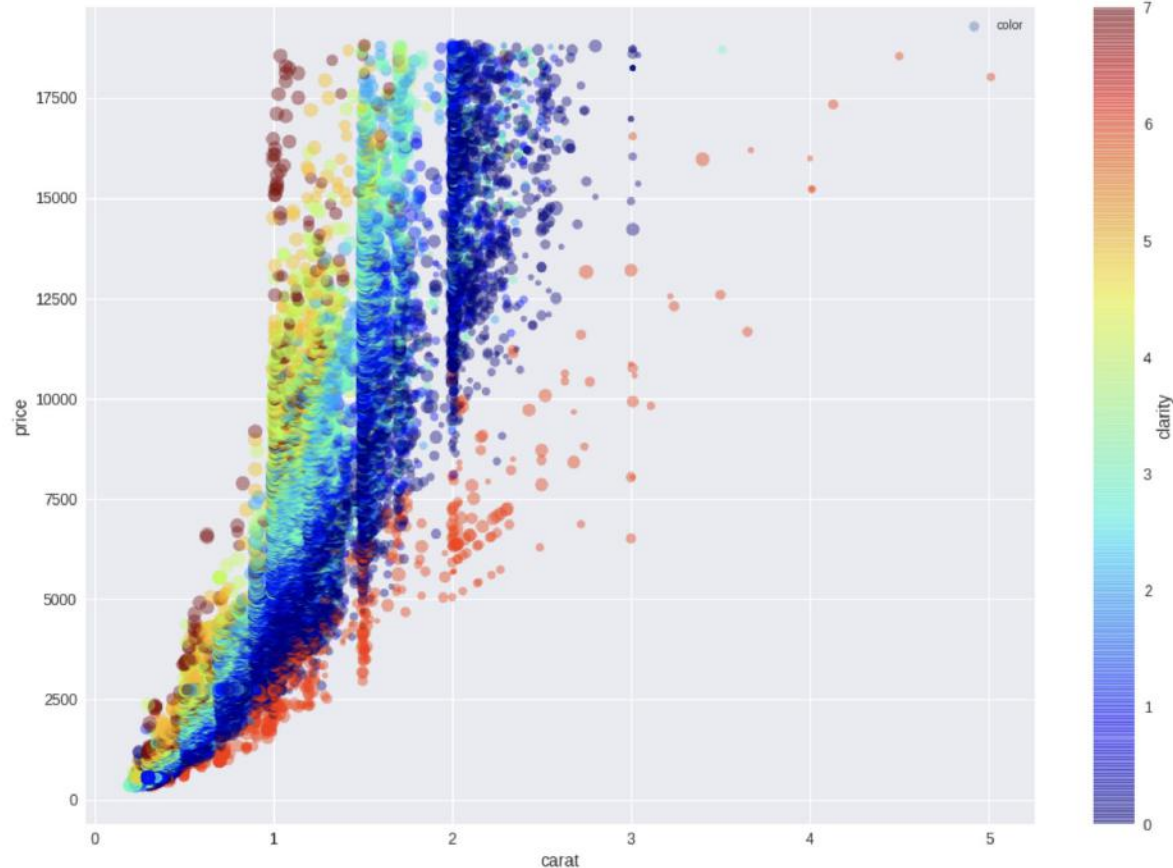


	A	B	C	D	E	F	G	H	I	J
1	carat	cut	color	clarity	depth	table	price	x	y	z
2	0.23	Ideal	E	SI2	61.5	55	326	3.95	3.98	2.43
3	0.21	Premium	E	SI1	59.8	61	326	3.89	3.84	2.31
4	0.23	Good	E	VS1	56.9	65	327	4.05	4.07	2.31
5	0.29	Premium	I	VS2	62.4	58	334	4.2	4.23	2.63
6	0.31	Good	J	SI2	63.3	58	335	4.34	4.35	2.75
7	0.24	Very Good	J	VVS2	62.8	57	336	3.94	3.96	2.48
8	0.24	Very Good	I	VVS1	62.3	57	336	3.95	3.98	2.47
9	0.26	Very Good	H	SI1	61.9	55	337	4.07	4.11	2.53
10	0.22	Fair	E	VS2	65.1	61	337	3.87	3.78	2.49
11	0.23	Very Good	H	VS1	59.4	61	338	4	4.05	2.39
12	0.3	Good	J	SI1	64	55	339	4.25	4.28	2.73
13	0.23	Ideal	J	VS1	62.8	56	340	3.93	3.9	2.46
14	0.22	Premium	F	SI1	60.4	61	342	3.88	3.84	2.33
15	0.31	Ideal	J	SI2	62.2	54	344	4.35	4.37	2.71
16	0.2	Premium	E	SI2	60.2	62	345	3.79	3.75	2.27
17	0.32	Premium	E	I1	60.9	58	345	4.38	4.42	2.68
18	0.3	Ideal	I	SI2	62	54	348	4.31	4.34	2.68
19	0.3	Good	J	SI1	63.4	54	351	4.23	4.29	2.7
20	0.3	Good	J	SI1	63.8	56	351	4.23	4.26	2.71
21	0.3	Very Good	J	SI1	62.7	59	351	4.21	4.27	2.66
22	0.3	Good	I	SI2	63.3	56	351	4.26	4.3	2.71
23	0.23	Very Good	E	VS2	63.8	55	352	3.85	3.92	2.48

Example - Diamonds

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- Now we take besides the **weight** also **clarity** and **color** into account
- **Clarity** and **color**, as well as **cut** are **non numeric attributes**
- Non numeric attributes cannot be directly used in regression. They have to be **mapped to numerical data**
E.g. colors:
"D"=0, "E"=1, "J"=2, etc.
- Beware of **induced numerical dependencies**



Non Linear (polynomial) Regression

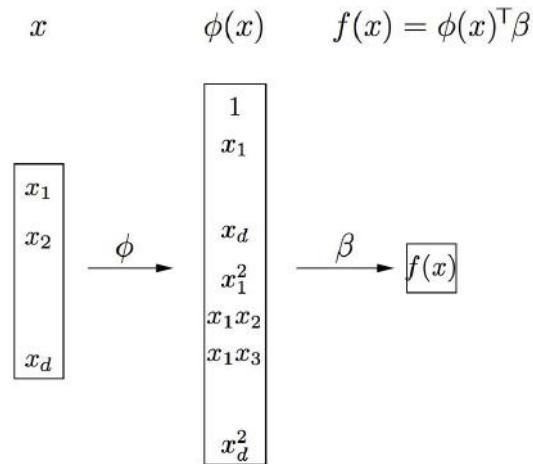
Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- Replace the inputs $x_i \in \mathbb{R}^d$ by some **non-linear features** $\phi(x_i) \in \mathbb{R}^k$
- The optimal β is the same $\hat{\beta}^{ls} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$ but with $\mathbf{X} = \begin{pmatrix} \phi(x)_1^\top \\ \vdots \\ \phi(x)_P^\top \end{pmatrix} \in \mathbb{R}^{n \times k}$
- **What are “features”?**
 - a) Features are an arbitrary set of basis functions
 - b) Any function linear in β can be written as $f(x) = \phi(x)^\top \beta$
for some ϕ - which we denote as **“features”**

Non Linear (polynomial) Regression

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

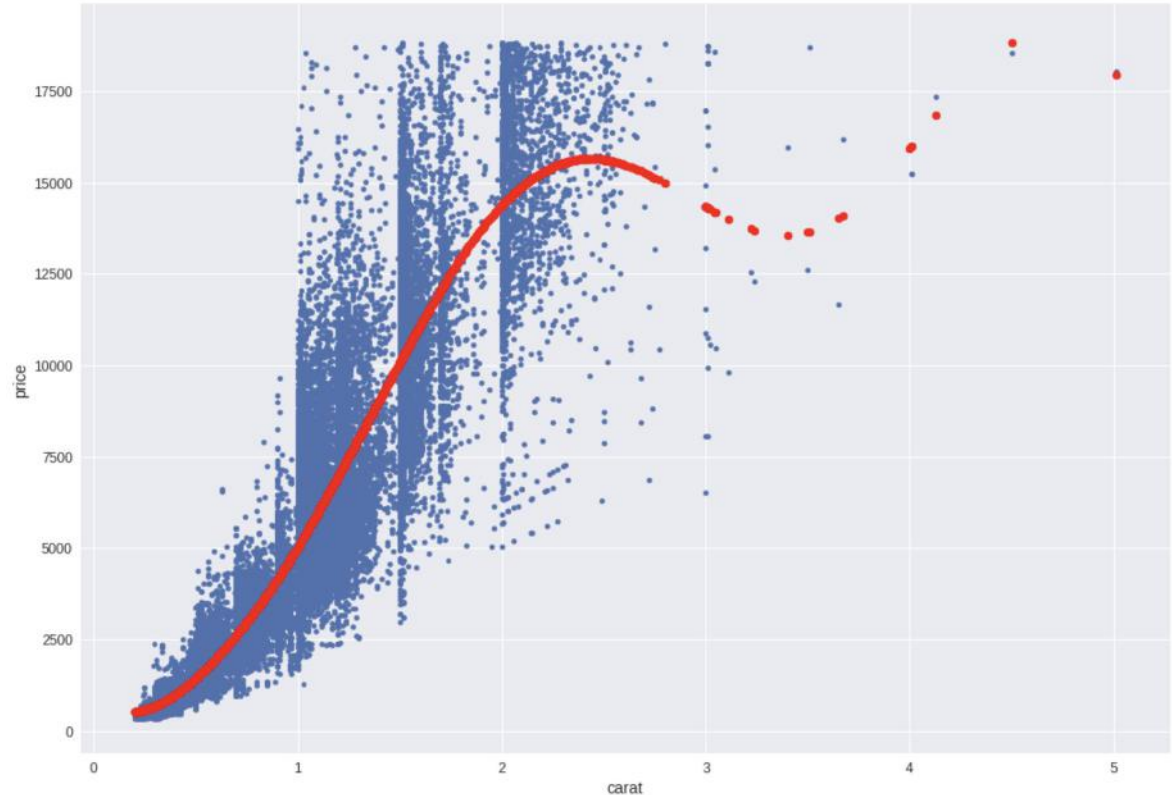
- Linear features: $\phi(x) = (1, x_1, \dots, x_N) \in \mathbb{R}^{1+N}$
- Quadratic features $\phi(x) = (1, x_1, \dots, x_N, x_1^2, x_1x_2, x_1x_3, \dots, x_n^2) \in \mathbb{R}^{1+N+\frac{N(N+1)}{2}}$



Example - Diamonds

Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- We might also try a **non linear (polynomial) regression**
- Here degree=8,
RMSE = 1434.91

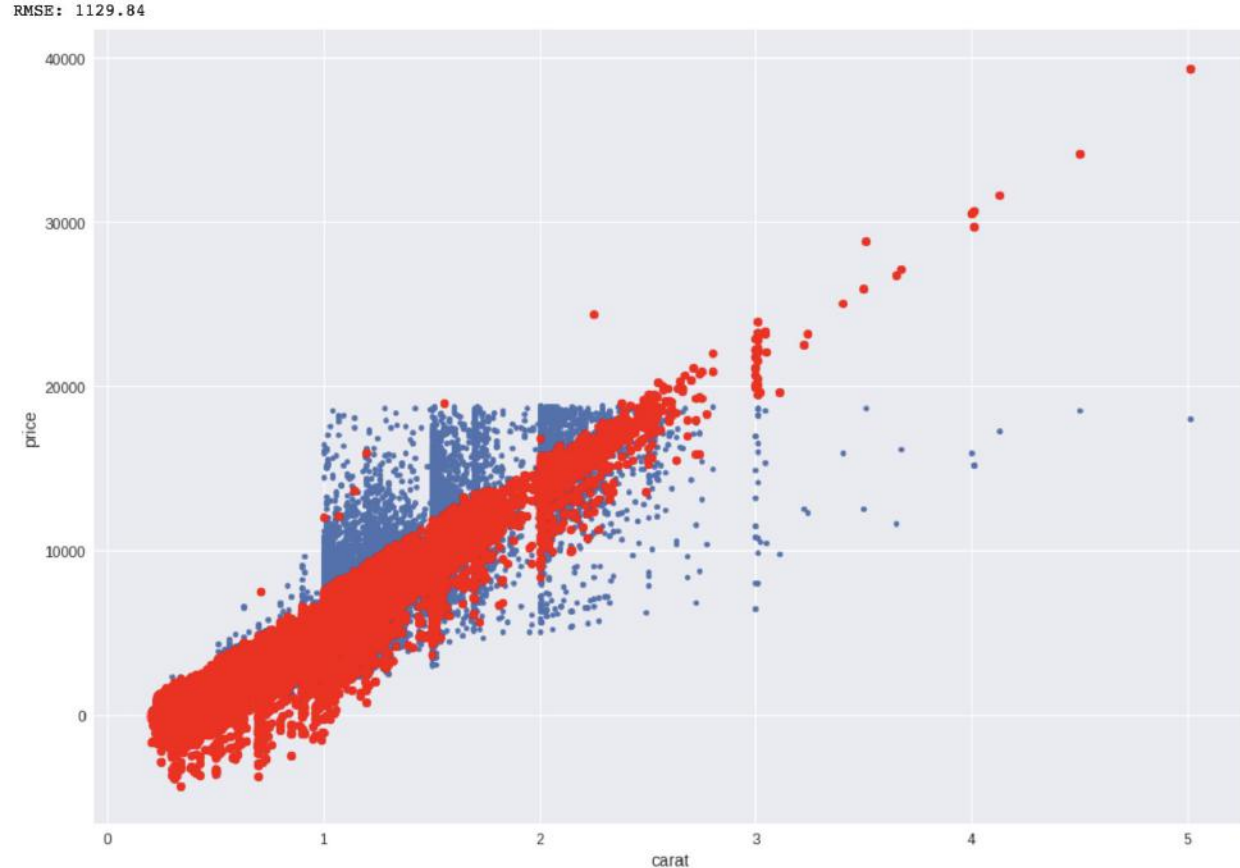


Example - Diamonds

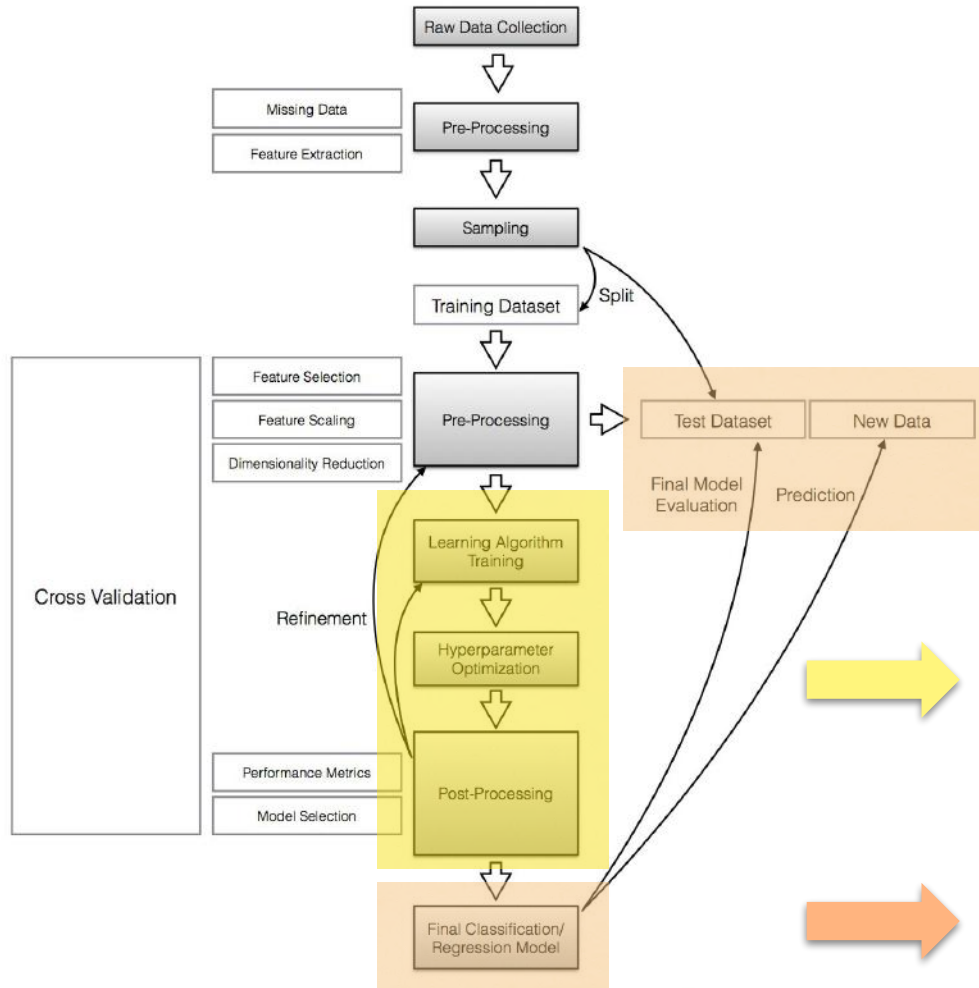
Information Service Engineering - Lecture 5 Basic Machine Learning - 5.4 Basic Machine Learning Algorithms

- **Numerical data has to be scaled**, i.e. features with large values have more influence in regression than features with small values
- **Non-numerical data must be mapped to a numerical representation** while avoiding potentially wrong numerical dependencies

RMSE = 1129.84

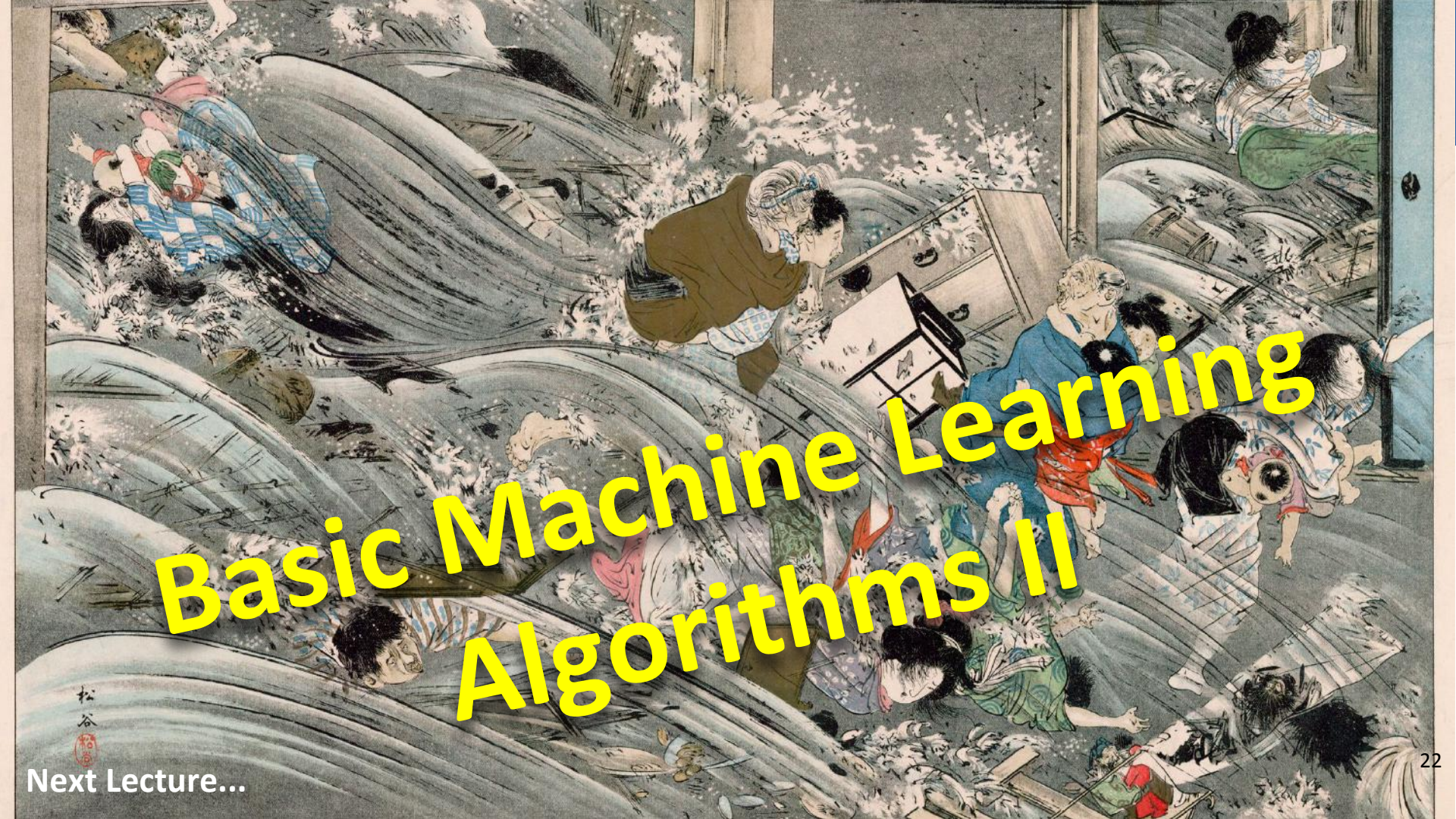


Supervised Learning



Training/Optimisation Phase

Prediction Phase



Basic Machine Learning Algorithms II

松谷
Next Lecture...

Picture References:

- P.9: Collection of diamonds, <https://commons.wikimedia.org/wiki/File:Diamonds.jpg>
- P.22: Print illustrating the tsunami that followed the 7.2 magnitude Meiji Sanriku earthquake of 1896
[https://commons.wikimedia.org/wiki/File:Print_illustrating_the_tsunami_that_followed_the_7.2_magnitude_Meiji_Sanriku_earthquake_of_1896_\(13720269294\).jpg](https://commons.wikimedia.org/wiki/File:Print_illustrating_the_tsunami_that_followed_the_7.2_magnitude_Meiji_Sanriku_earthquake_of_1896_(13720269294).jpg)