Final projects

Present (10 mins) the research approach to the topic of your interest:

- Research question
- Data and its collection
- Data analysis
- Possible outcomes

Machine learning

Machine Learning

what society thinks I do

what my friends think I do

what my parents think I do

$$L_P = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i$$

$$\alpha_i \geq 0, \forall i$$

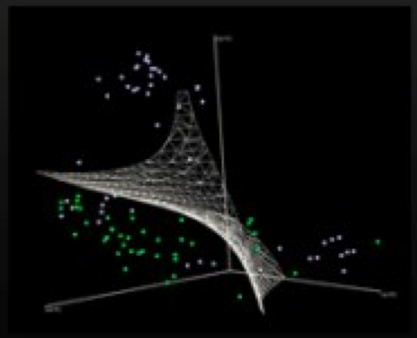
$$\mathbf{w} = \sum_{i=1}^{l} \alpha_i y_i \mathbf{x}_i, \sum_{i=1}^{l} \alpha_i y_i = 0$$

$$\nabla \hat{g}(\theta_t) = \frac{1}{n} \sum_{i=1}^{n} \nabla \ell(x_i, y_i; \theta_t) + \nabla r(\theta_t).$$

$$\theta_{t+1} = \theta_t - \eta_t \nabla \ell(x_{i(t)}, y_{i(t)}; \theta_t) - \eta_t \cdot \nabla r(\theta_t)$$

$$\mathbb{E}_{i(t)}[\ell(x_{i(t)}, y_{i(t)}; \theta_t)] = \frac{1}{n} \sum_i \ell(x_i, y_i; \theta_t).$$

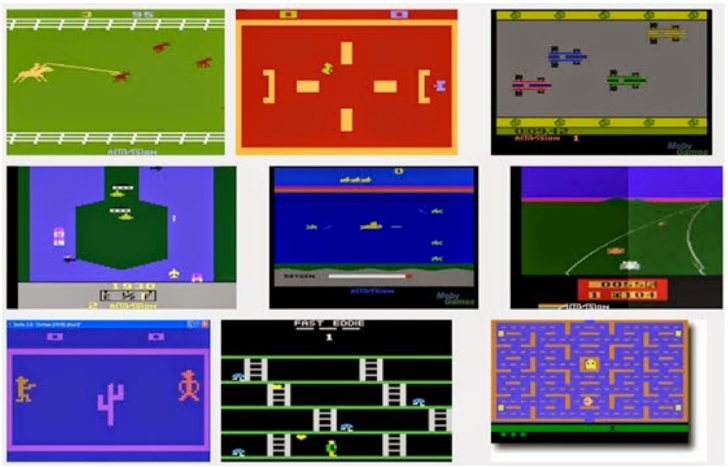
what other programmers think I do



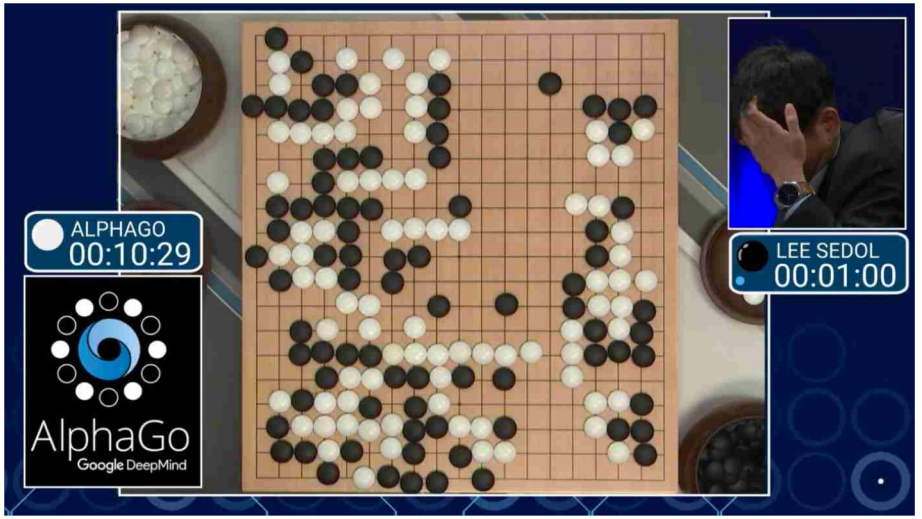
what I think I do

```
>>> from sklearn import svm
```

what I really do



Deepmind development – beets humans on 49 Atari games



AlphaGo defeated professional Go player

Classic IBM Watson in Jeopardy game

Google self-driving - accident free car

Artificial intelligence

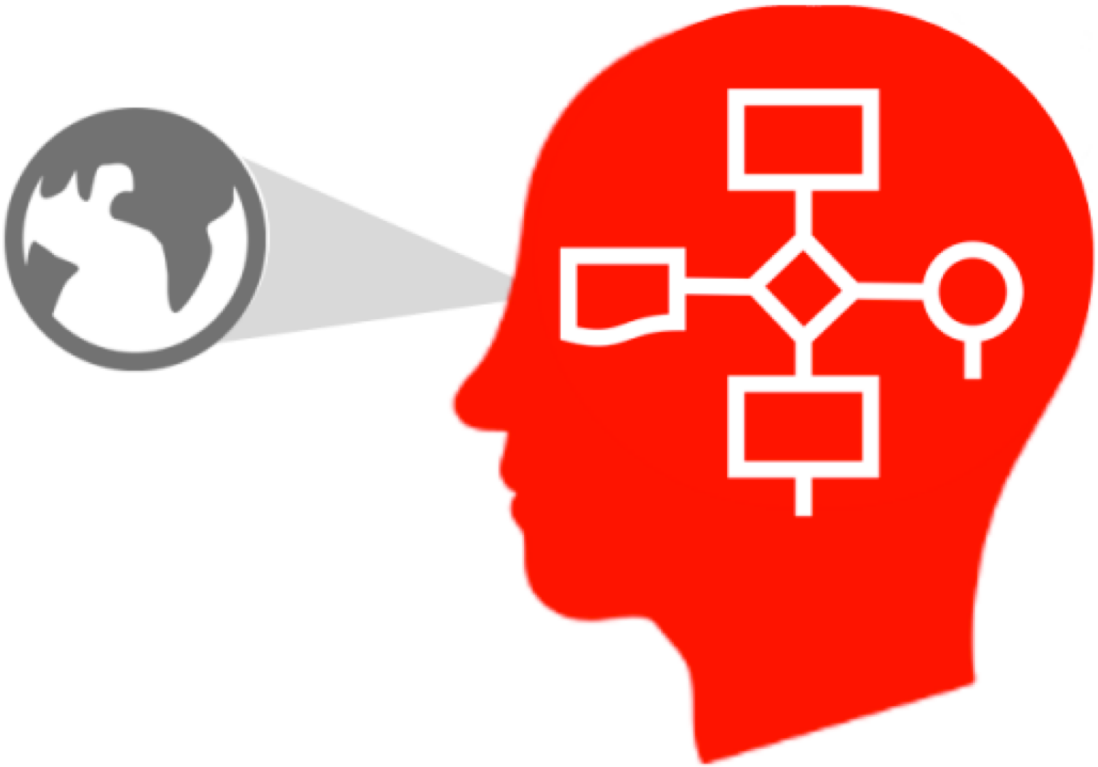
Computers ability to do tasks traditionally in the domain of humans

But how do the computers solve the tasks, and comes the explanations of the world?

RE-FRAMING

A **mental model** is:
'an explanation of someone's thought process about how something works in the real world'.

Wikipedia



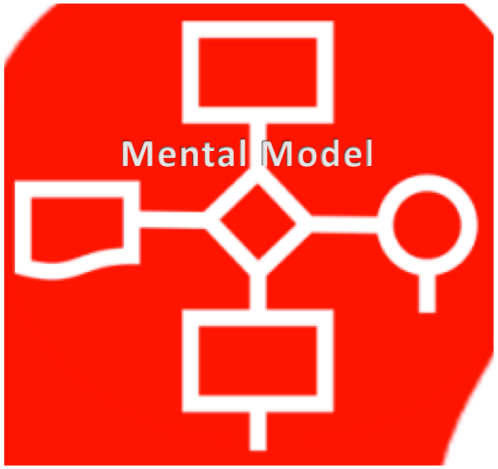
http://salesstrategypitstop.theasggroup.com/2014/12/03/changing-behavior-by-working-on-the-underlying-mental-models/

What is a model

We are creating a model – a mental model, the way we view the world:

- Generalization on gender
  - If I would apply to IT company to work on software testing I would hear that males are not attentive to details enough;
- Branding is common way to create a mental model for us.
  - If the product usage was satisfactory, or commercial appealing – I will buy new product from this company

Human Decision Making Process



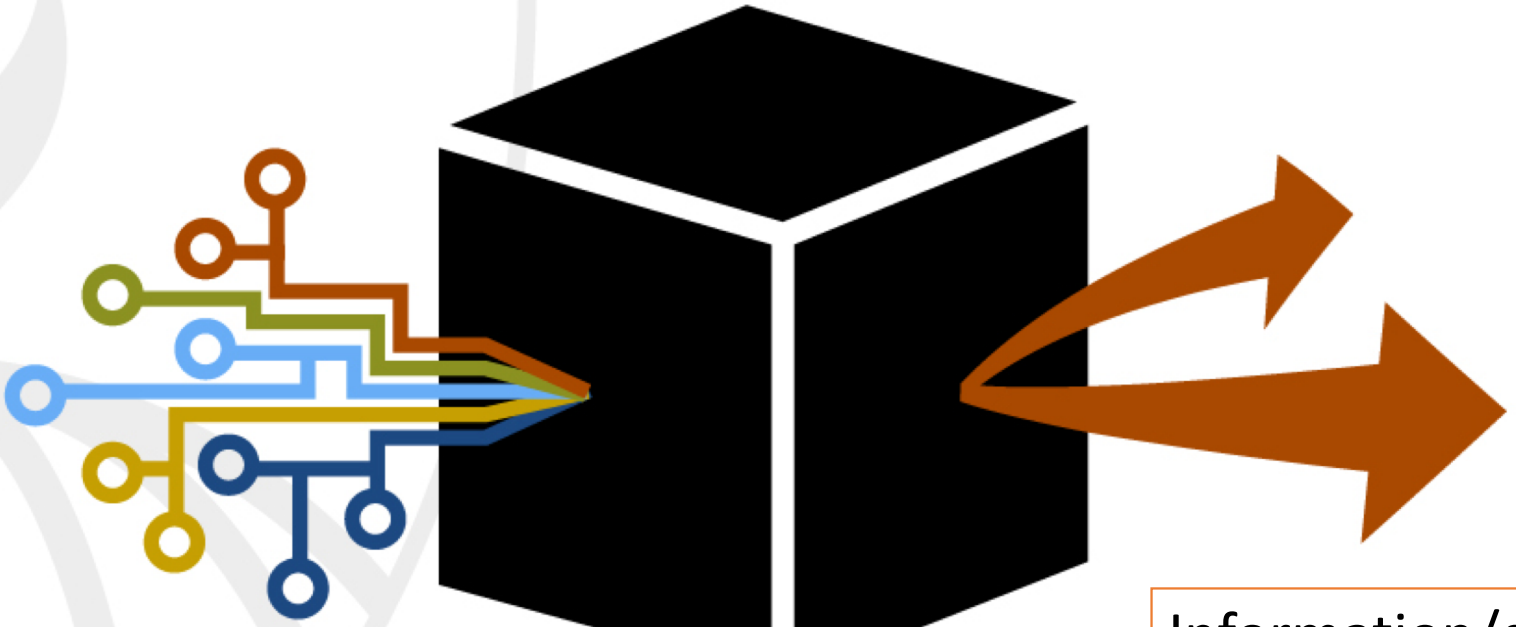
The scheme

- Based on on our experience, observations, examples of life, other peoples opinions we create a mental model by generalizing that experience to find an easy way for a decision making.

  E.g. Bank in which I invested got bankrupted – I never keep my money in a bank; Journal rejected my article – I never submit to this journal again, or I am not suitable for academia; I scored few goals in amateur game – I am good football player.

- We have observations, that creates model, and features on which we are deciding what decision to take.
  - Person is male with Turku University degree – I am hiring him (Features: Gender, education institution).

Machine Decision Making Process

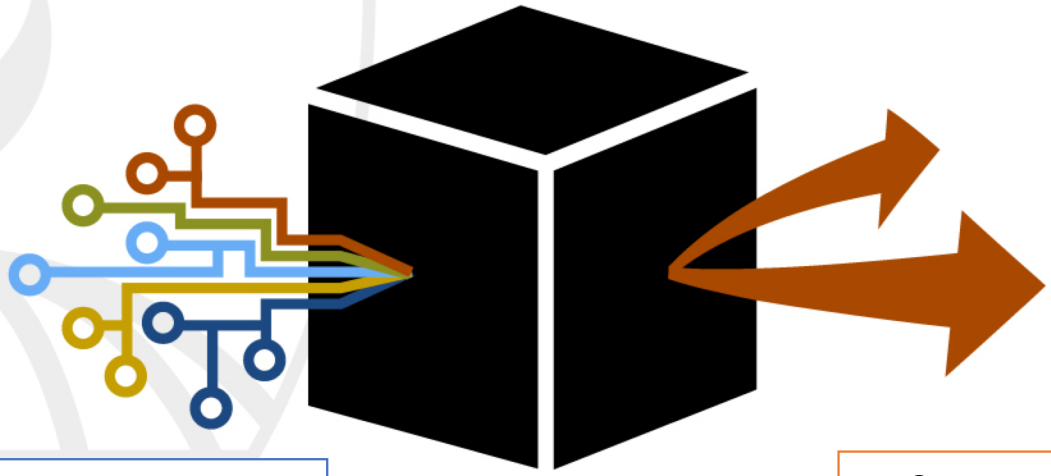


Data/Evidences

Model

Information/decision

What is a feature for the machine learning model?



Data/Evidences

Information/decision

Model

Selling an Apartment

85 m^2
Kallio
Sauna
4^{th} floor
....

300.000 €

What is Machine Learning

"field of study that gives computers the ability to learn without being explicitly programmed." - Arthur Samuel (1959)

- Implementing knowledge to computers without hardcoding that knowledge
- Giving the data and hoping that machine will create a meaning of it

How machines are learning?

Supervised learning – training machine while knowing the right answers

Real world examples – solving mathematical equations

Reinforcement learning – giving the feedback on each step of learning

Real world examples - hill climbing, searching way out from a labyrinth

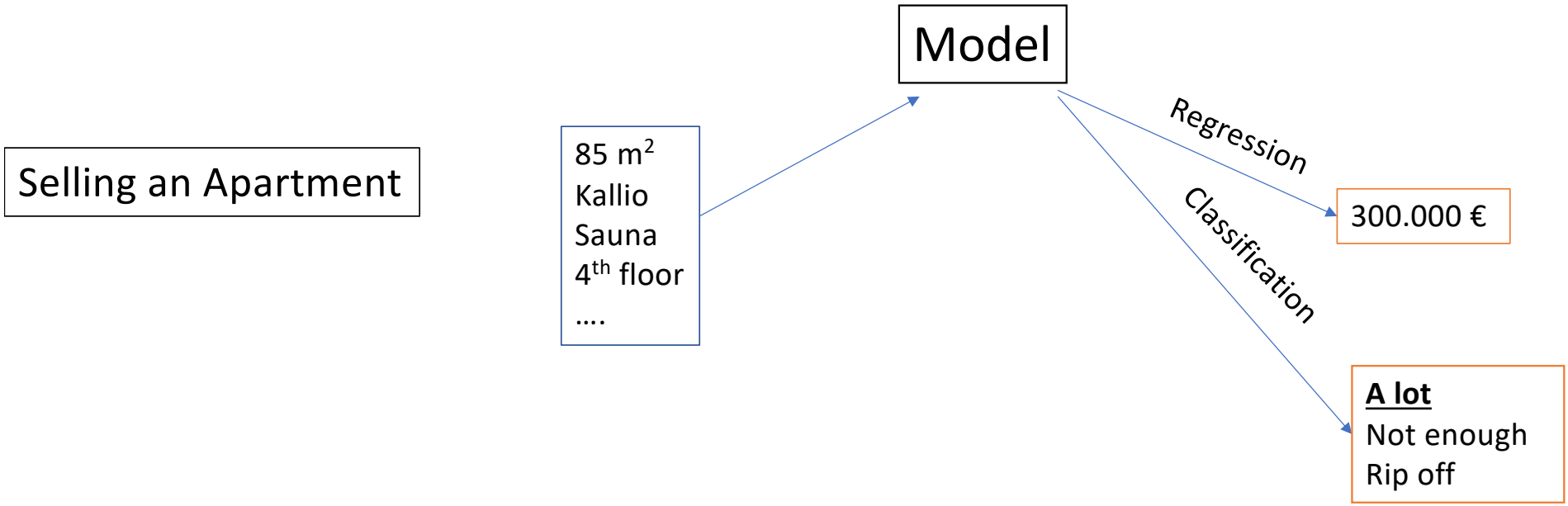
Unsupervised learning – giving the data and machine does the rest

Real world examples – our brain work

Supervised learning

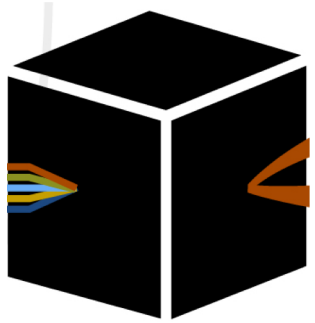
Regression – output is continuous data (-∞, +∞) or just any number

Classification – output is classes, groups, of just defined choices



How machines are coming to the conclusion

Creating a model based on **Trial** and **Error**:



E.g. Mathematics lesson at school

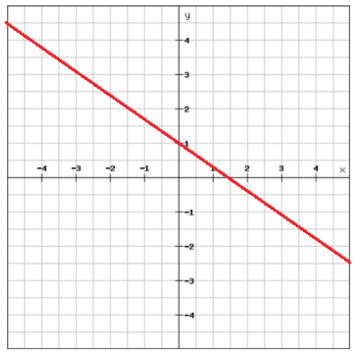
- We have mathematical **equation**, we know that there is a **right answer**, we **solving** it based on our **knowledge** and **comparing** whether **answer** we've got it's the same as the **real answer**. If not - we are **adjusting** our **knowledge** (mental model), until we are reaching real answer.

Regression – is a model for predictions

Mental model expressed with mathematical equation

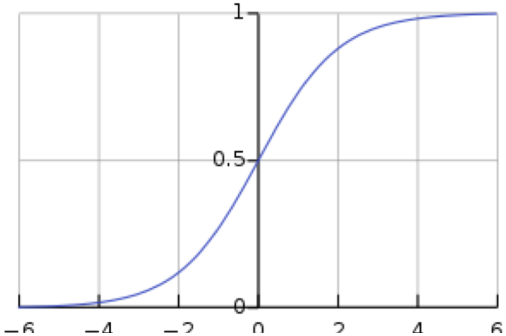
- Linear regression - formula that can draw straight line on canvas

  $Y = b*X$

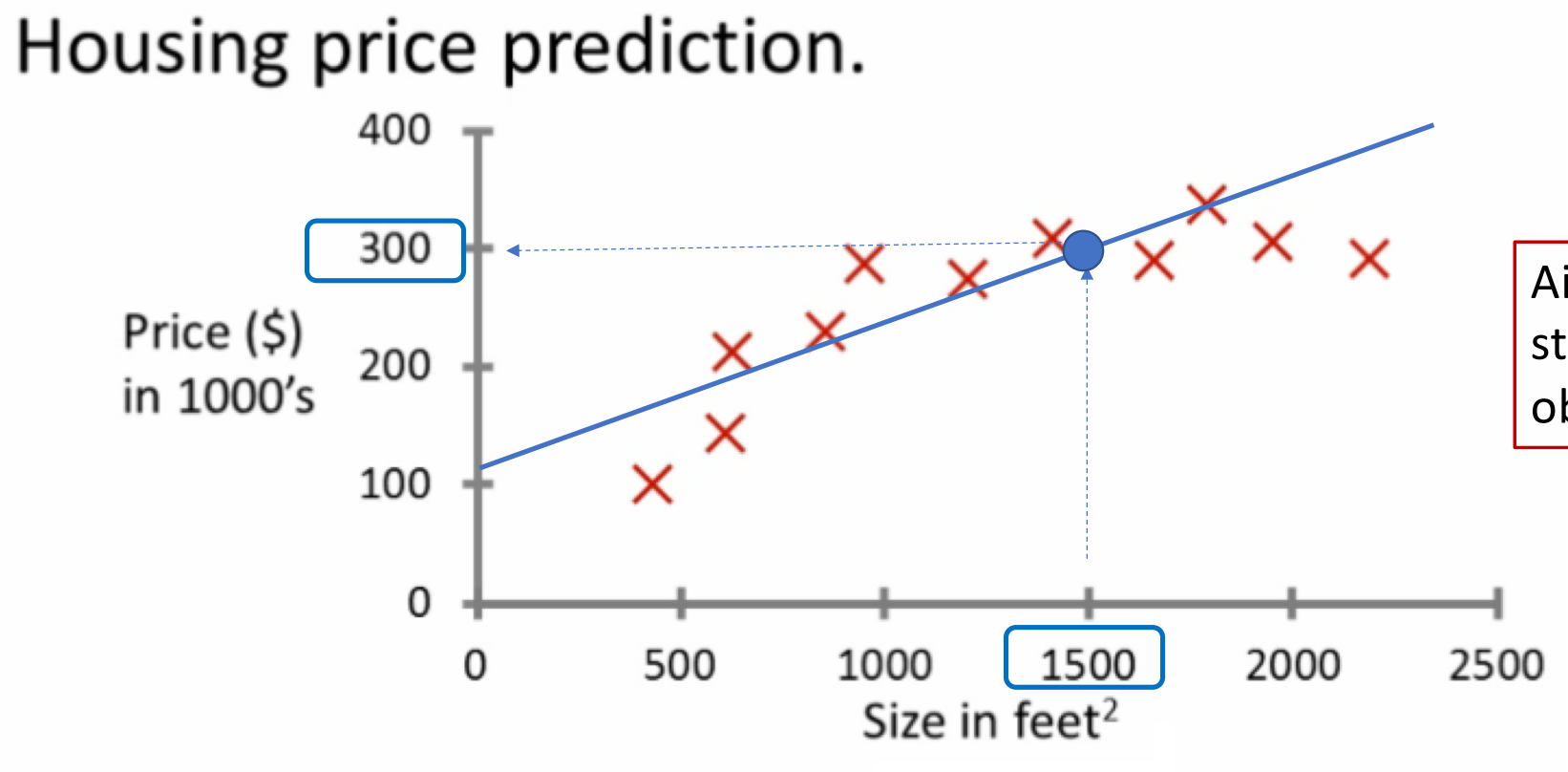


- Logistic regression - bended line on canvas

  $Y = 1 / (1 + e^{-x})$



Example of Linear Regression



Aim is to draw the straight line between observation points

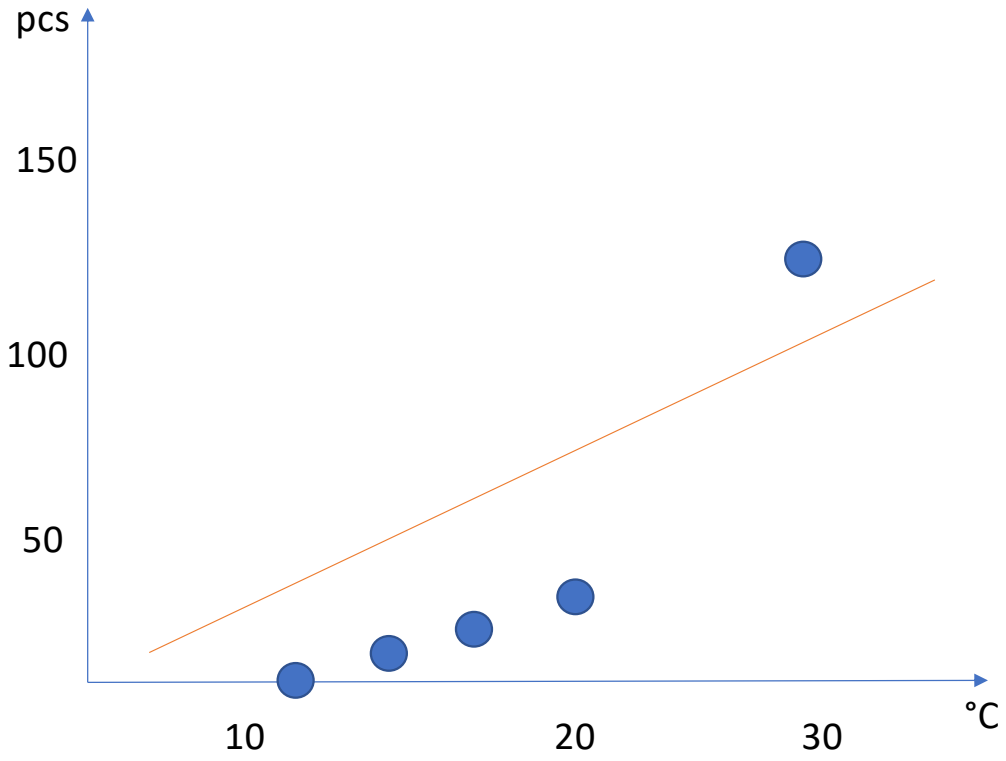
If you have **1500** ft^2 it has worth of **300.000** $

Example: Ice cream shop

If its 20°C for the next day, how much ice cream to order

Observations

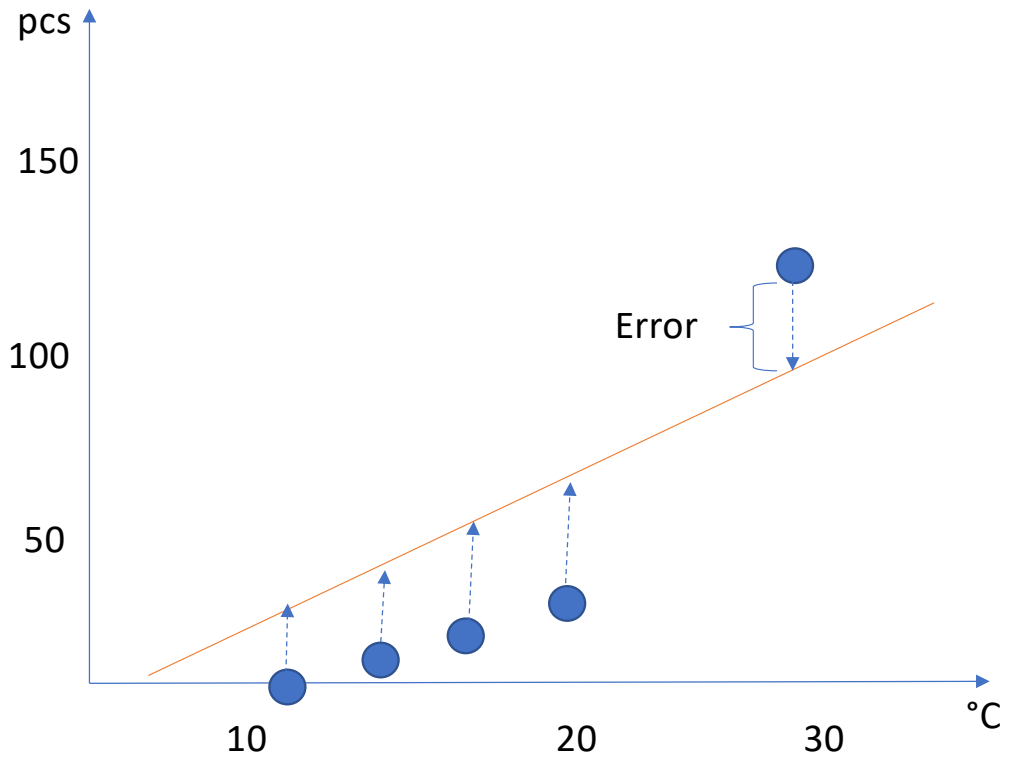
Temperature	Ice Cream sales
20°C	24 pcs
17°C	15 pcs
28°C	135 pcs
12°C	0 pcs
15°C	17 pcs
…	…



Ordering ice creams for the shop based on weather forecast

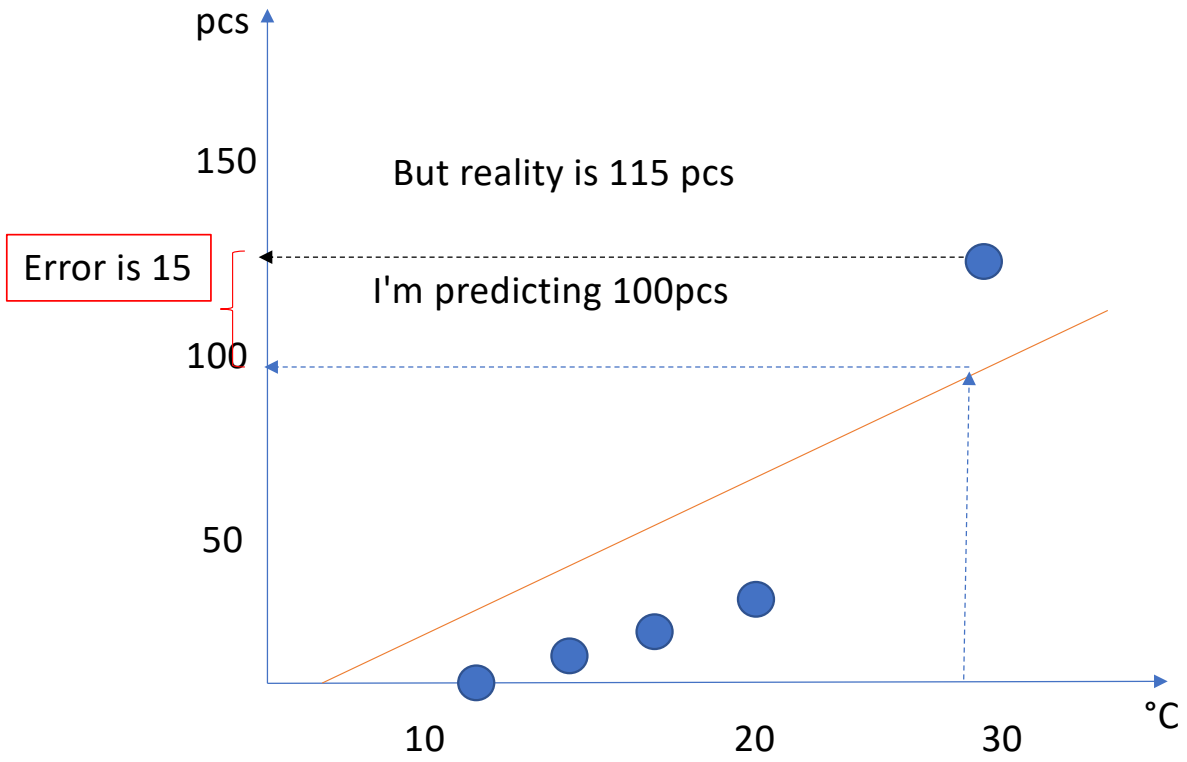
previous experiences

Mental model based on experiences and used for future situation



Least Squared Errors method
Drawing various lines and calculating difference between a line and every point. After squaring and summing them.

Error – how far is my prediction from the real case based on previous observations



We finding all errors:
-15, -28, - 30, -25, 15

Squaring them to make a number positive and punish too big differences:
225, 784, 900, 625, 225

Summing them:
225 + 784 + 900 + 625 + 225 = **2759**
That's our error for this particular line

Repeating with the next line

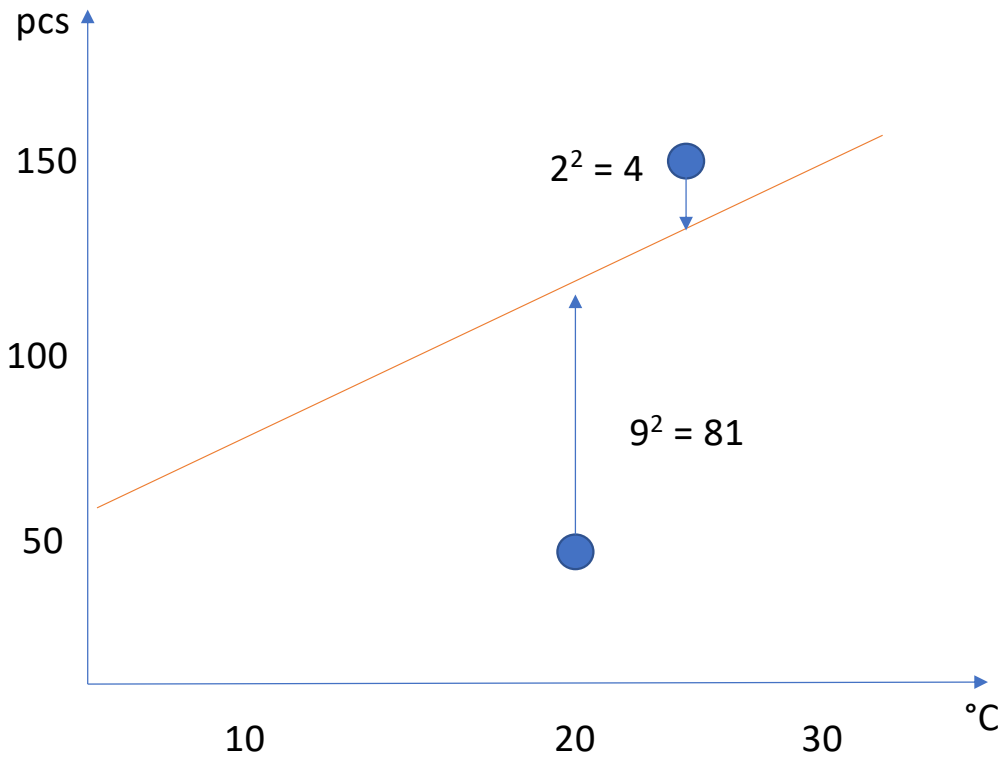
Choosing the line with the least error

Why we square the error (difference between real and predicted value)?

- Removing negatives, otherwise after summation
  the error could be close to zero although
  the differences between real and predicted
  value

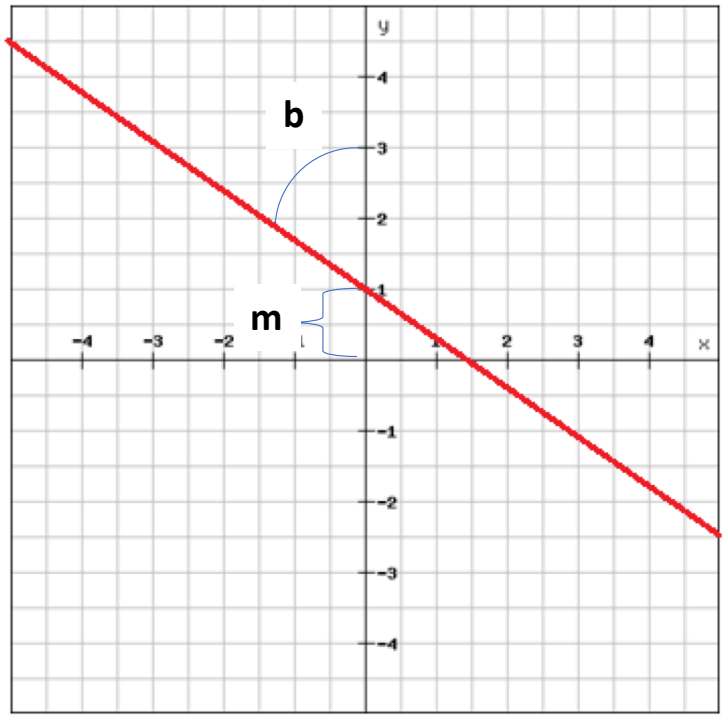
$$|-5| = 5$$

- Punishing too big error (difference), if difference is 2,
  ordinary error will be 2, while squared 4.
  if difference is 9 ordinary error is 9,
  while squared - 81



Mathematical expression of regression

Y = b*X + m



Y = m*X + b

What we need to have to create a model/line/theory:

Y – result/outcome/value to be predicted

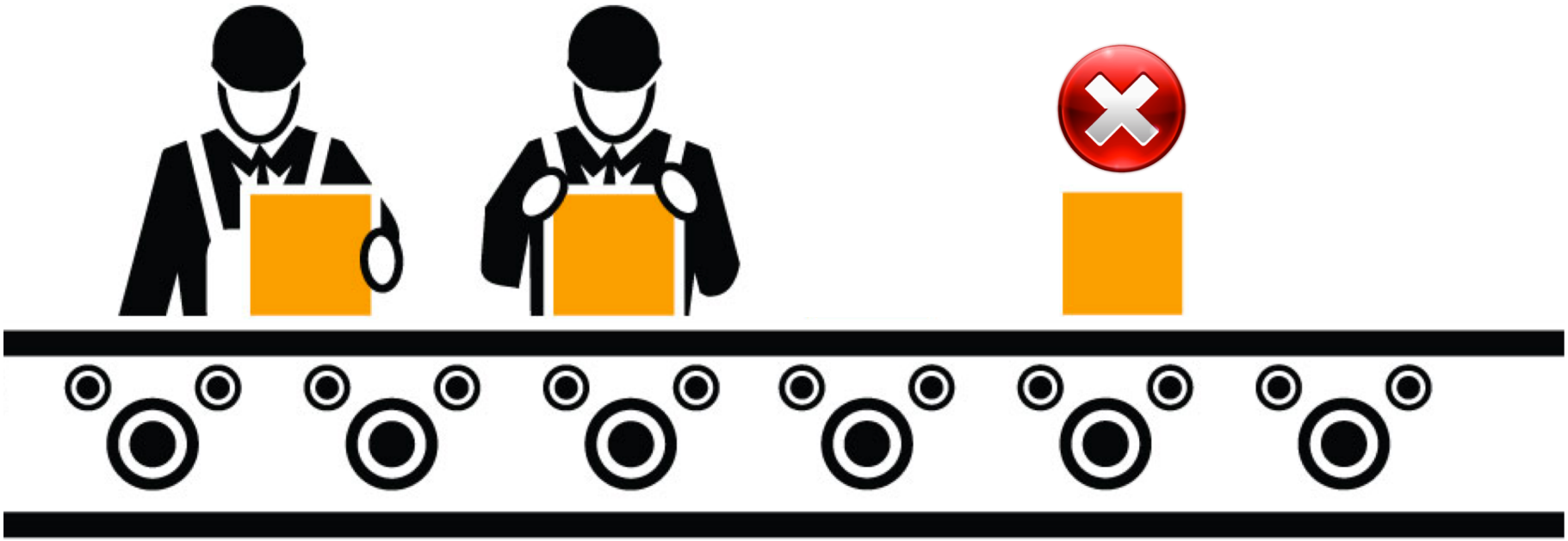
X – observation/input/on what value will be predicted

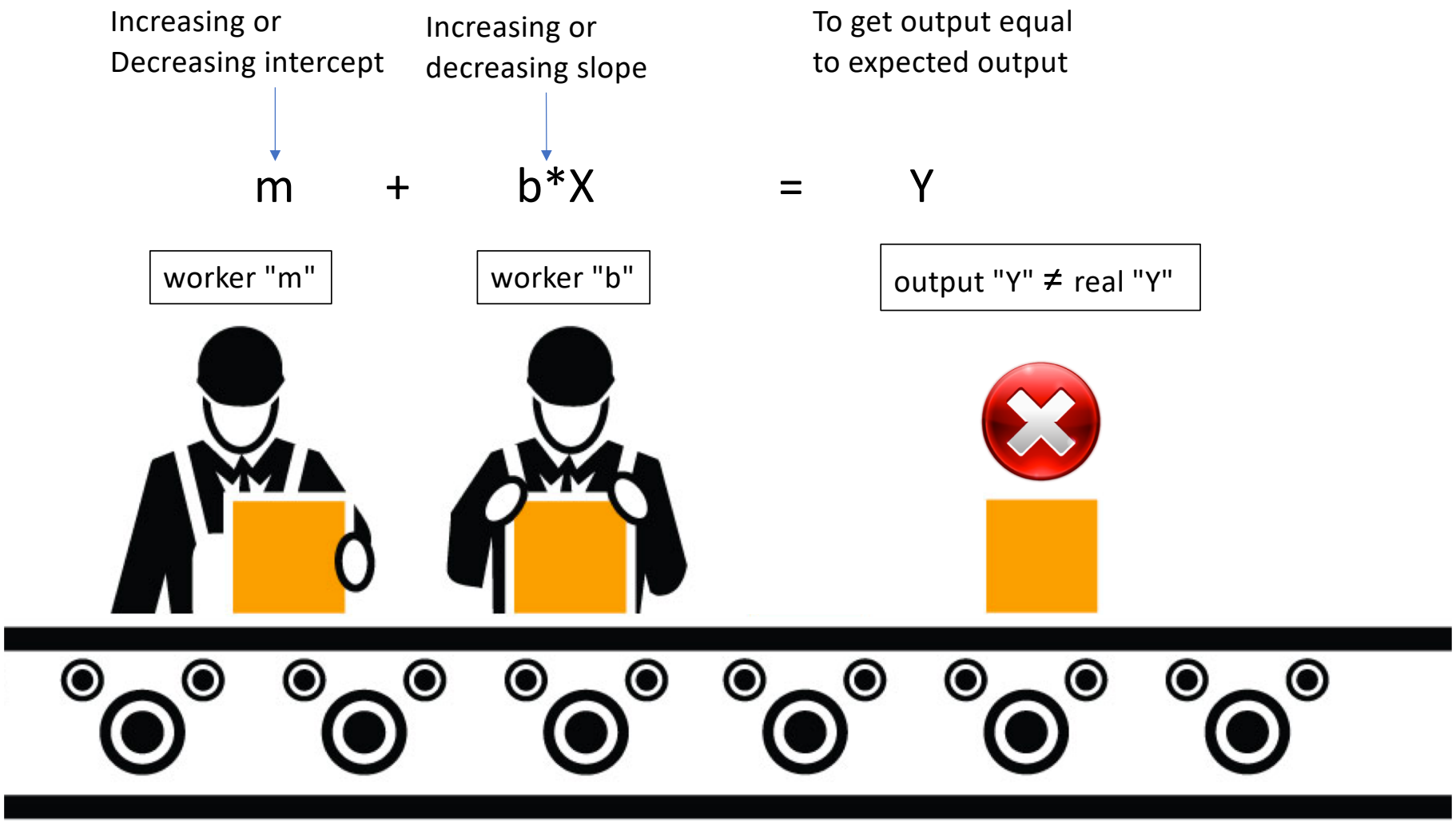
What computer needs to come up with to create a model/line/theory:

b – slope or a coefficient of an input variable

m – intercept or a beginning point what would outcome would be if variable(s) are equal to zero

Output not good, whom to blame?




Increasing or
Decreasing intercept
Increasing or
decreasing slope
To get output equal
to expected output
m
+
b*X
=
Y
worker "m"
worker "b"
output "Y" ≠ real "Y"

The aim of linear regression is to find appropriate (best) "**m**" and "**b**":

Example:

Y = b* X + m

24 = b* 20 + m

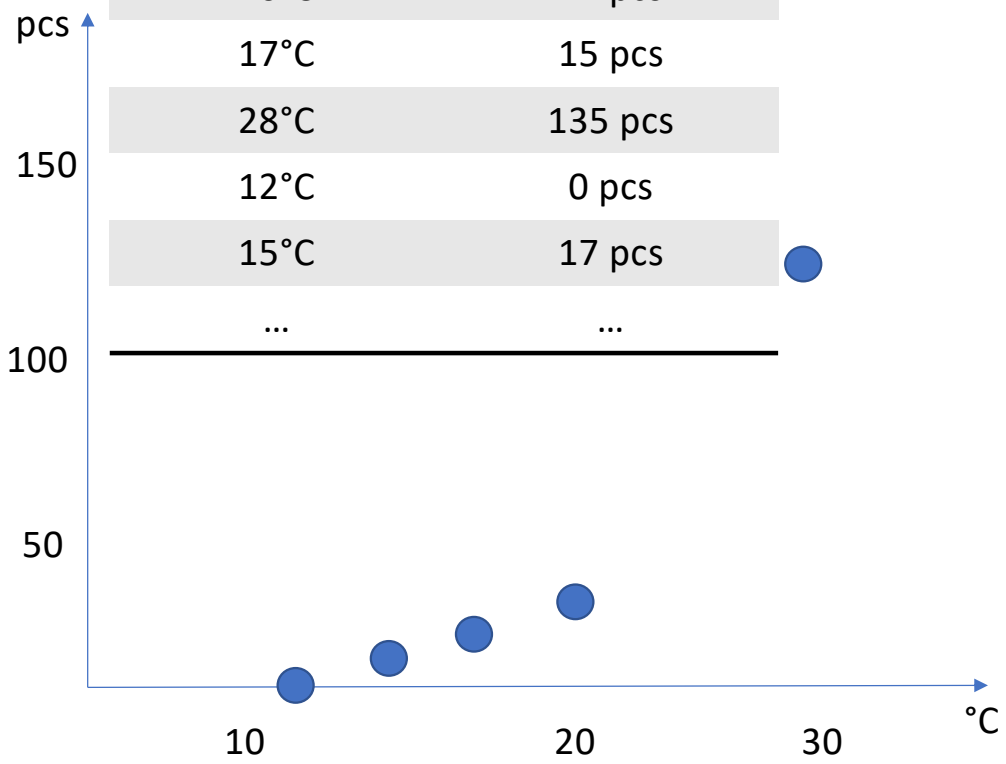
15 17

135 28

0 12

17 15

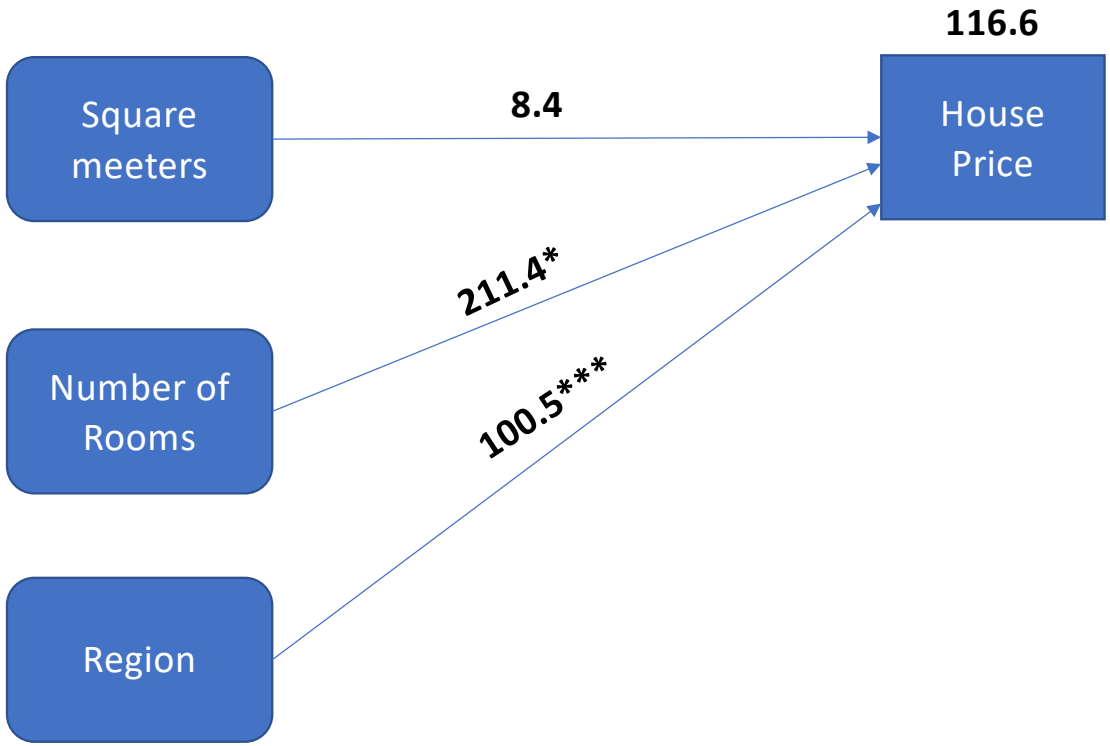
Temperature	Ice Cream sales
20°C	24 pcs
17°C	15 pcs
28°C	135 pcs
12°C	0 pcs
15°C	17 pcs
…	…



b = 8.4; m = -116.6

Y = 8.4*X – 116.6

Reading regression

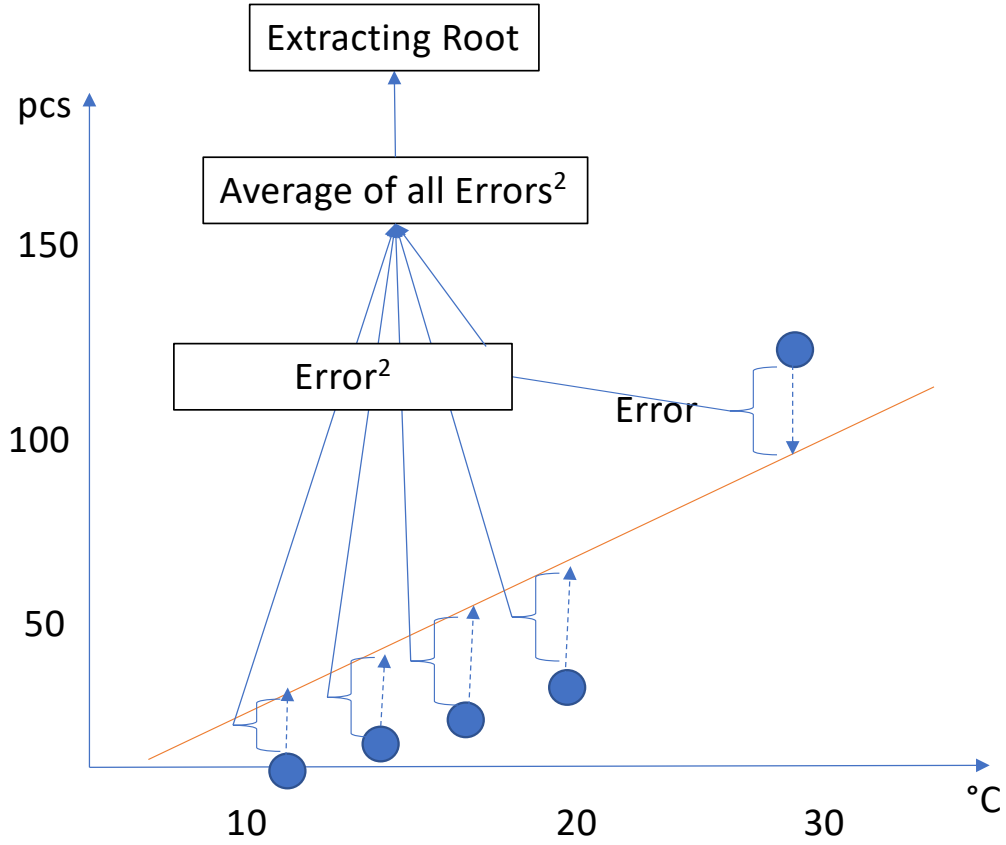


RMSE

Root-mean-square error (RMSE) is a frequently used measure of the differences between values predicted by a model and the values actually observed.

It is used for linear models, predicting continuous values

We need to find "Best Possible Line" – **#RMSE** (Root Mean Squared Error)



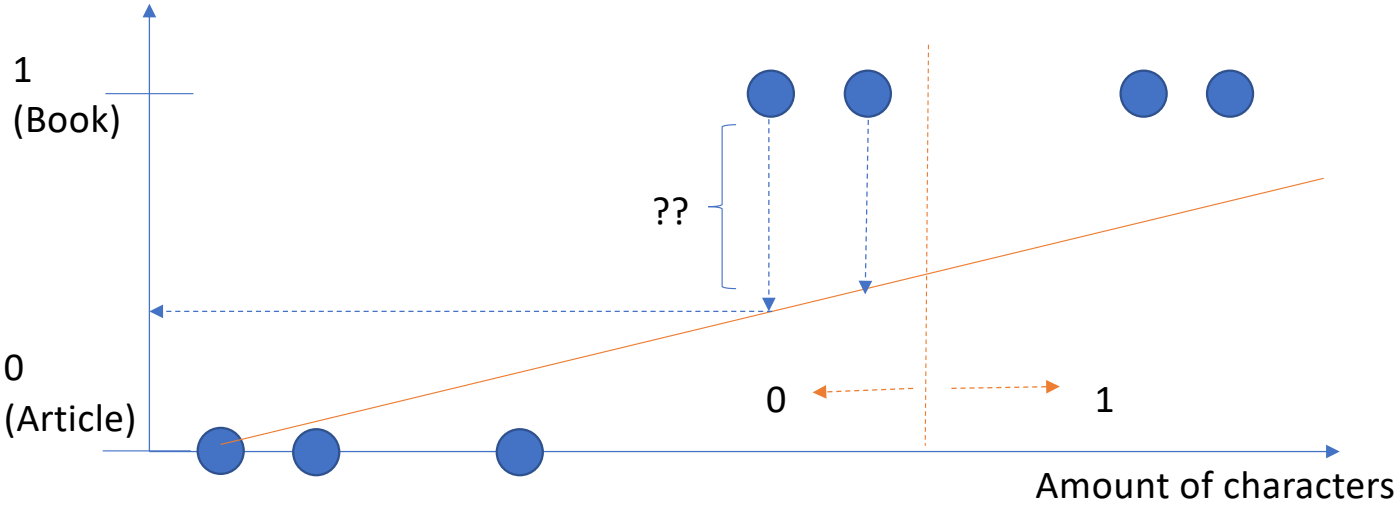
The aim is to find the model that has the lowest Error rate (difference between real and predicted values)

Logistic Regression

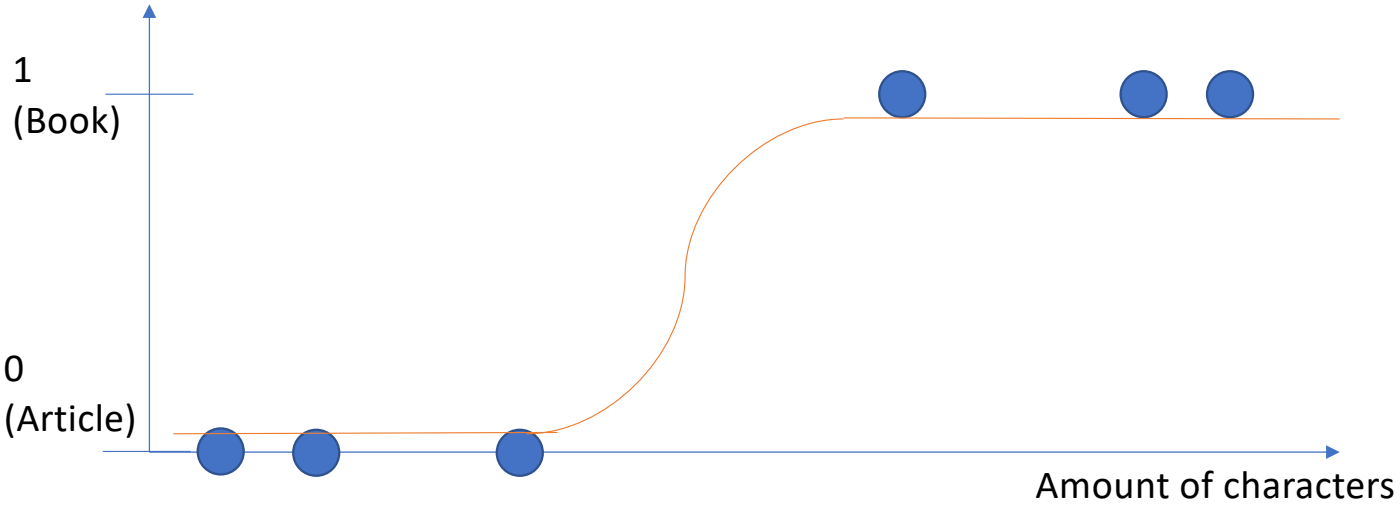
- A method usually used for classification- choosing one among multiple options available, which has no order

What type of Ice Creams will be sold?

- **Chocolate**
- **Vanilla**
- **Strawberry**

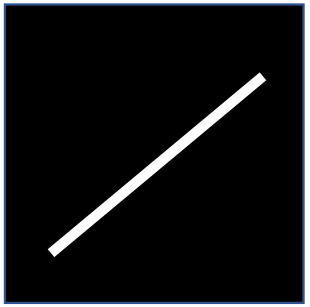


Linear regression
Output: 1 > or <0 0r 0<x<1



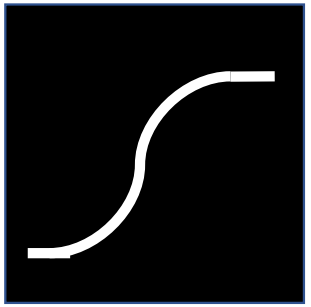
Logistic regression
Output: 1 or 0

Differences in models



Gives continuous number
(any number)

VS.



Gives probability
(percentage)

Probability

- How likely is something to happen?

How likely we get head?

1 out of 2 options

50%

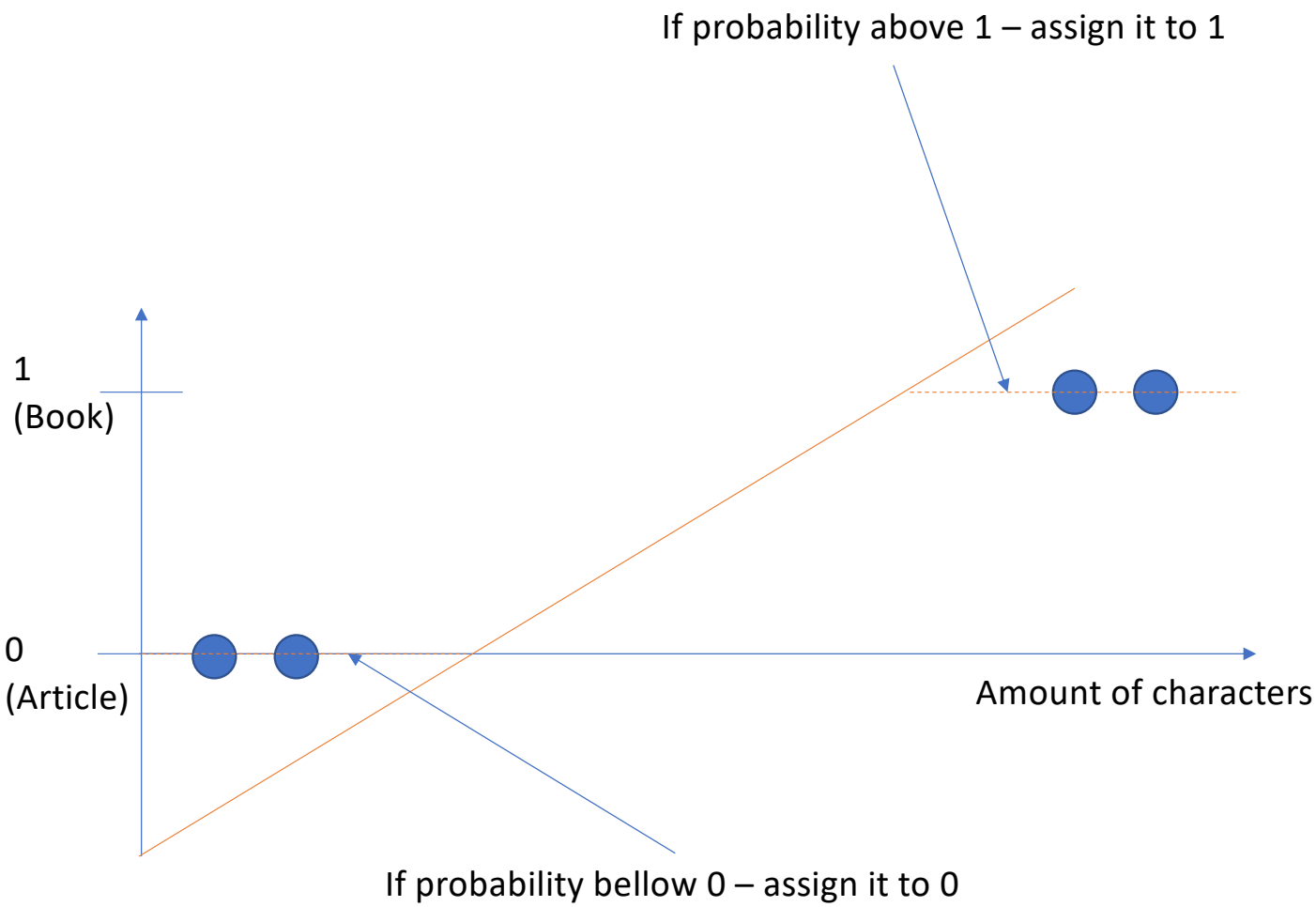
How likely we get tail?

1 out of 2 options

50%

If you would choose ice cream flavor randomly, what probability each of them will get?

- **Chocolate – 0.33 (33%)**
- **Vanilla – 0.33 (33%)**
- **Strawberry – 0.33 (33%)**


If probability above 1 – assign it to 1
1
(Book)
0
(Article)
Amount of characters
If probability bellow 0 – assign it to 0

Mathematical way:
Tow rules to satisfy for transforming linear regression to logistic regression

1. Probability should be always positive (**$p \geq 0$**)

2. Probability should be less than 1 (**$p < 1$**)

1. Probability should be always positive ($p \geq 0$)

Options:

- $|x| \geq 0$ - absolute value
- $x^2 \geq 0$ - squared value
- **$e^x \geq 0$ - exponential value**

$p = e^{m + b*x}$

m + b*x - linear regression formula

e – exponential value equals to 2.72….

But the result still can get over 1

2. Probability should be less than 1 (p < 1)

What about:

Dividing by the same number only slightly larger?

x/x = 1

x/(x+1) < 1

(9/10) = 0.9

(99/100) = 0.99

Exponential value of linear regression

$$Y = \frac{e^{m+bx}}{e^{m+bx} + 1}$$

Exponential value of linear regression

With added small number

Problem, how to code the "**Y**" (outcome)?

We could decide that above 20 pieces – A lot;
and bellow 20 ice creams – not enough

Temperature	Ice Cream sales	Ice cream sales (coded)	Ice cream sales (binary coded)
20°C	24 pcs	**A lot**	**1**
17°C	15 pcs	**Not enough**	**0**
28°C	135 pcs	**A lot**	**1**
12°C	0 pcs	**Not enough**	**0**
15°C	17 pcs	**Not enough**	**0**
…	…		

Calculating "m" and "b" based on the data

$$Y = \frac{e^{m+bx}}{e^{m+bx} + 1}$$

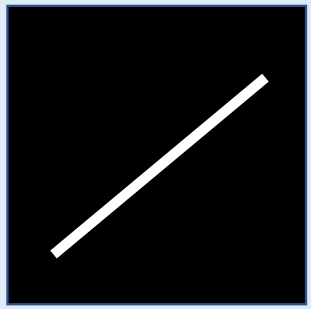
Finding **m** and **b**

Y = 1, 0, 1, 0, 0
X = 20, 17, 28, 12, 15

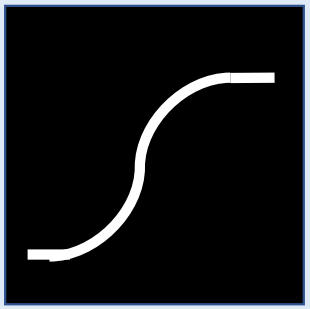
Temperature	Ice cream sales (binary coded)
20°C	**1**
17°C	**0**
28°C	**1**
12°C	**0**
15°C	**0**
…	

Classification trees

Linear Models



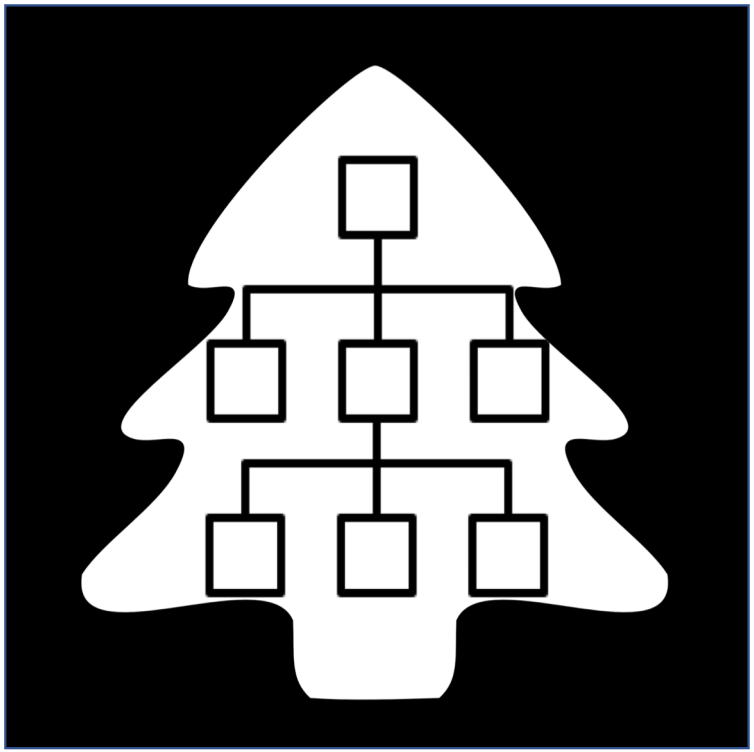
Gives continuous number
(any number)

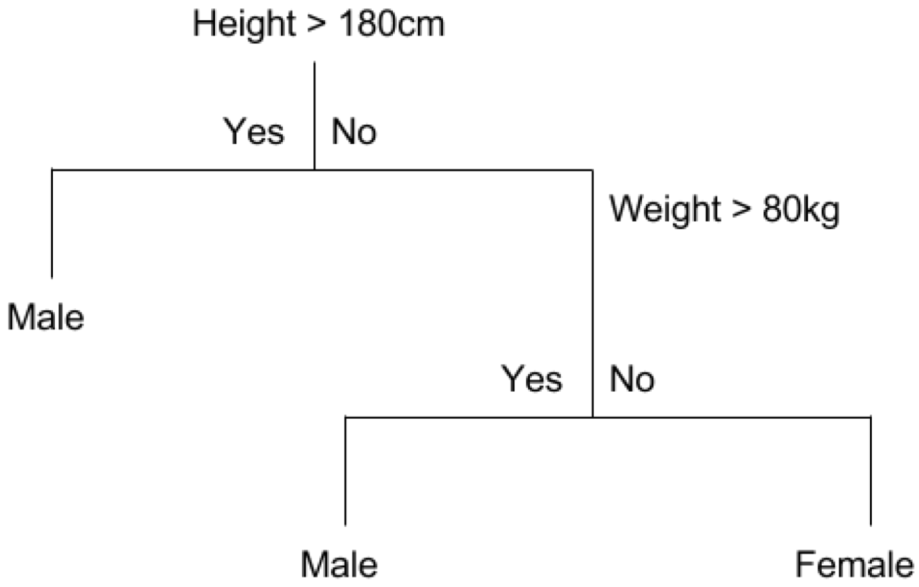


Gives probability
(percentage)

Tree Model

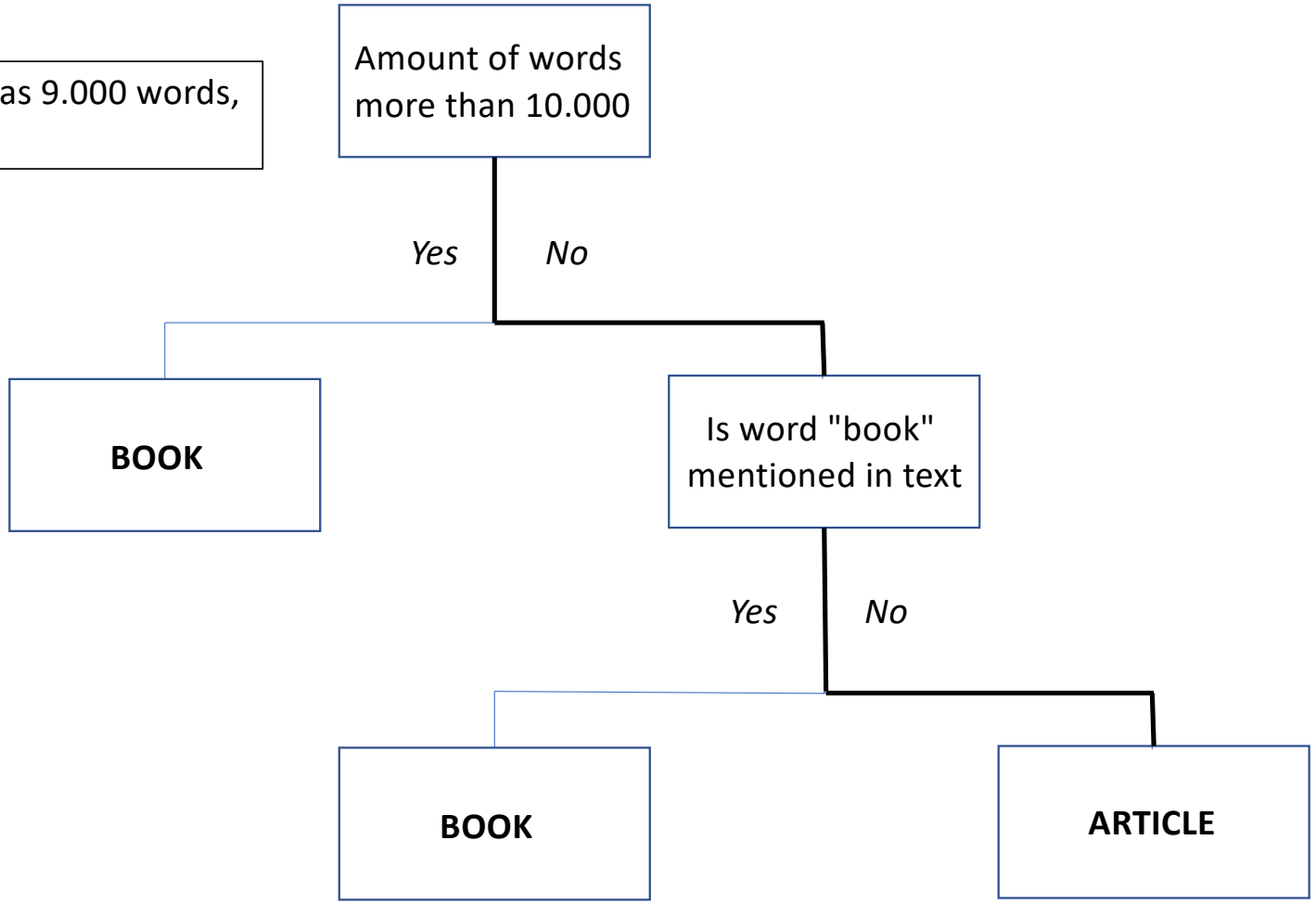
Gives label
(one of options)




Height > 180cm
Yes
No
Weight > 80kg
Male
Yes
No
Male
Female

Classification tree whether it’s a book or an article

If we have a manuscript that has 9.000 words,
Without 'book' in text



Decision tree for selling different flavor ice creams

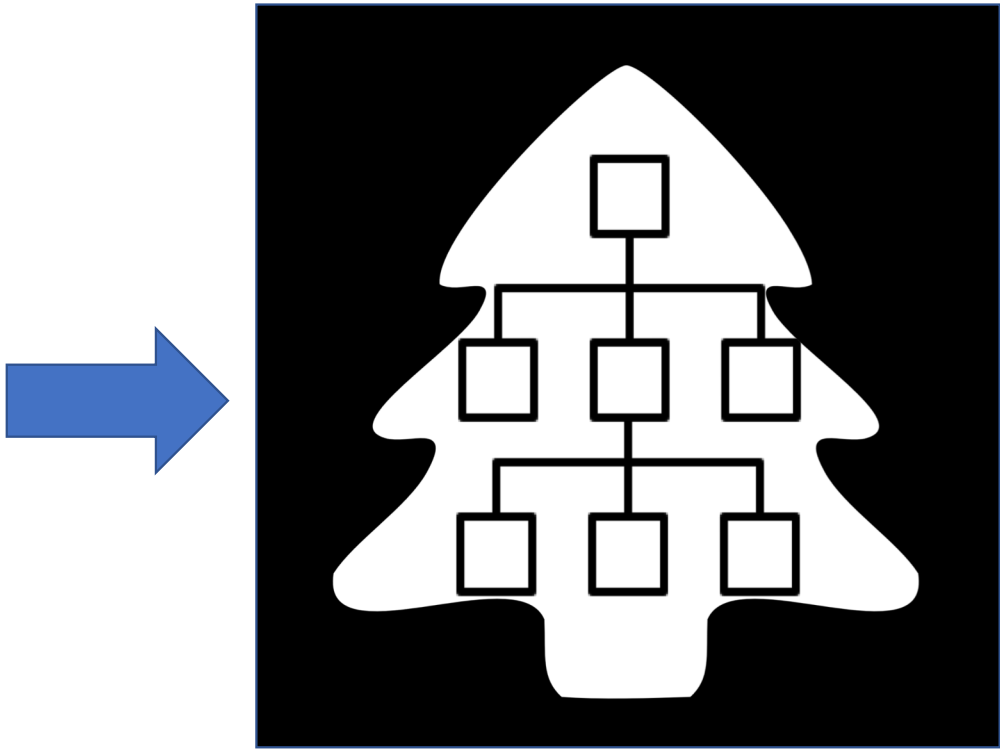
- **Chocolate**
- **Vanilla**
- **Strawberry**

Age	Nationality	Flavor
12	Indian	Vanilla
59	Finnish	Chocolate
28	German	Vanilla
24	Finnish	Strawberry
33	German	Chocolate
16	Finnish	Strawberry

Age or Gender is better predictor?

Aim: learning the "tree" from the data

Temperature	Region	Discount	Ice cream sales (binary coded)
20°C	Kallio	NO	**1**
17°C	City Center	NO	**0**
28°C	Arabia	YES	**1**
12°C	Kallio	YES	**0**
15°C	Kilo	NO	**0**
…	…		



Instead of a distance between points, for tree we can measure error rate

$$\text{Error rate} = \frac{incorrect\ predictions}{all\ possible\ predictions}$$

We had to guess on 100 manuscripts whether it's a book or an article and we guessed right on 68, thus:

$$\text{Error rate} = \frac{32}{100} = 32\%$$

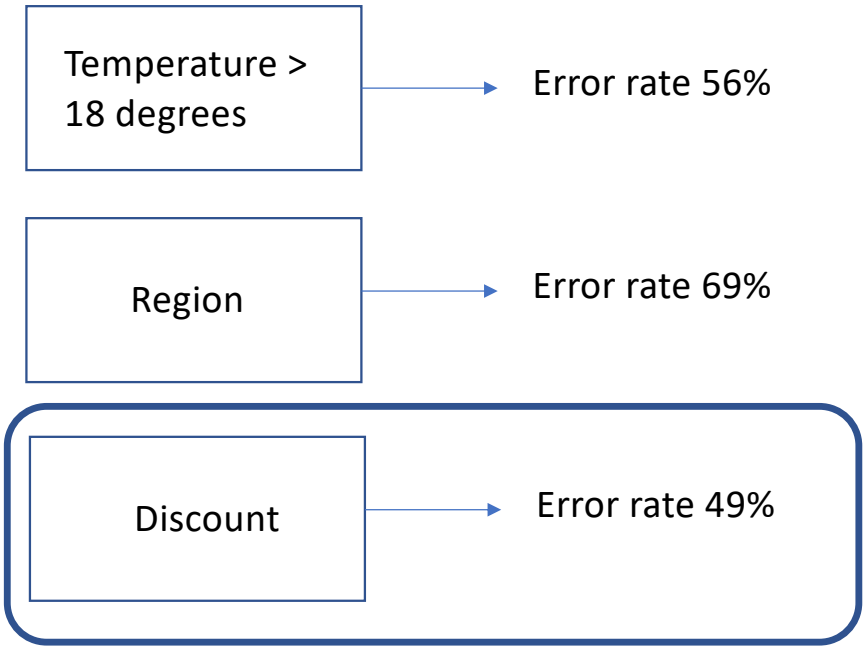
Simplest way to learn Classification Tree

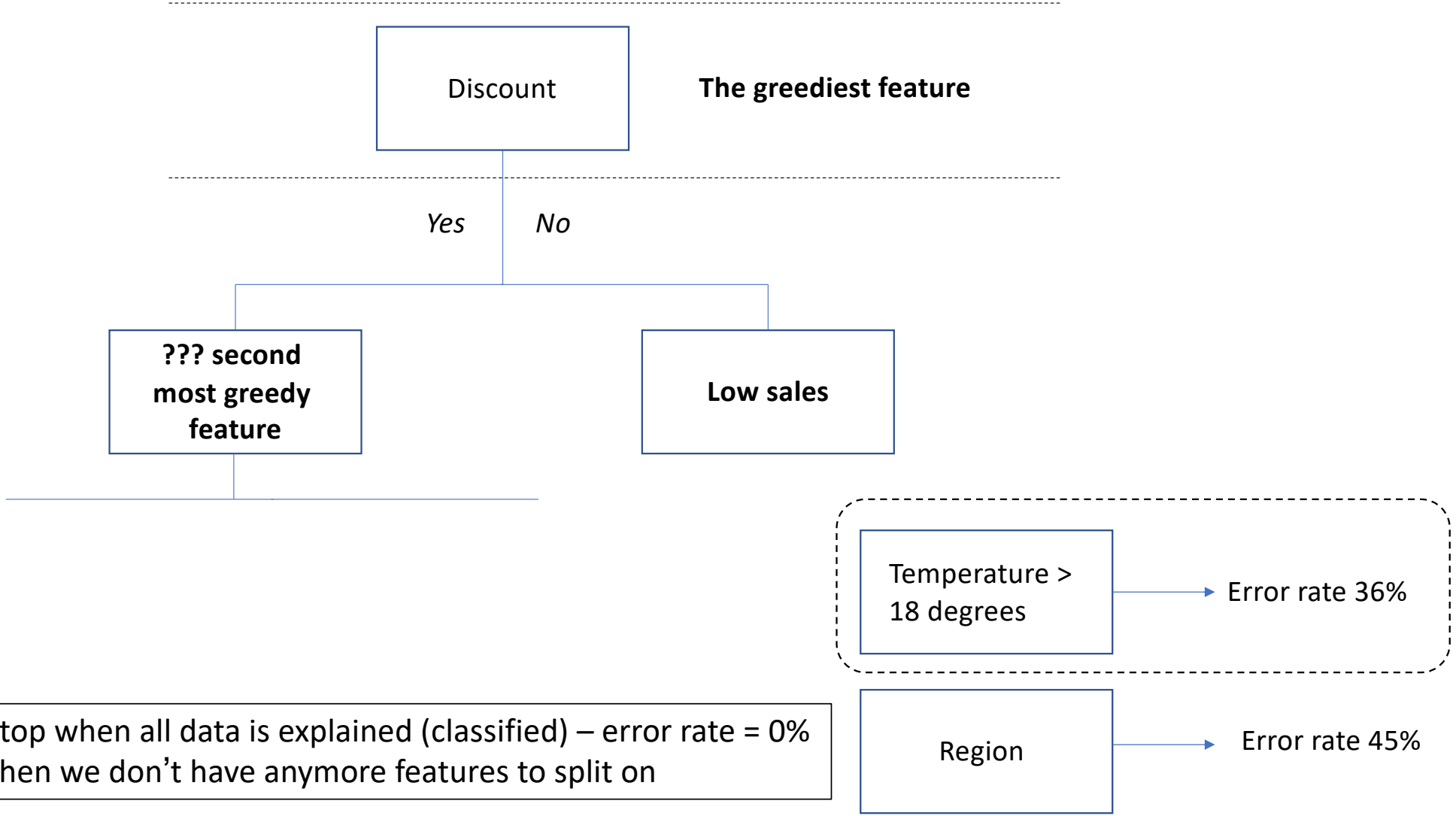
Recursive Greedy algorithm

Finding the most greedy feature

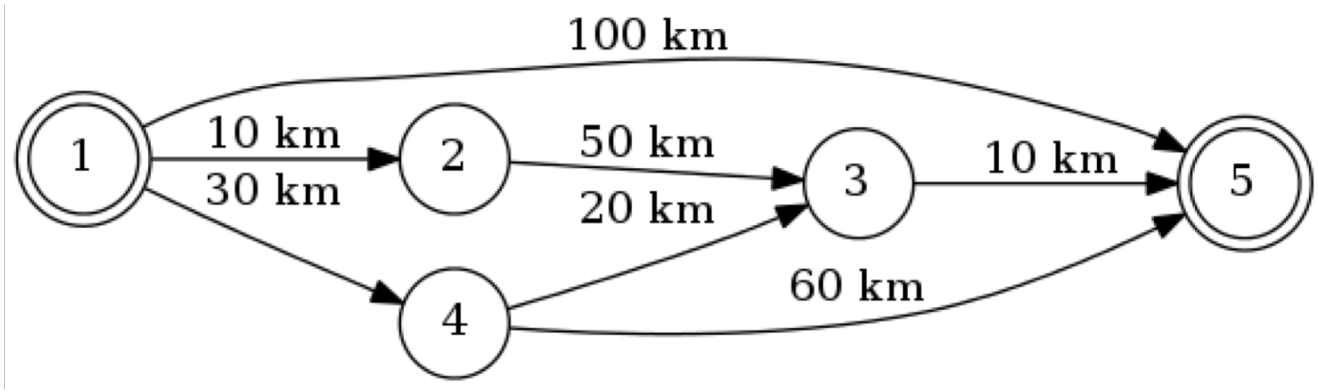
- If we divided all data based on one feature (creating a single branch), what best error rate will it be?

Temperature	Region	Discount	High Ice cream sales (binary coded)
20°C	Kallio	NO	**1**
17°C	City Center	NO	**0**
28°C	Arabia	YES	**1**
12°C	Kallio	YES	**0**
15°C	Kilo	NO	**0**
…	…		

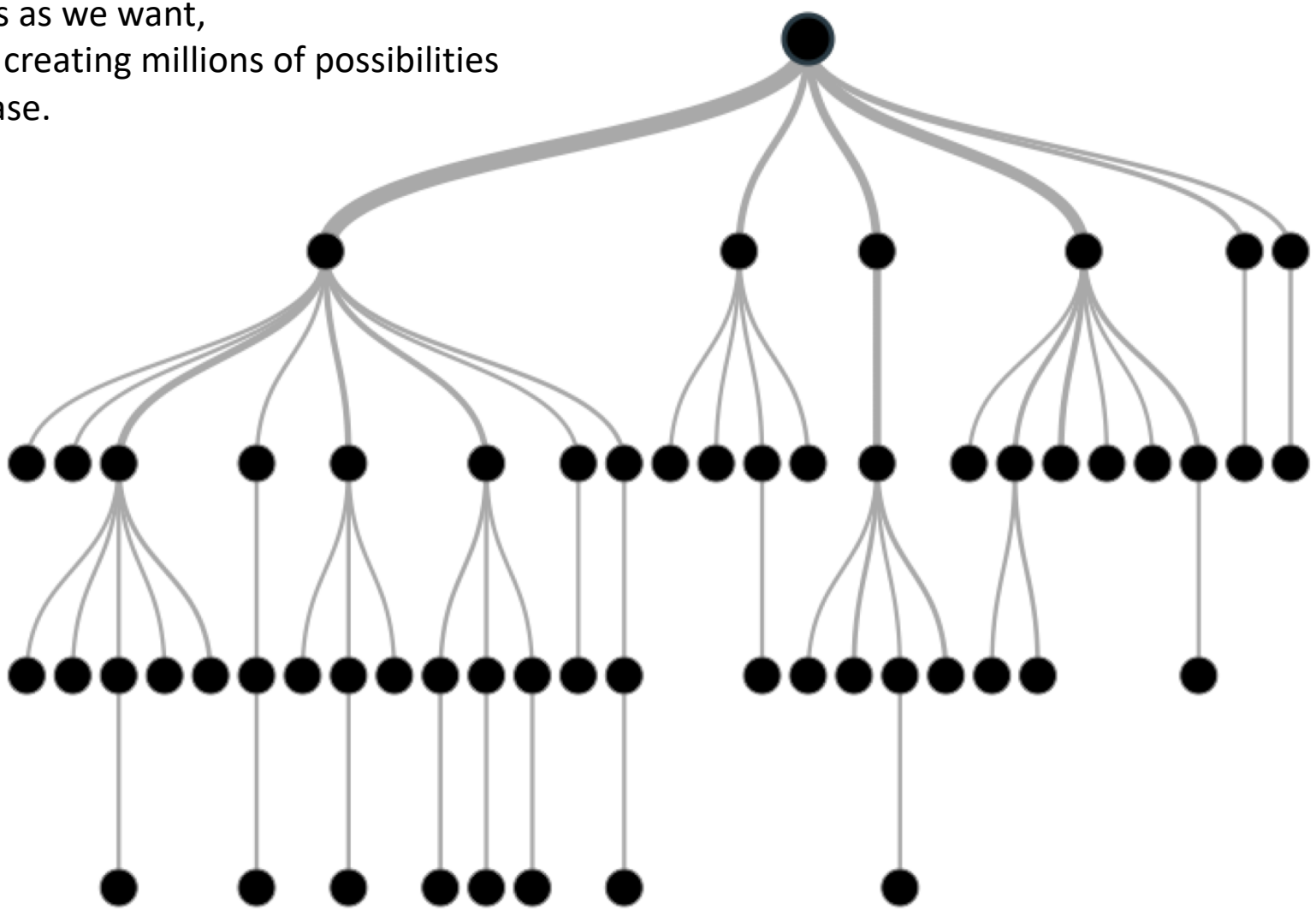



Discount
The greediest feature
Yes
No
??? second most greedy feature
Low sales
Temperature > 18 degrees
Error rate 36%
Region
Error rate 45%
We stop when all data is explained (classified) – error rate = 0%
Or when we don't have anymore features to split on

Calculated fastest way with and without Greedy algorithm

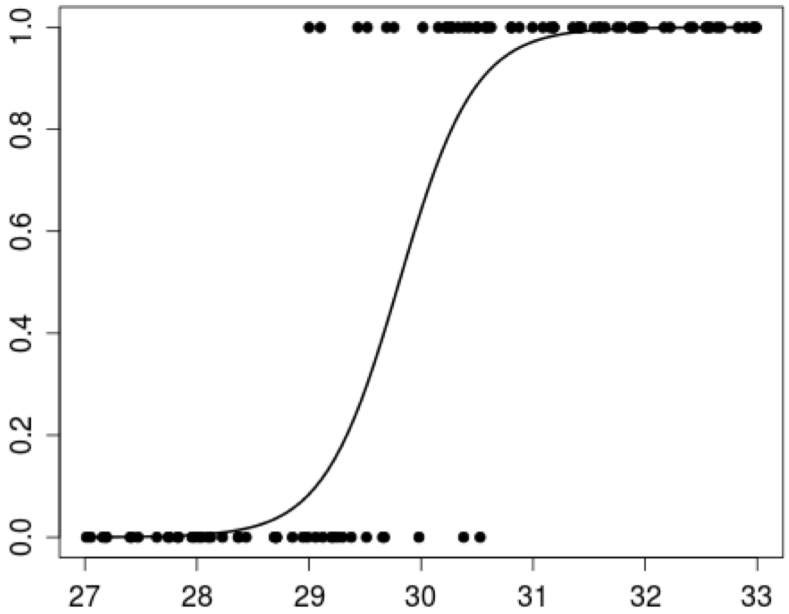


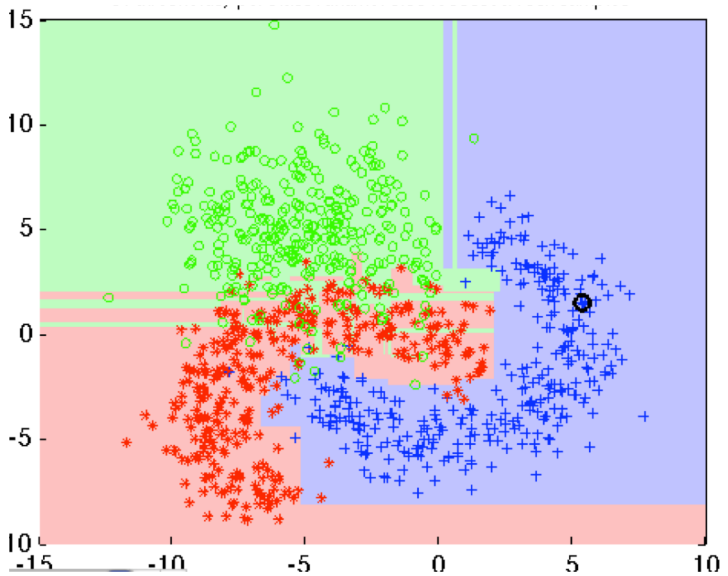
We can create as much branches as we want,
It can be thousands of branches creating millions of possibilities
and explaining every separate case.



Classification trees VS. Logistic regression

- Logistic regression have a better performance on simpler problems, and less probable for overfitting.
- Classification trees can be scaled to become very complex algorithms

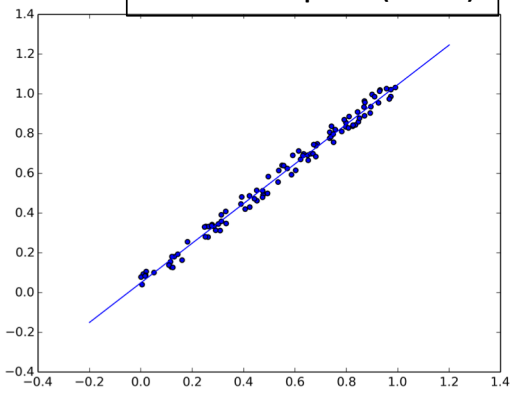




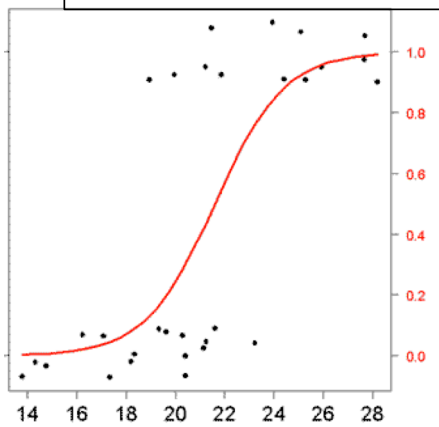
With each additional variable complexity is increasing

With each possibility of the shape complexity is increasing

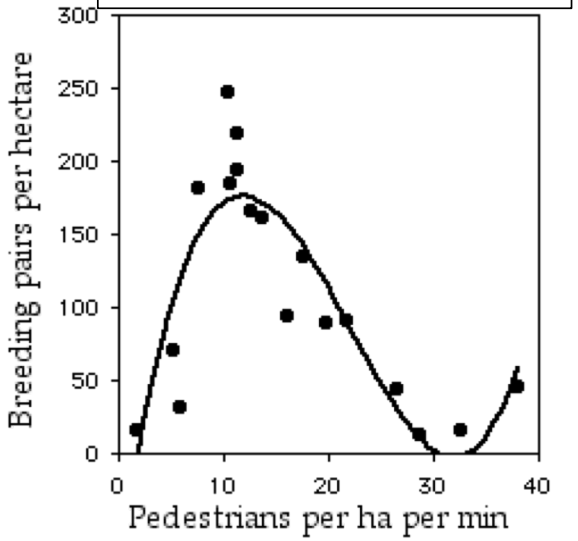
2D linear plot (m*X)



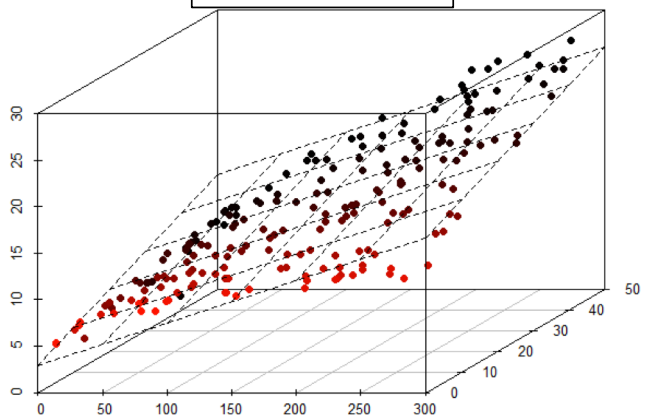
2D logistic plot ($m*x^2$)



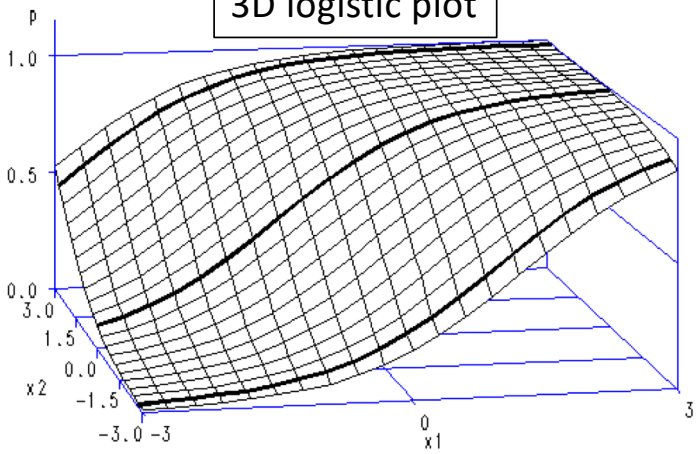
2D polynomial plot ($m*x^3$)



3D linear plot



3D logistic plot



4D, 5D, x^5, x^{10}

????

"Best Line" for Logistic Regression

That was for "continuous" values (linear regression).

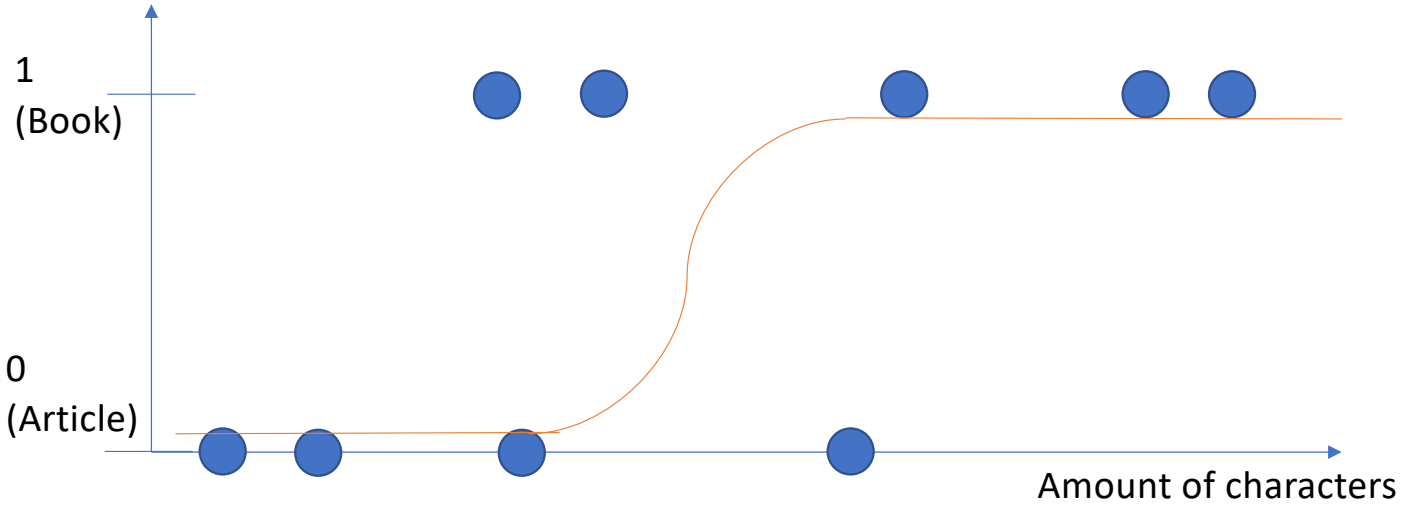
How we can calculate whether the model is good for "categorical" values – Logistic Regression

Accuracy

How many we hit, out of all possible hits?

In the example "whether it's a book or an article" our accuracy would be:

$$Accuracy = \frac{correct\ answers}{All\ answers} = \frac{6}{9} = 67\%$$

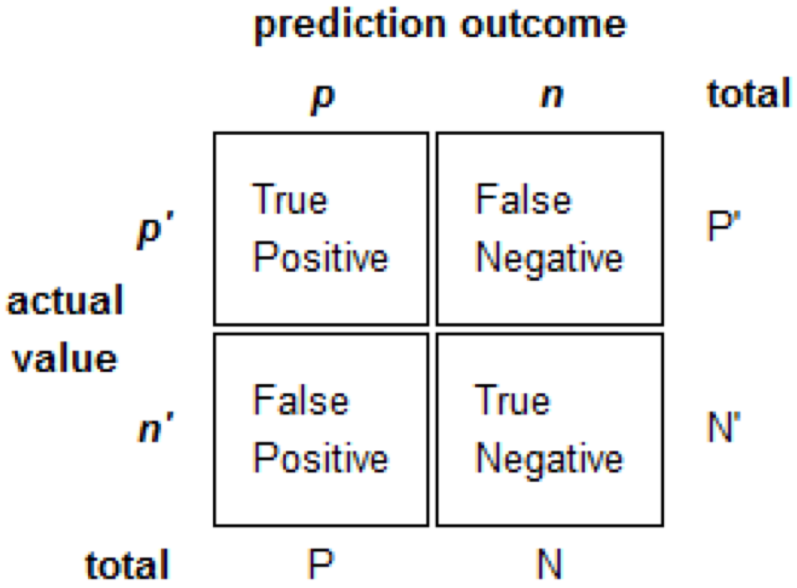


When measuring Accuracy is not enough

- If we have relatively equal amount of different categories (e.g. out of 100 manuscripts 56 are articles and 44 are books) then Accuracy is a pretty nice measurement to determine Best Fit Model.

- But say we have 90 articles and 10 books. Our model guess correct with 80% precision – which looks like a decent result. However, if we just choose all 100 manuscripts as articles – result would be 90% accurate. We end up with completely useless model that gives better accuracy? How to tackle this situation?

#Confusion_matrix
allows estimate distributions of guesses



$$Precision = \frac{tp}{(tp+fp)}$$

How many real 1s we identified, out of all 1s we predicted

$$Recall = \frac{tp}{(tp+fn)}$$

Out of all cases where answer was 1, how many we identified as 1

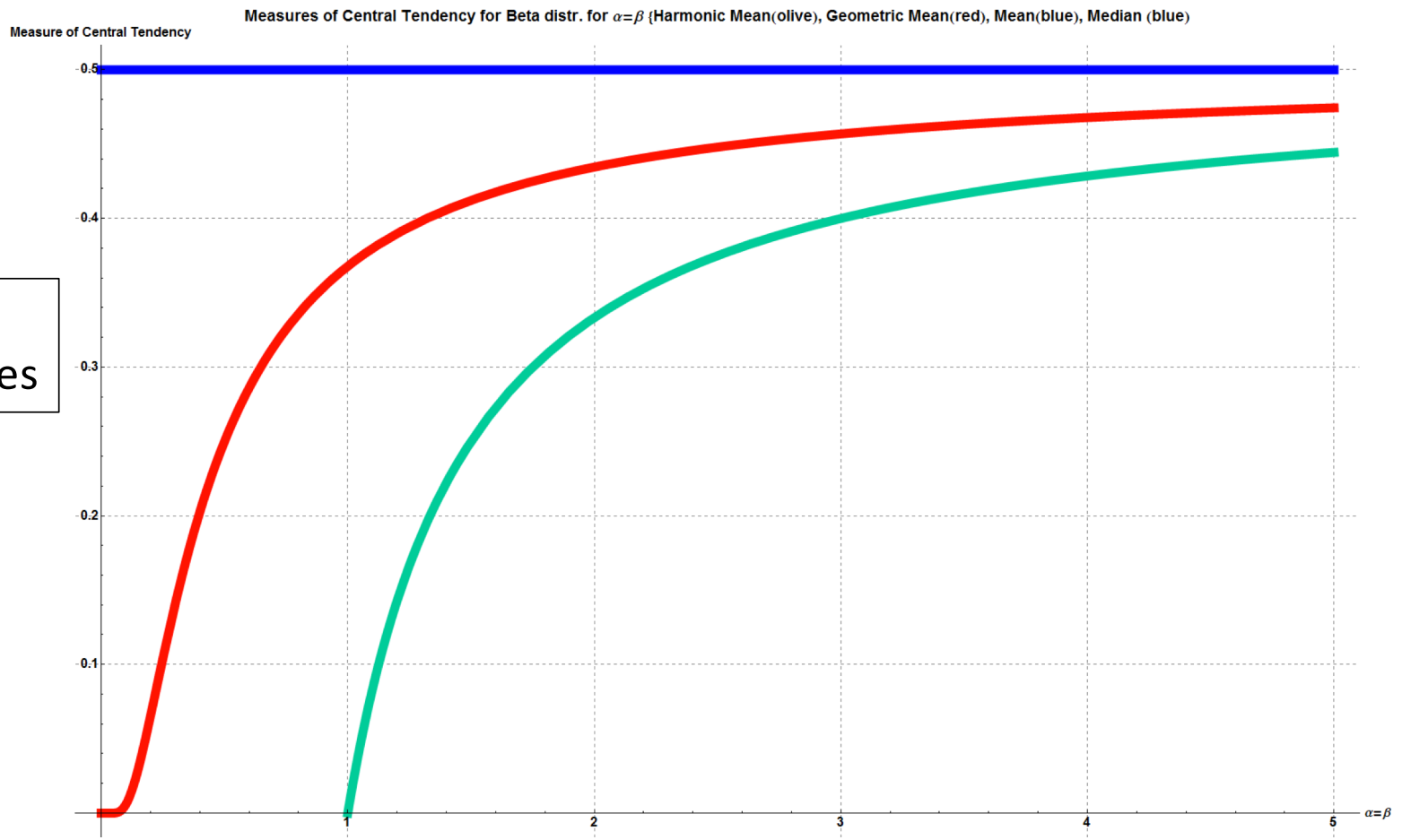
#Confusion_matrix on "Book or Article" example

	Predicted Book	Predicted Article
Real Book	3 (TP) 	2 (FN) 
Real article	1 (FP) 	3(TN) 

When you classify spam messages, we are ok with higher false negatives thus we aim at – **high Precision**
When we predict whether patient has cancer, we are ok with false positives, as we want to find locate all patients – **high Recall**

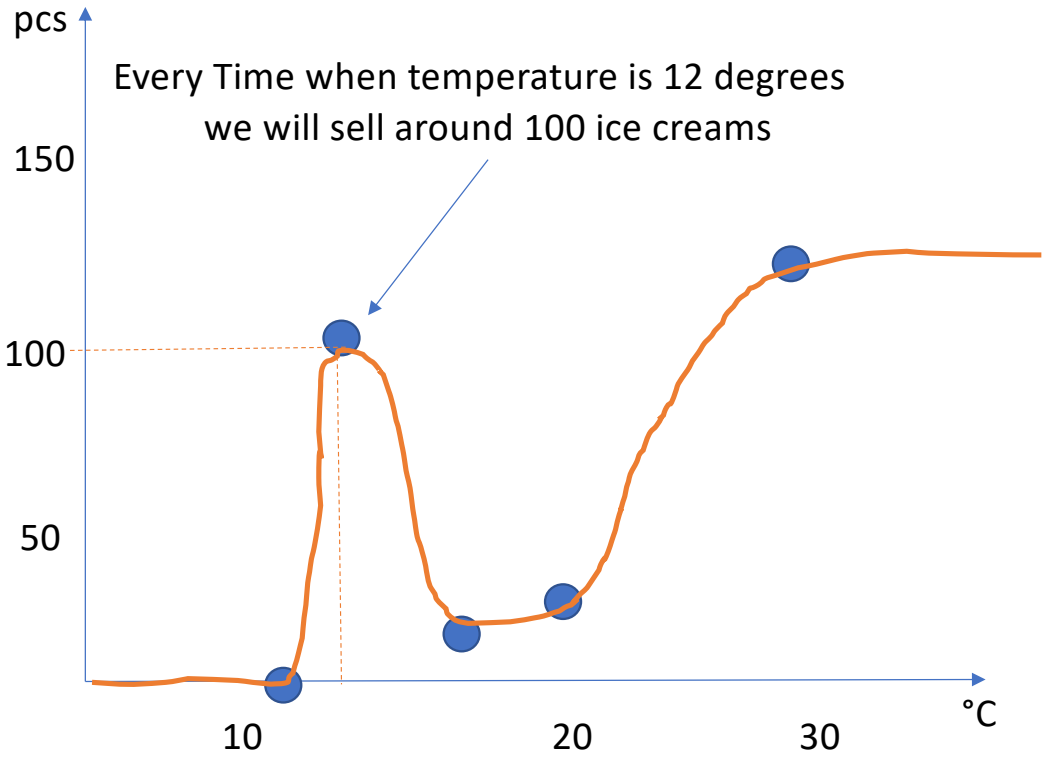
To combine Precision and Recall to one score

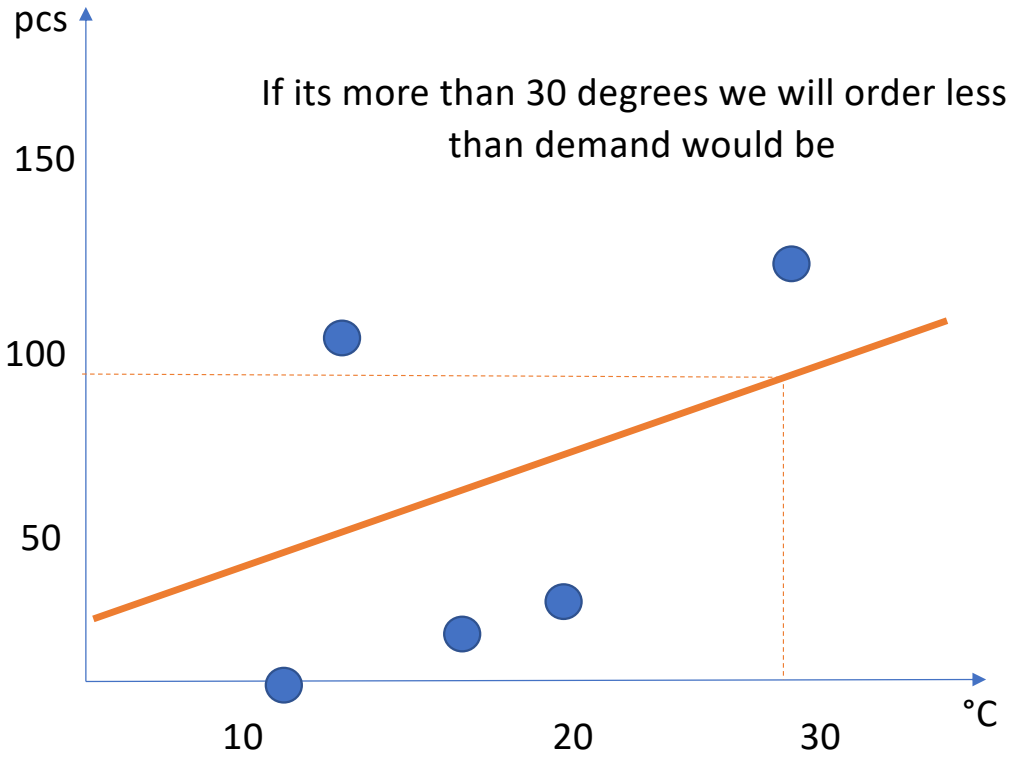
Harmonic Mean (F1)
for penalizing small values



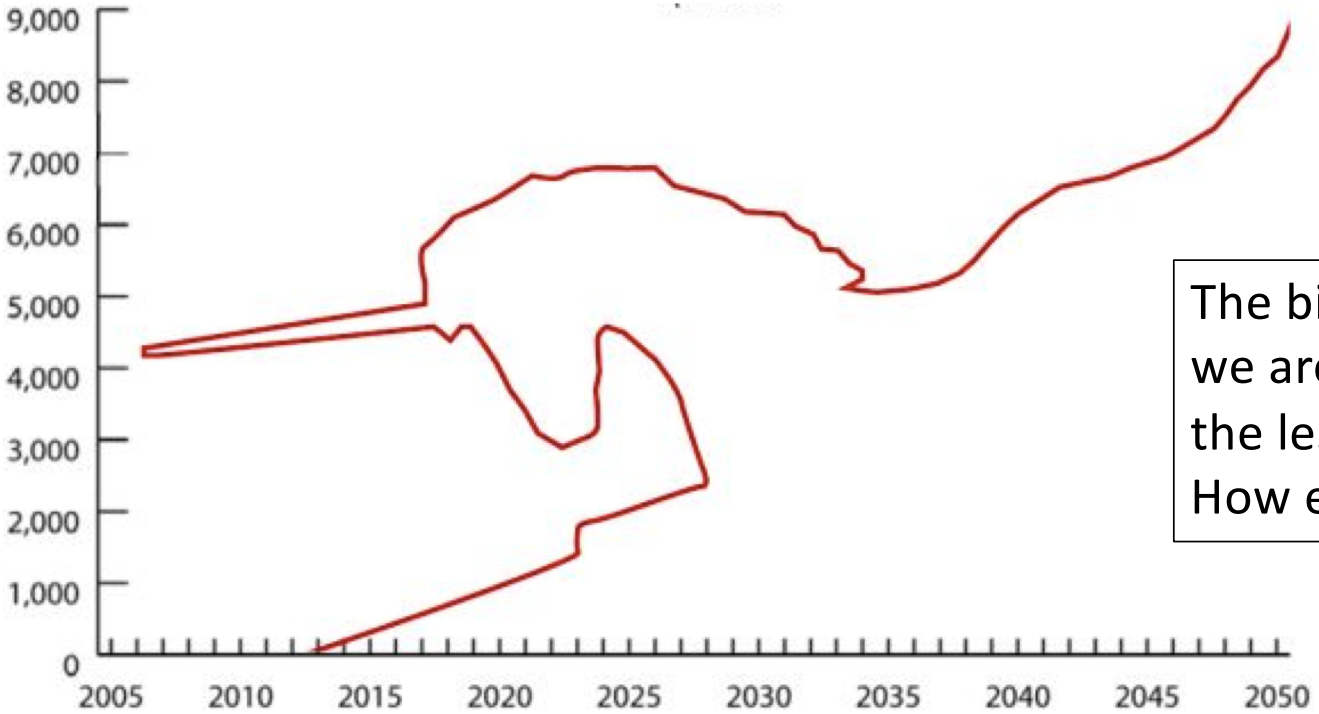
Overfitting
Underfitting

Ice Cream case





- If the model is Overfitted its applicable only on that data on which we built the model, and its accuracy will be quite high (close to 100%)



The bigger the dataset on which
we are training our model,
the less likely the model will get Overfitted.
How else we can deal with Overfitting?

Test and train

Train the model and after Test its result with the data the tool haven't seen yet:

- Like in high school. You learning how to solve mathematical equations by solving, while knowing what the answer should be. During exam, you see only equations but not the answers.

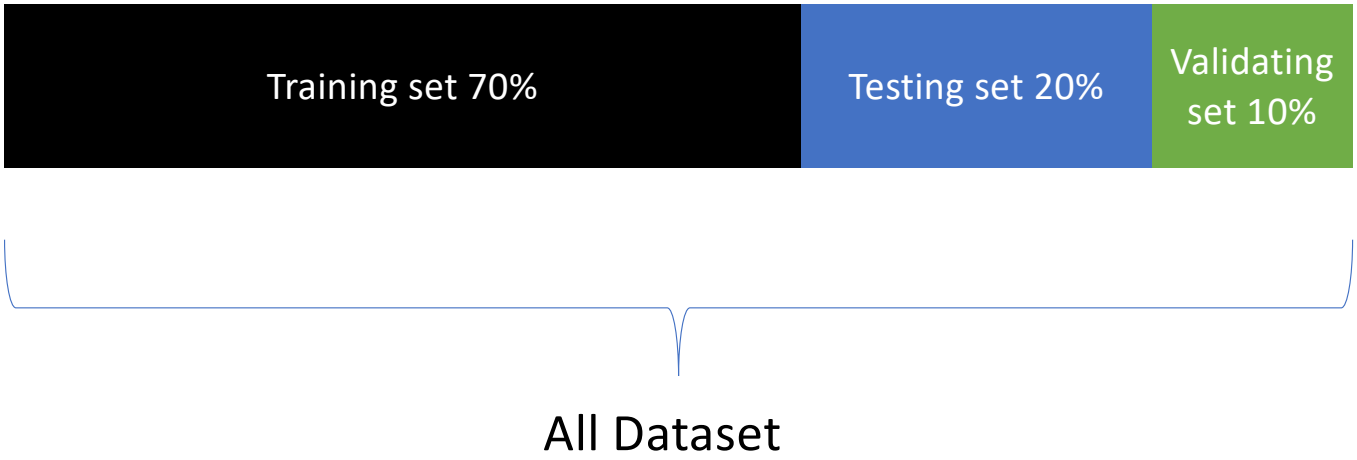
But we also want to make sure that solution is valid on a data the computer haven't seen before (testing data set)

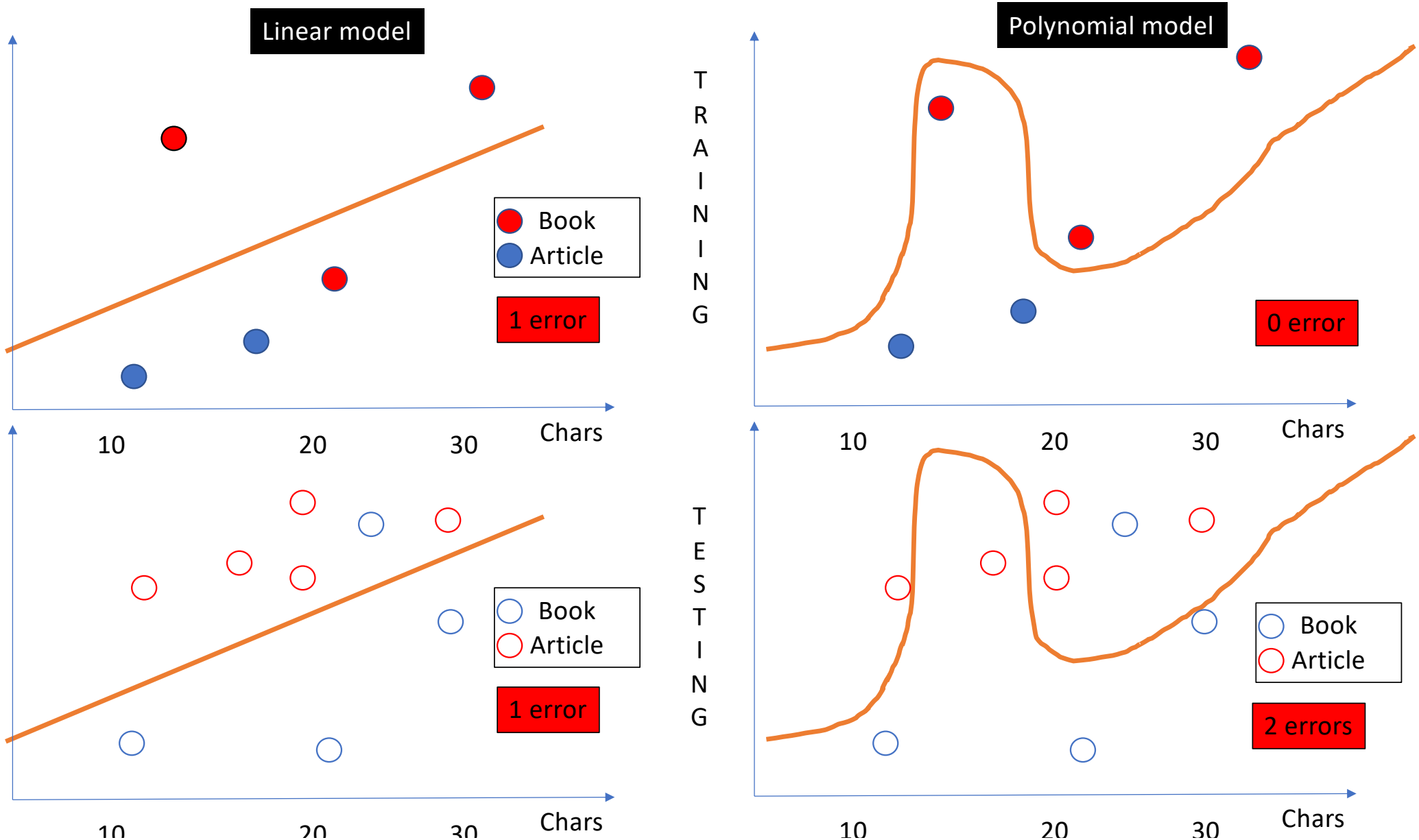
- If we keep taking the same exam, we will memorize question and answers, thus we will train on the exam data as well.

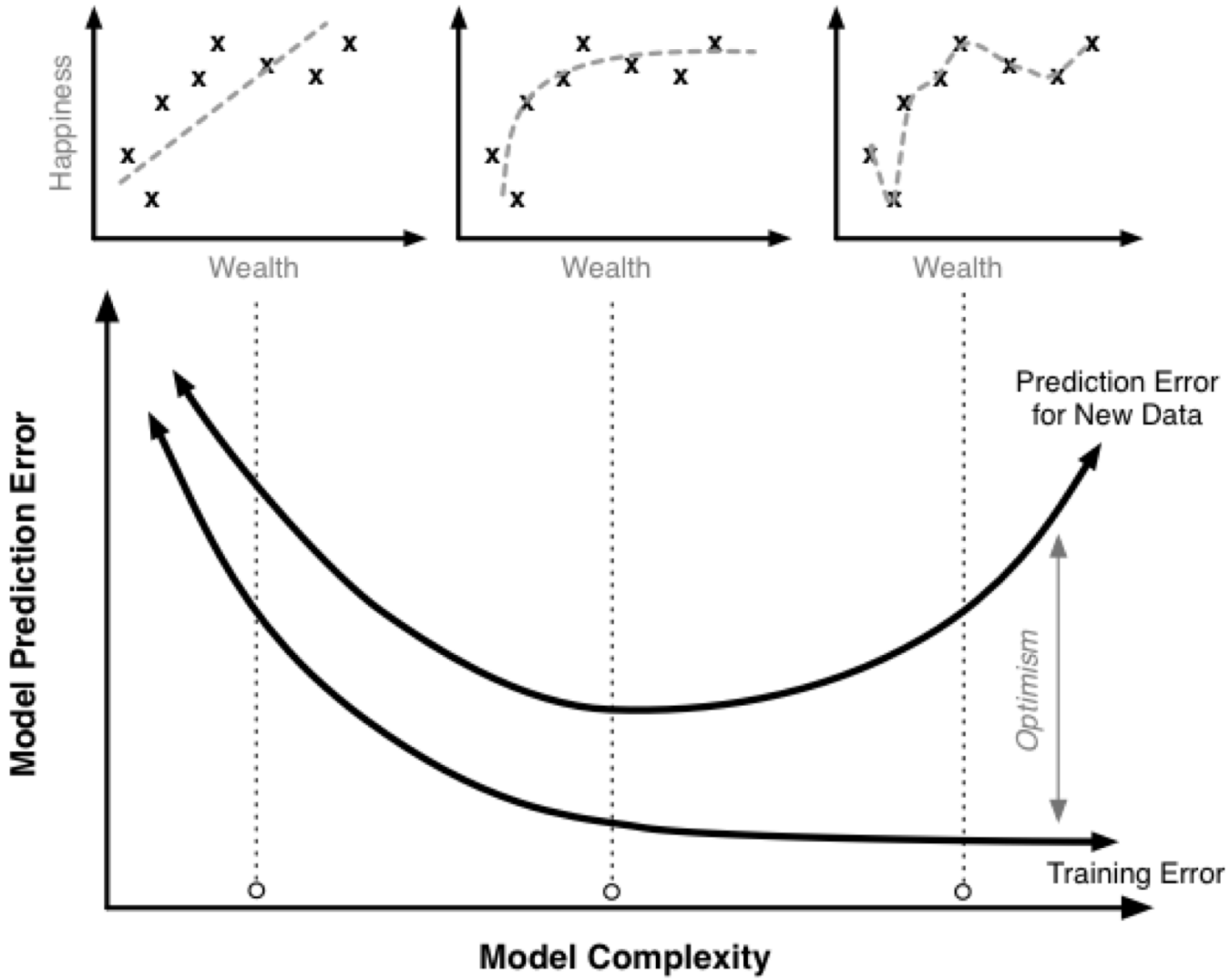
The aim of machine learning is to create a model based on which unseen problems can be solved, not to relate the problem to exact example.

- "field of study that gives computers the ability to learn without being explicitly programmed." - Arthur Samuel (1959)

Then we need final Validation data set




Linear model
Polynomial model
TRAINING
TESTING
Book
Article
1 error
0 error
2 errors
10
20
30
Chars


Happiness
Wealth
Wealth
Wealth
Model Prediction Error
Prediction Error
for New Data
Optimism
Training Error
Model Complexity

Errors are welcome, we don't want 100% accuracy in training data

n-fold cross validation and random falsification

Bank and bad customers case

- Only 1% percent will be bad – hard for ML to identify that without sampling

- Explanatory – you get only variable/features coded, normalized in digits, and variables has no meaningful name (such as var_1). You cannot do any hypotheses engineering only to trust machine on this. Machine can find a very good way to predict bad customers, which can be applied later with good success ratio (explain). But the researcher won't have any understanding why the things are happening and what influences them (explore)

Machine learning problem

The main reason why gradient descent is used for linear regression is the computational complexity: it's computationally cheaper (faster) to find the solution using the gradient descent in some cases.

- If there are more than one variable (e.g. 10.000) and many observations (e.g. 10.000), least squares method can be time consuming and don't fit into operative memory (e.g. 80gb)?
- If we are not sure what kind of function (model) could be used to predict?

Problem	Tool	Measurements
 How much to ask for a house	 Regression	Least squared error
 What is this book about	 Topic modelling	F1 score
 Are my customers happy with products	 Sentiment analysis	Confusion matrix
	 Network analysis	Internal/external validity

Further reading

- R^2

  http://blog.minitab.com/blog/adventures-in-statistics-2/regression-analysis-how-do-i-interpret-r-squared-and-assess-the-goodness-of-fit
- RMSE
- Confusion matrix
  - https://tryolabs.com/blog/2013/03/25/why-accuracy-alone-bad-measure-classification-tasks-and-what-we-can-do-about-it/
- Gradient Descent
  - https://www.youtube.com/watch?v=BR9h47Jtqyw

Further Readings

- Machine learning

  https://www.youtube.com/watch?v=IpGxLWOIZy4

  http://digitalhumanities.org:8081/dhq/vol/3/2/000041/000041.html

  https://medium.com/machine-learning-for-humans/why-machine-learning-matters-6164faf1df12

  https://medium.com/@v_maini/supervised-learning-740383a2feab

  https://medium.com/@v_maini/supervised-learning-2-5c1c23f3560d

  https://medium.com/@v_maini/supervised-learning-3-b1551b9c4930

- Logistic Regression

  https://machinelearningmastery.com/logistic-regression-for-machine-learning/