Machine Learning

Topic 8. Lecture 8

Classical machine learning. Supervised learning. Regression

Yury Sanochkin

ysanochkin@hse.ru

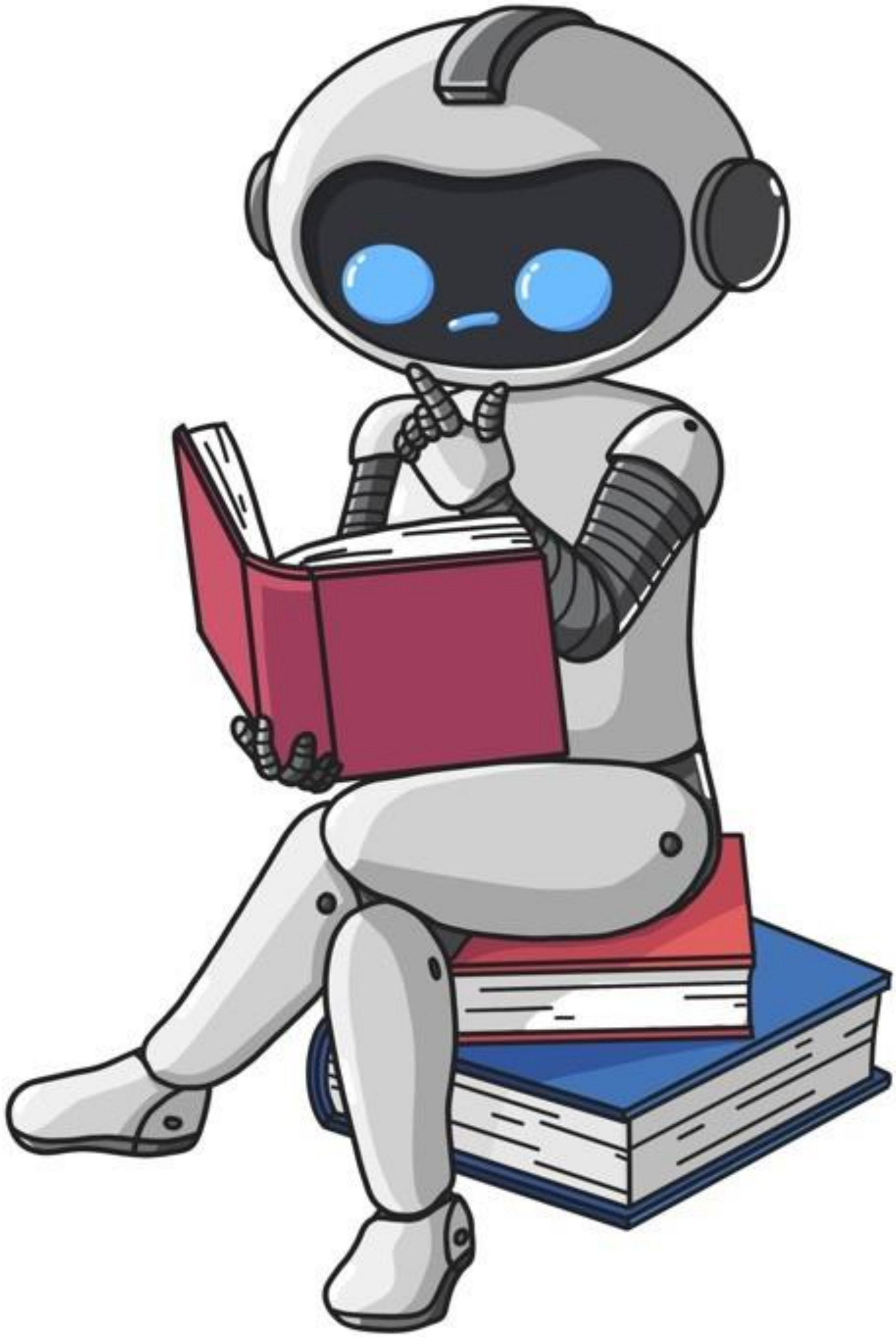
NRU HSE, 2025

Machine Learning

- Let's remember what the concept of classical machine learning is all about!

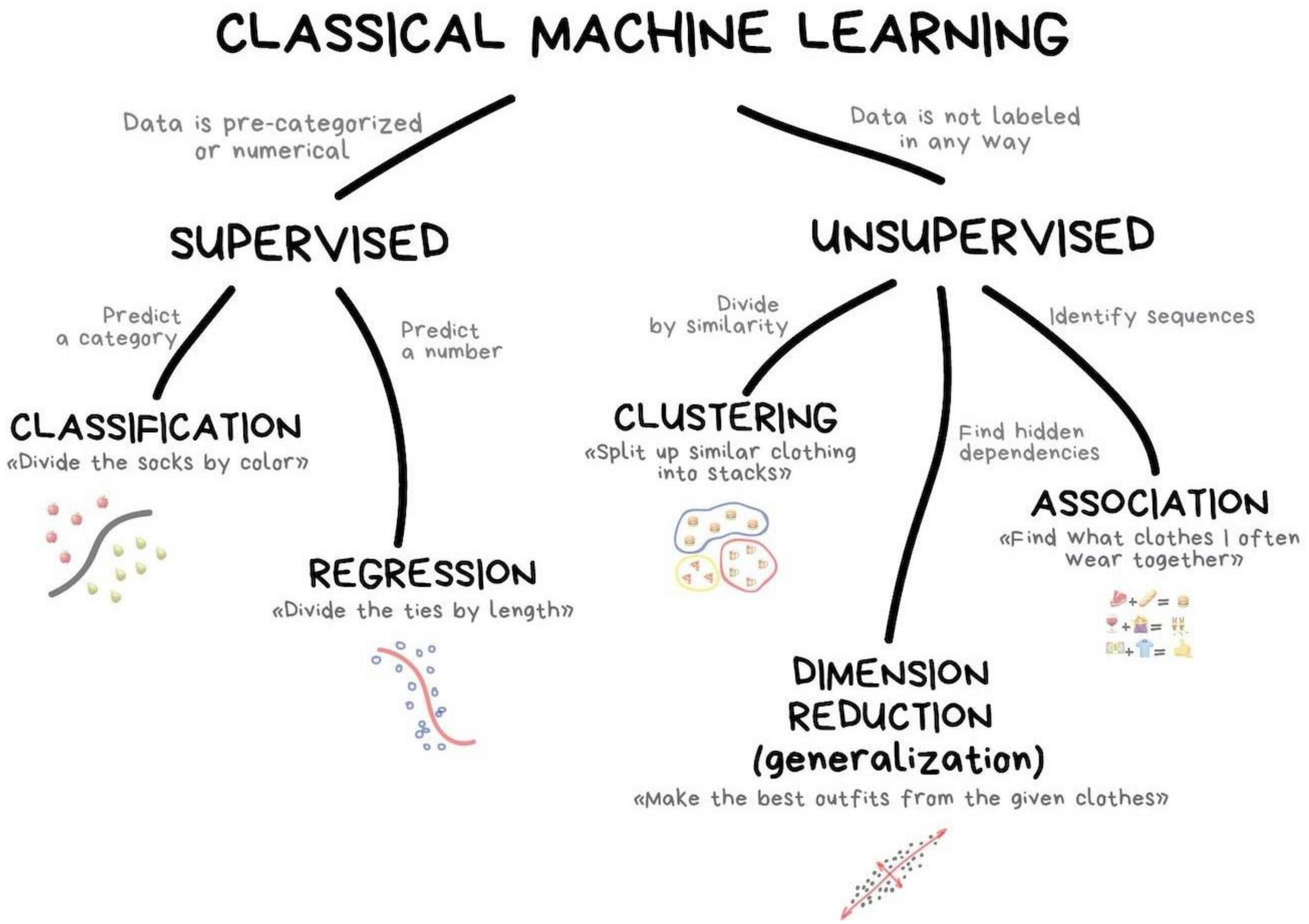
Machine Learning

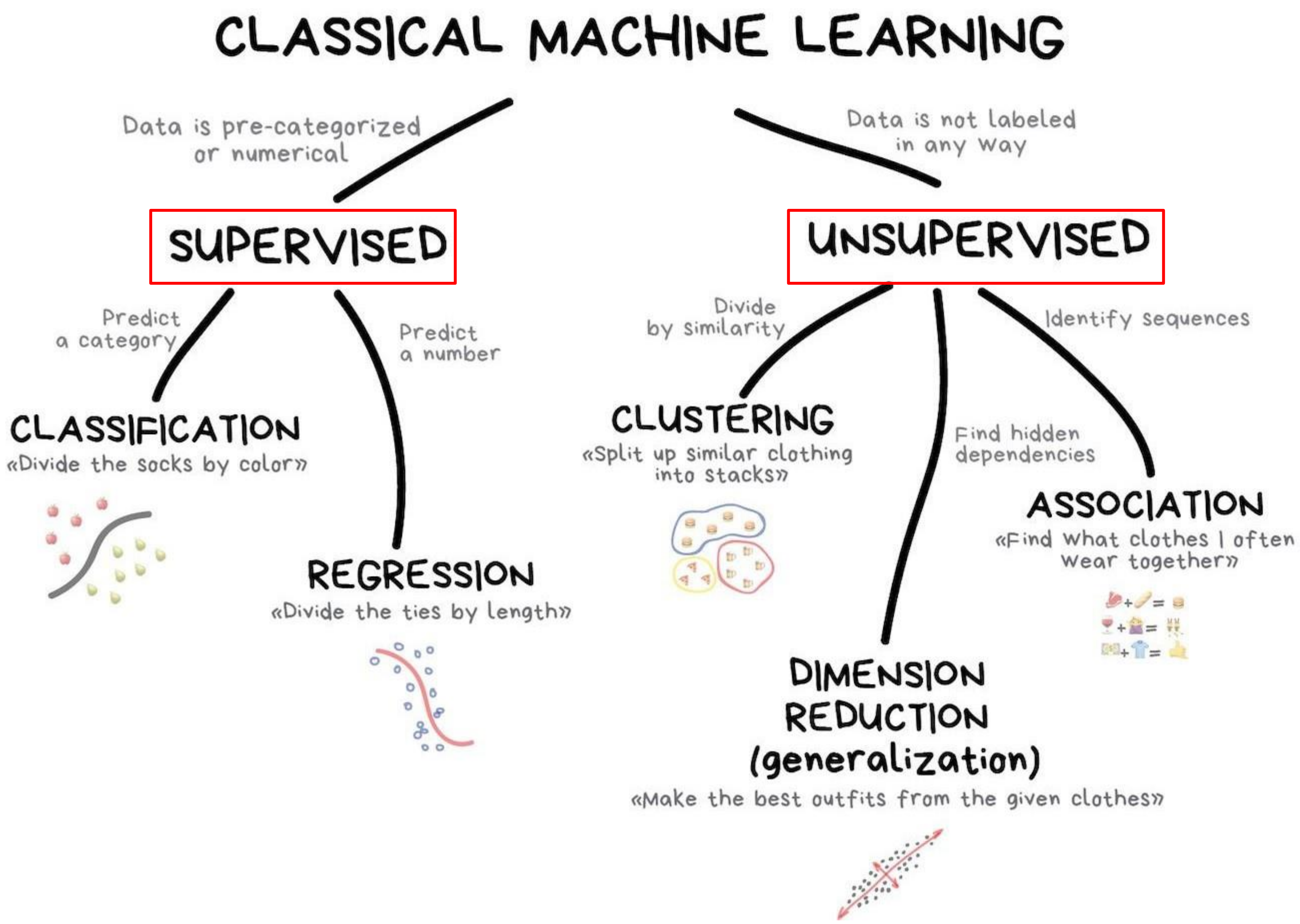
The science of finding patterns in data using a computer and mathematics.



Machine Learning

- What two categories can we divide classical machine learning tasks into?


CLASSICAL MACHINE LEARNING
Data is pre-categorized or numerical
Data is not labeled in any way
SUPERVISED
UNSUPERVISED
Predict a category
Predict a number
CLASSIFICATION
«Divide the socks by color»
REGRESSION
«Divide the ties by length»
Divide by similarity
Find hidden dependencies
Identify sequences
CLUSTERING
«Split up similar clothing into stacks»
ASSOCIATION
«Find what clothes I often wear together»
DIMENSION REDUCTION (generalization)
«Make the best outfits from the given clothes»

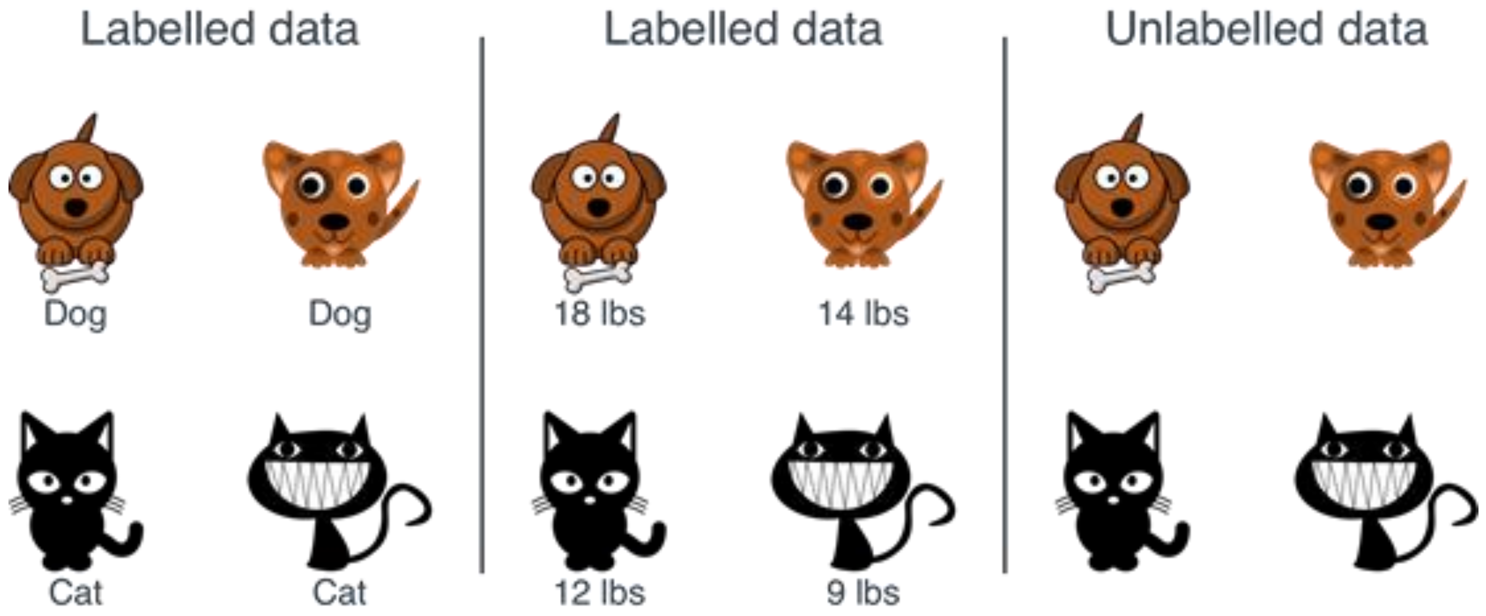


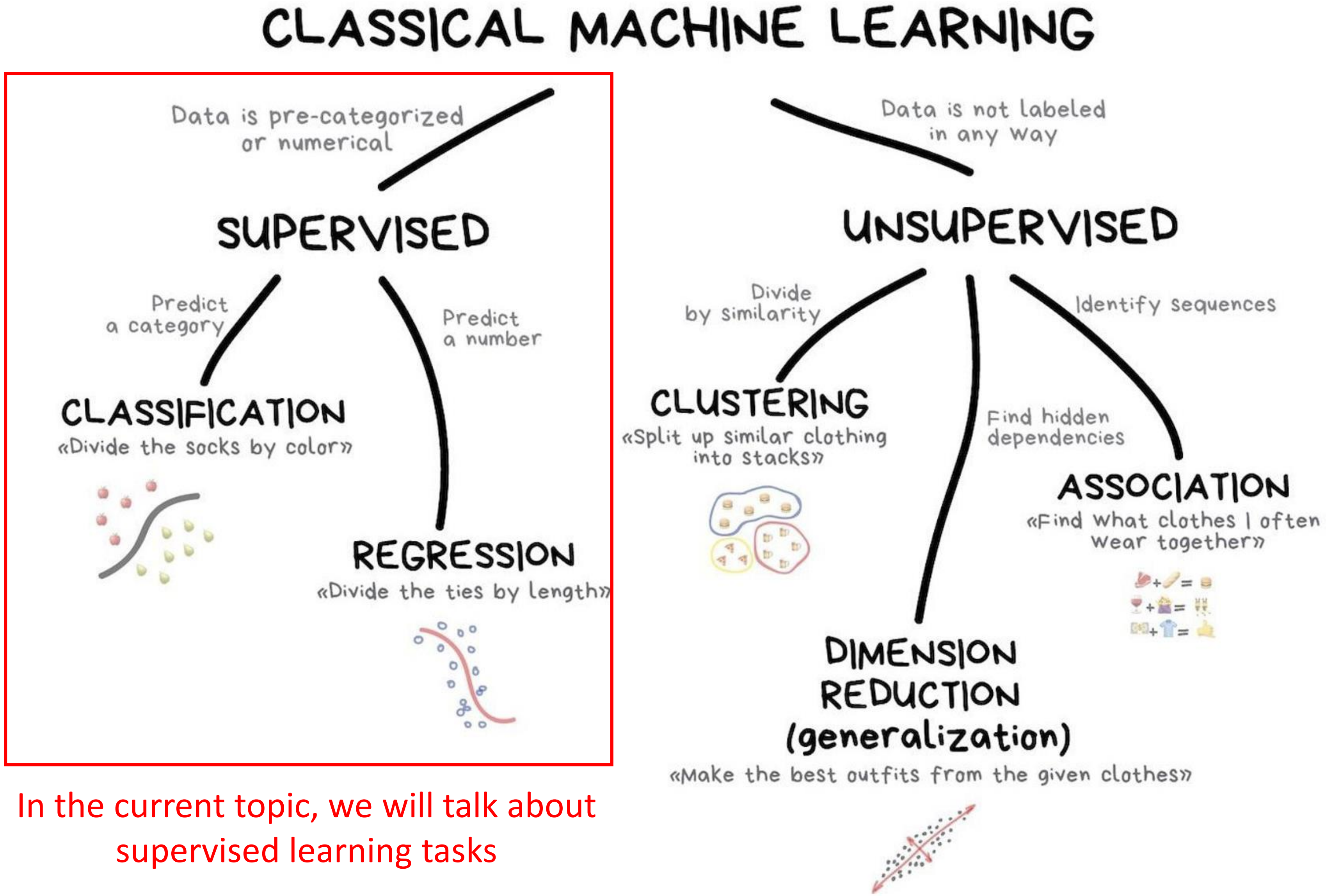
CLASSICAL MACHINE LEARNING
Data is pre-categorized or numerical
Data is not labeled in any way
SUPERVISED
UNSUPERVISED
Predict a category
Predict a number
CLASSIFICATION
«Divide the socks by color»
REGRESSION
«Divide the ties by length»
Divide by similarity
Identify sequences
Find hidden dependencies
CLUSTERING
«Split up similar clothing into stacks»
ASSOCIATION
«Find what clothes I often wear together»
DIMENSION REDUCTION (generalization)
«Make the best outfits from the given clothes»

Machine Learning

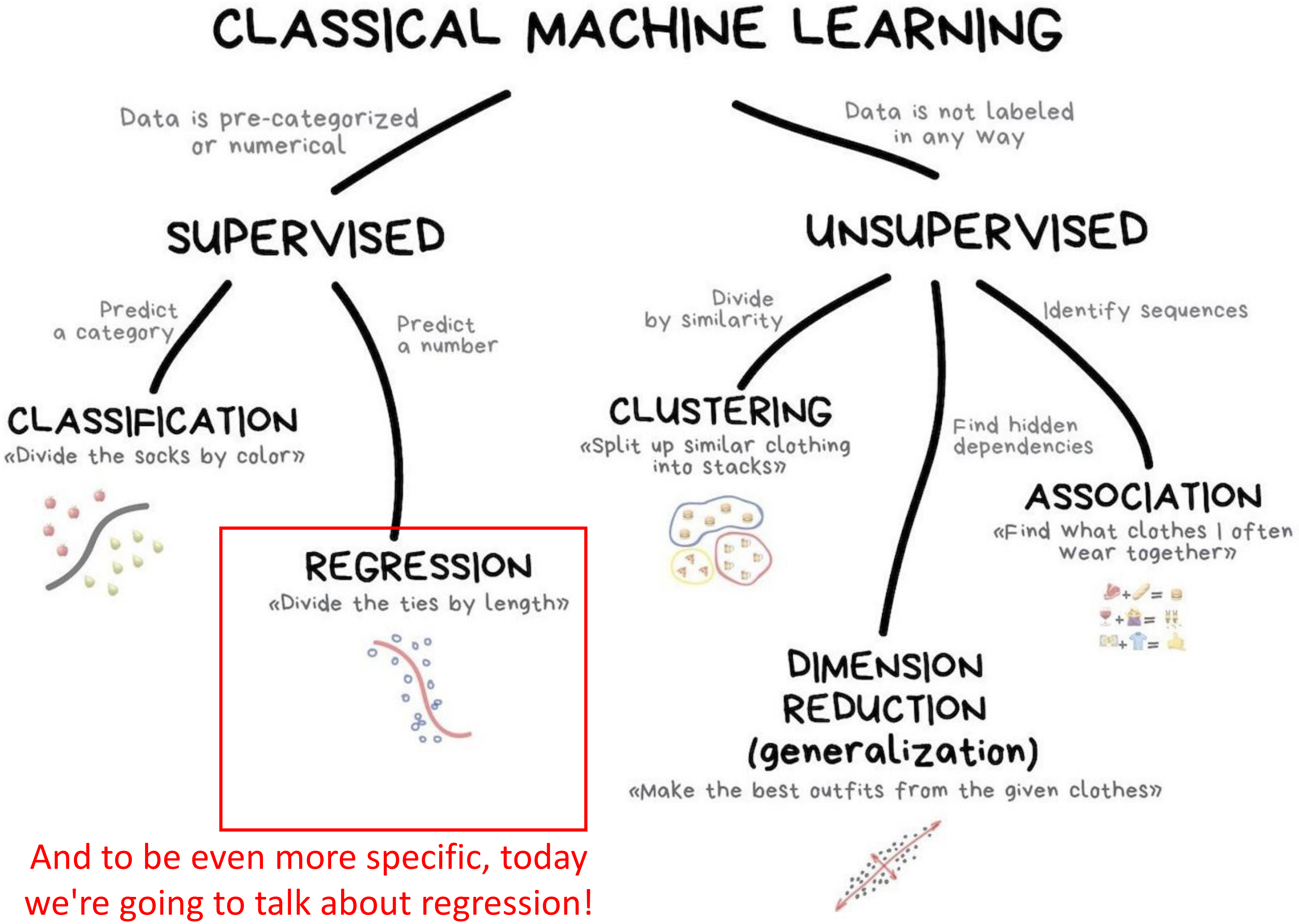
- These blocks of machine learning tasks are inextricably linked to the notion of labeled/unlabeled data

Labelled vs unlabelled data





In the current topic, we will talk about supervised learning tasks



And to be even more specific, today we're going to talk about regression!

Supervised learning

- Let's go over the basic notations again!

Supervised learning

- Let's go over the basic notations again!
- X — the set of all objects in the feature space
- Y — the range of values of the target variable

Supervised learning

- Let's go over the basic notations again!
- X — the set of all objects in the feature space
- Y — the range of values of the target variable
- What does machine learning represent in these terms?
- It's actually about finding an unknown dependency:
- $f: X \to Y$ — an unknown pattern, function
- It may even have a stochastic nature!

Supervised learning

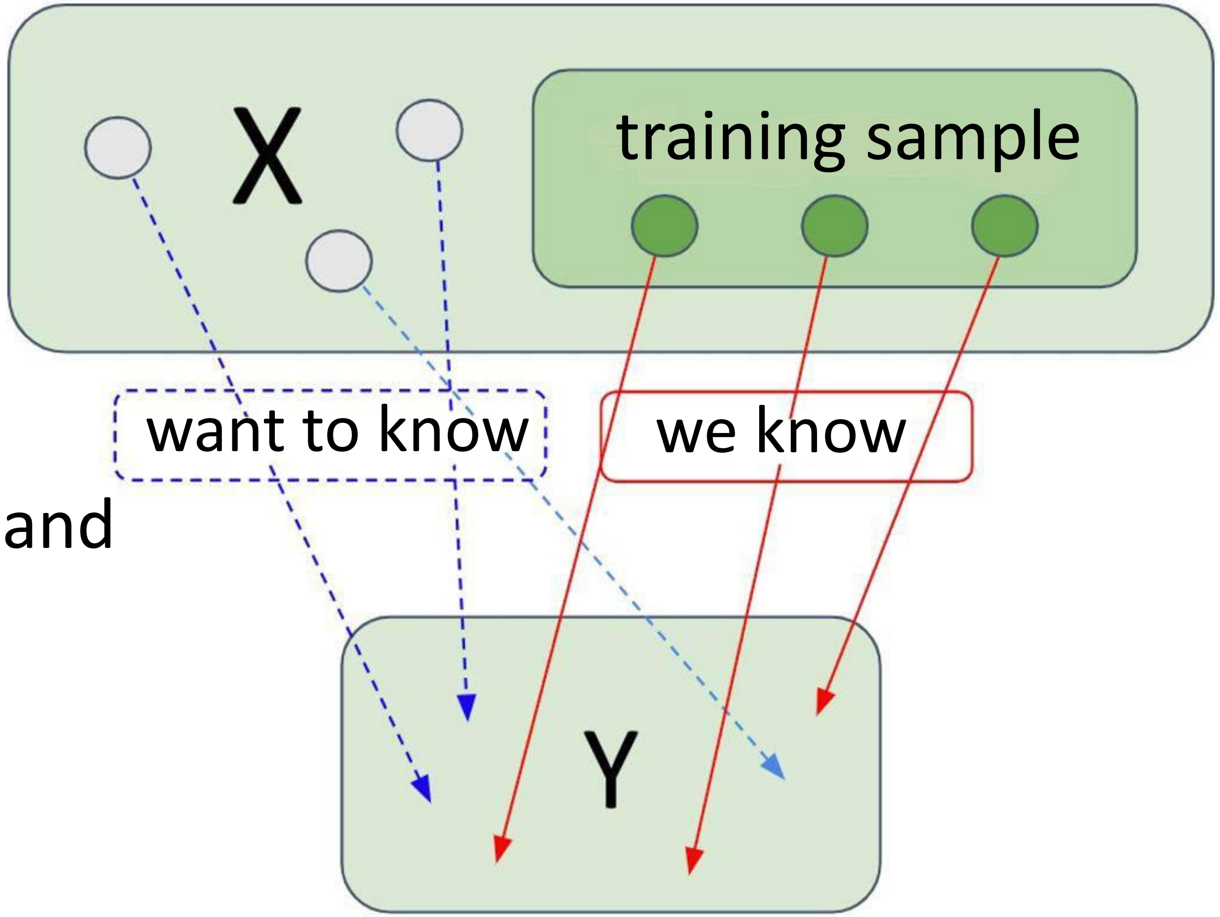
- How do we implement it?

- Given: Training sample of the form $\{(X_i, y_i)\}_{i=1}^{n}$
- Purpose: To maximize the accuracy and approximation of f.

Supervised learning

- How do we implement it?

- Given: Training sample of the form $\{(X_i, y_i)\}_{i=1}^{n}$
- Purpose: To maximize the accuracy and approximation of f.



Regression problem

- What is a regression problem?
- Simply put, it's a task where we want to predict some numerical (real) value.

Regression problem

- What is a regression problem?
- Simply put, it's a task where we want to predict some numerical (real) value.
- Examples of regression problem :
  - Predicting the cost of housing for a real estate company
  - Predicting delivery time
  - Predicting taxi demand in a specific area at a specific hour of the next day
  - And so on

Quality metrics and error function

Quality metrics and error function

- Suppose we have a basic understanding of setting up a machine learning task (and even managed to train a simple model – which is true, by the way: we tried it in the seminar!)
- But how do we know if our model is good or not?

Quality metrics and error function

- Suppose we have a basic understanding of setting up a machine learning task (and even managed to train a simple model – which is true, by the way: we tried it in the seminar!)
- But how do we know if our model is good or not?
- For this, there is a concept called a quality metric and an error function.

Quality metrics and error function

- Suppose we have a basic understanding of setting up a machine learning task (and even managed to train a simple model – which is true, by the way: we tried it in the seminar!)
- But how do we know if our model is good or not?
- For this, there is a concept called a quality metric and an error function.
- By the way, what is the difference between them? :)

Quality metrics and error function

- Quality metrics are used for the direct evaluation of the trained algorithm's performance, taking into account our business needs.
- Simply put: we look at the result produced by the model and compare it to the correct answers.

Quality metrics and error function

- Quality metrics are used for the direct evaluation of the trained algorithm's performance, taking into account our business needs.
- Simply put: we look at the result produced by the model and compare it to the correct answers.
- The error function, on the other hand, evaluates a certain mathematical function that the model tries to minimize during the training process.

Quality metrics and error function

- Quality metrics are used for the direct evaluation of the trained algorithm's performance, taking into account our business needs.
- Simply put: we look at the result produced by the model and compare it to the correct answers.
- The error function, on the other hand, evaluates a certain mathematical function that the model tries to minimize during the training process.

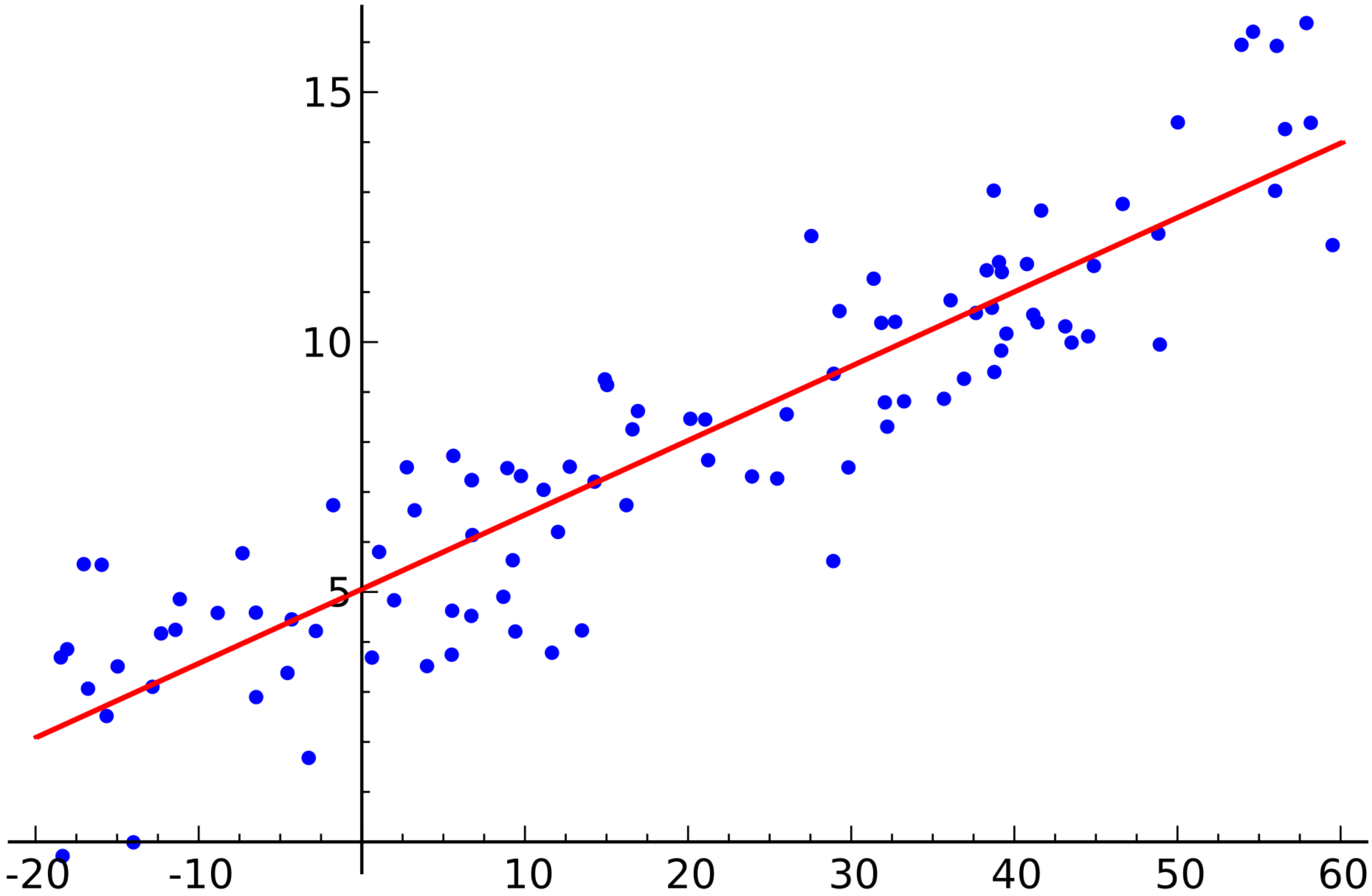
It would seem - what does gradient descent and all our talk about calculus have to do with it...?

Quality metrics and error function

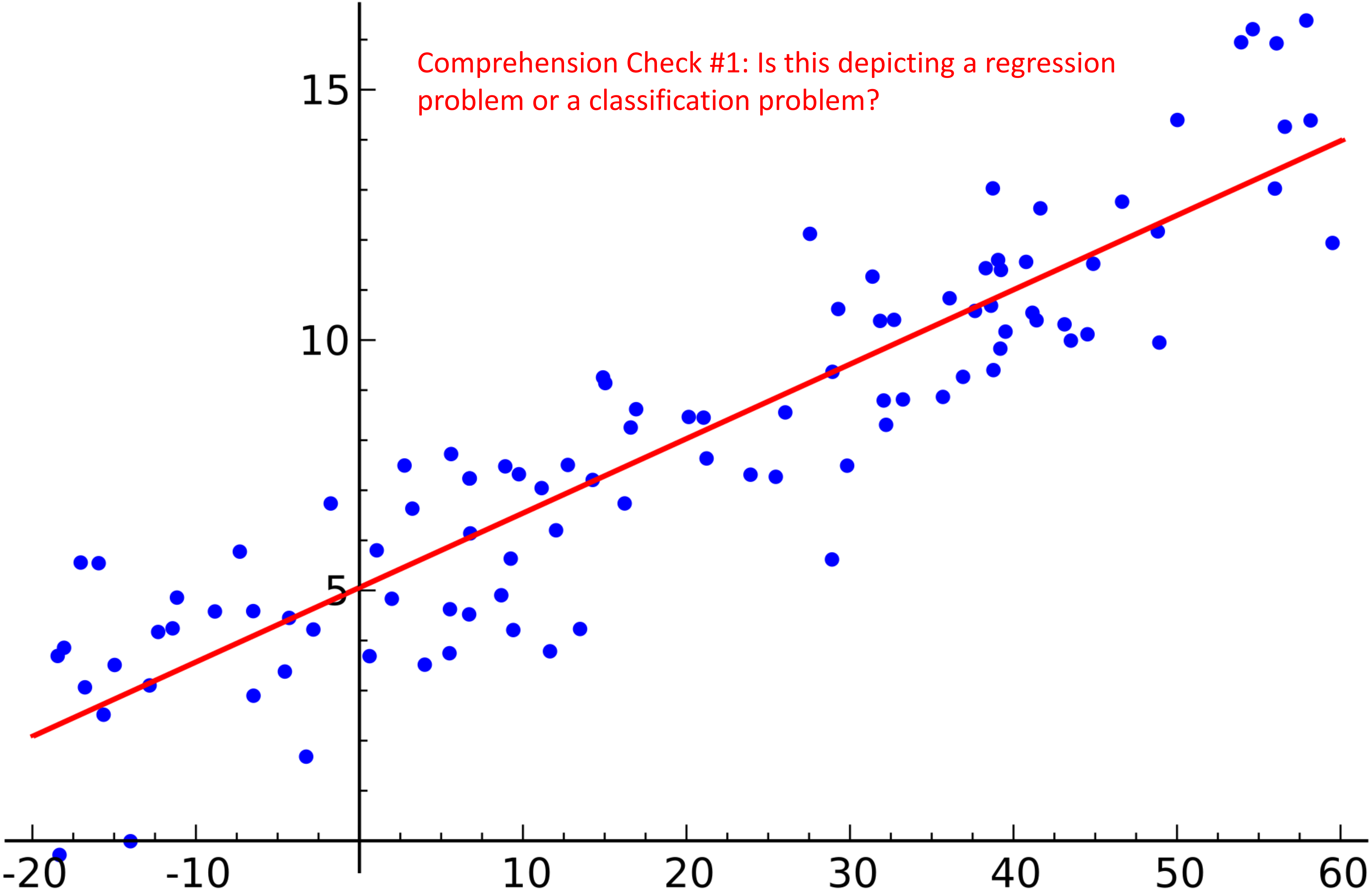
- However, the good news is that for regression tasks, these two concepts are very often the same! :)
- This, however, cannot be said about classification…

Quality metrics and error function

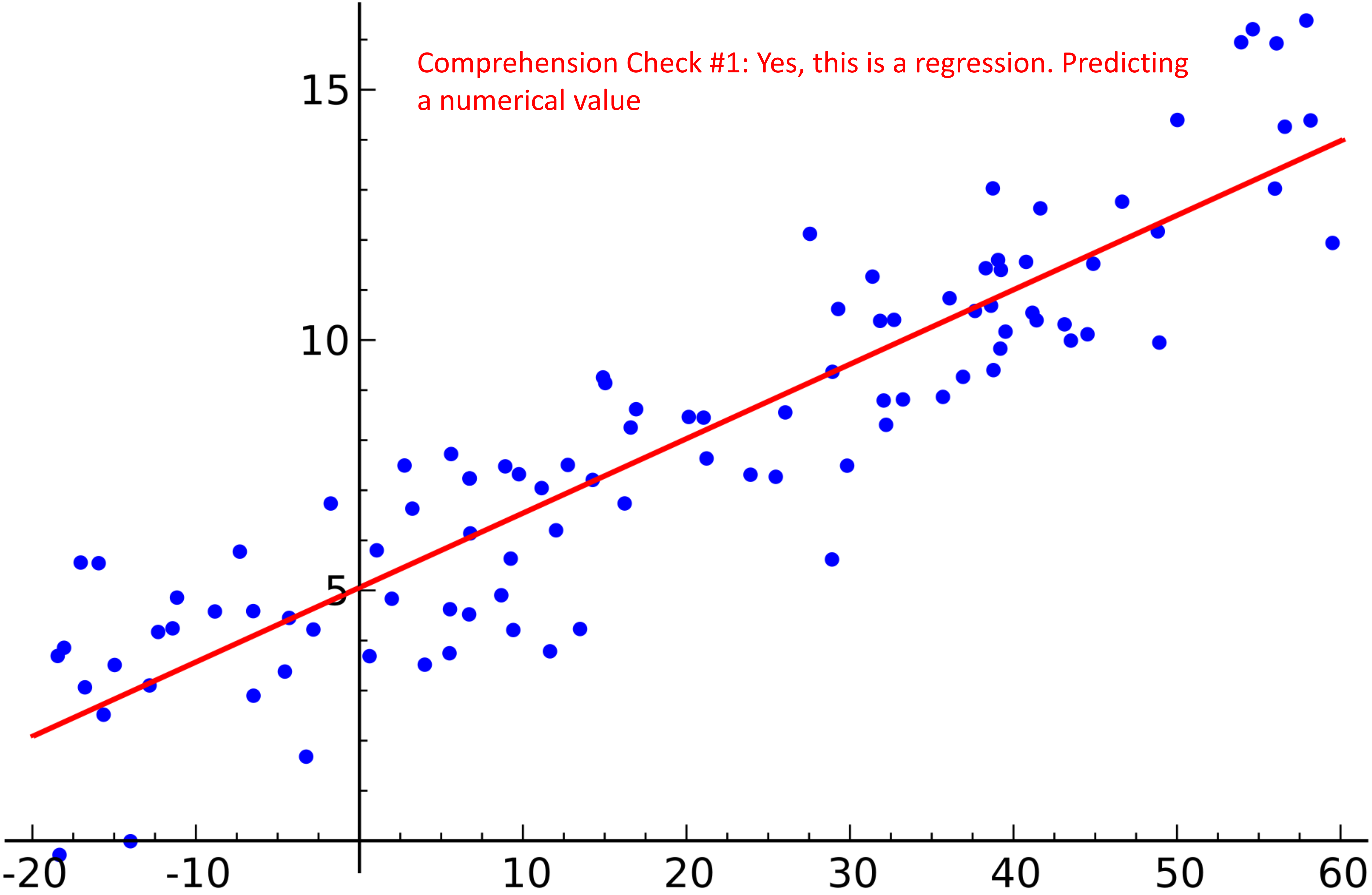
- However, the good news is that for regression tasks, these two concepts are very often the same! :)
- This, however, cannot be said about classification…
  - By the way, why do you think that is?

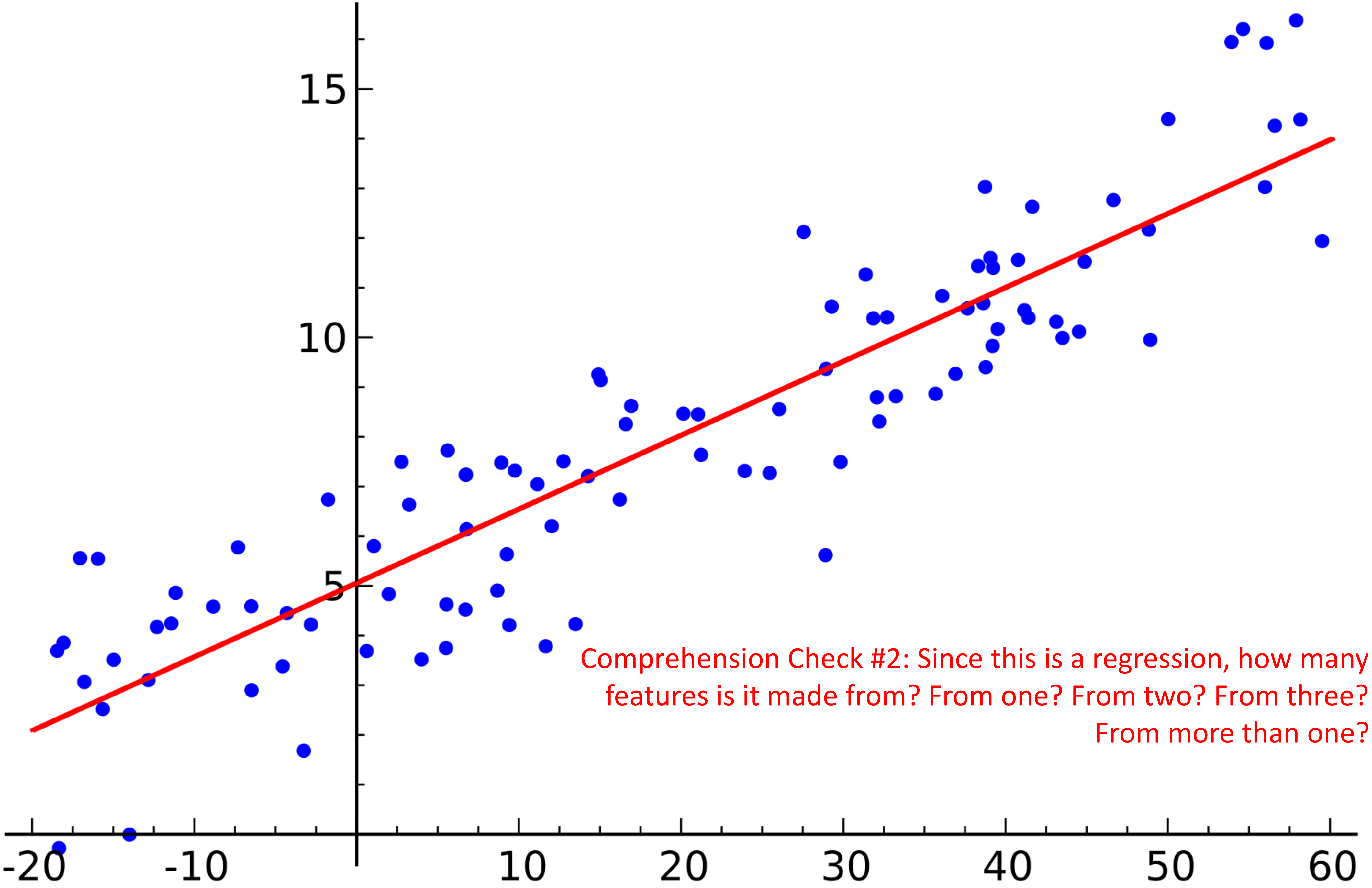


15
10
5
-20
-10
10
20
30
40
50
60

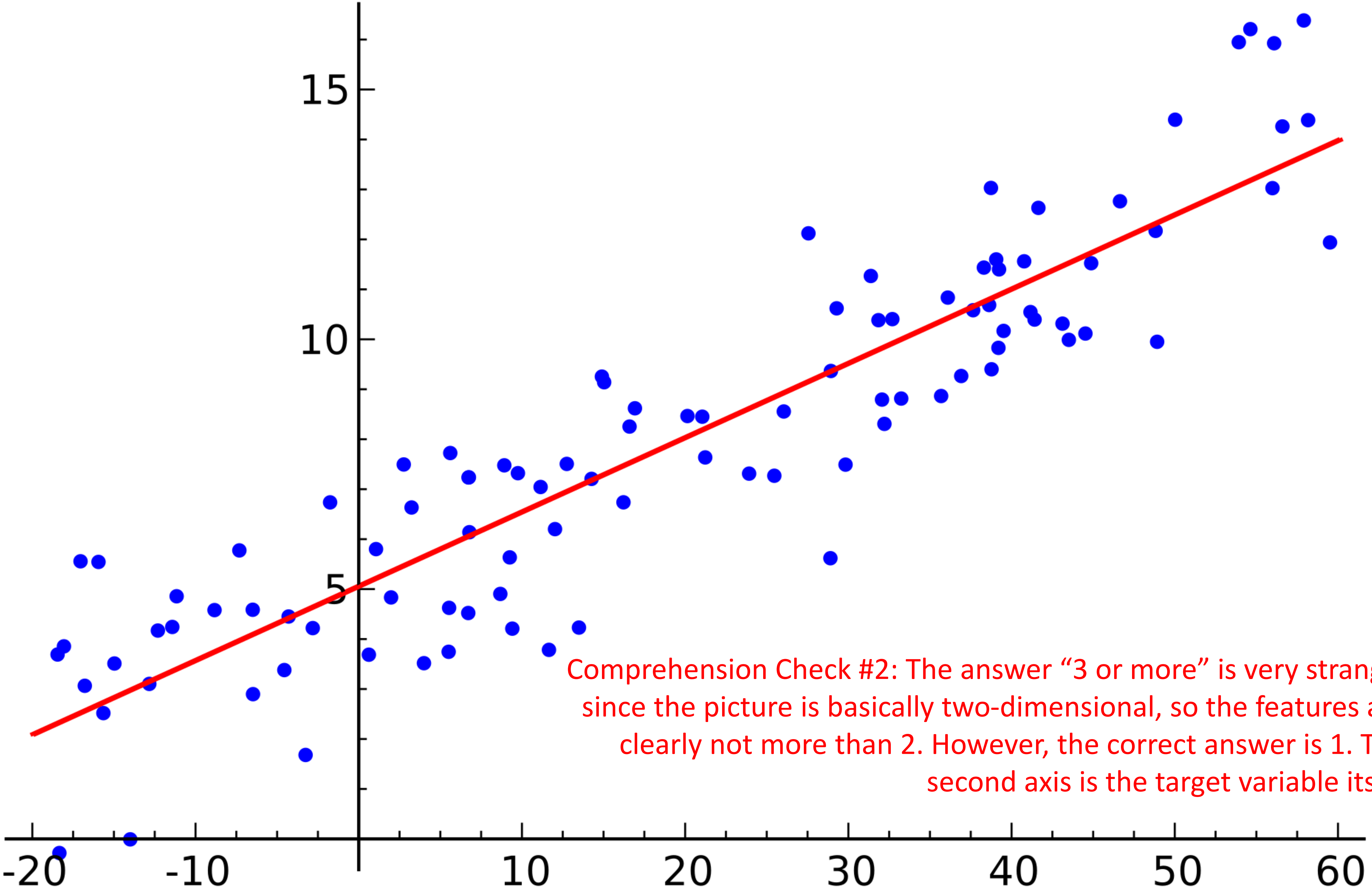

Comprehension Check #1: Is this depicting a regression problem or a classification problem?
15
10
5
-20
-10
10
20
30
40
50
60

Comprehension Check #1: Yes, this is a regression. Predicting a numerical value





Comprehension Check #2: Since this is a regression, how many features is it made from? From one? From two? From three? From more than one?



Comprehension Check #2: The answer “3 or more” is very strange, since the picture is basically two-dimensional, so the features are clearly not more than 2. However, the correct answer is 1. The second axis is the target variable itself

Quality metrics and error function

- What quality metrics for regression do you know?

Quality metrics and error function

- What quality metrics for regression do you know?

- Mean Squared Error (MSE):

$$\frac{1}{n}\sum_i^n(\widetilde{y}_i - y_i)^2$$

- Mean Absolute Error (MAE):

$$\frac{1}{n}\sum_i^n|\widetilde{y}_i - y_i|$$

- Max Error:

$$\max_i|\widetilde{y}_i - y_i|$$

Quality metrics and error function

- What quality metrics for regression do you know?

- Mean Squared Error (MSE):

$$\frac{1}{n}\sum_i^n(\tilde{y}_i - y_i)^2$$

- Mean Absolute Error (MAE):

$$\frac{1}{n}\sum_i^n|\tilde{y}_i - y_i|$$

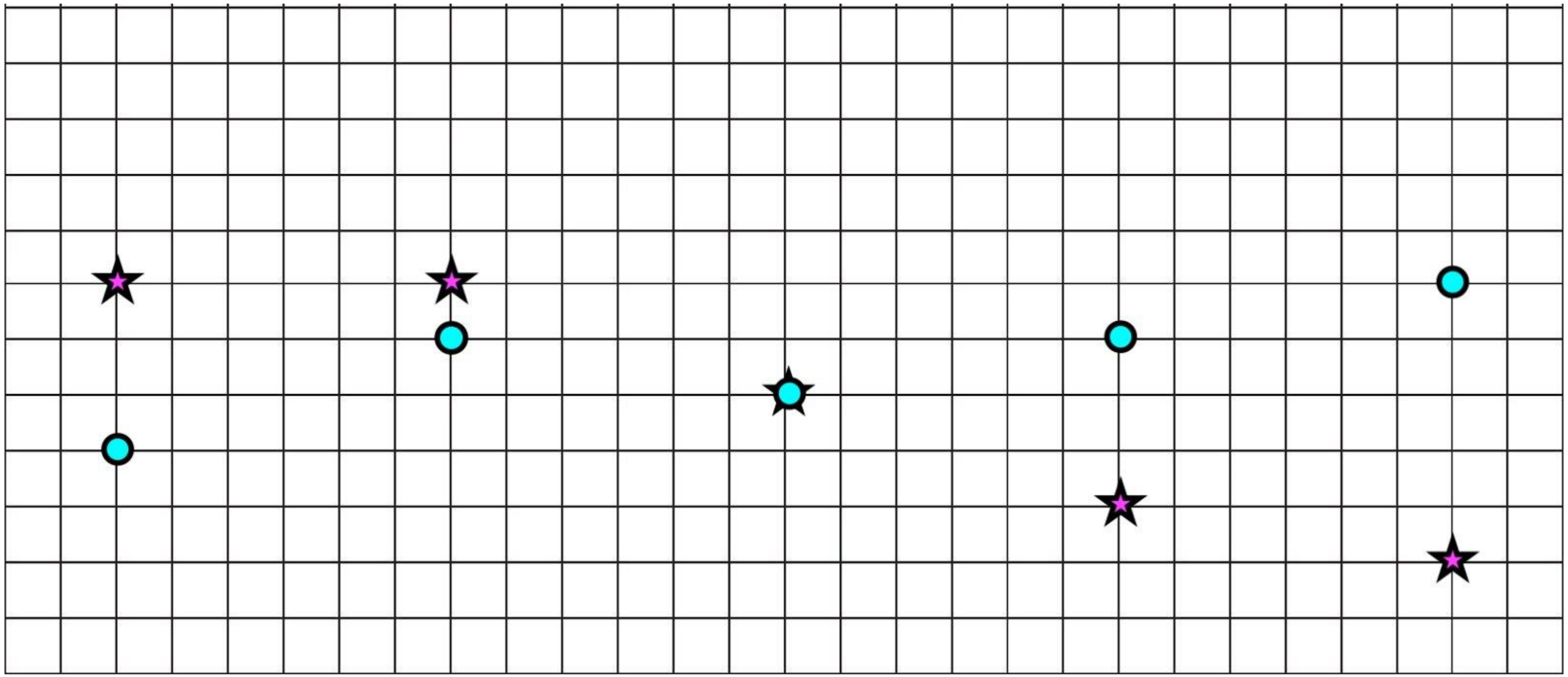
- Max Error:

$$\max_i|\tilde{y}_i - y_i|$$

Comment on the notation: what is $\widetilde{y_i}$, y_i, what does n indicate

Quality metrics and error function

- Let us draw these metrics and practice the simplest case!



Quality metrics and error function

- However, all the previous metrics have an important drawback!
- What is it?

Quality metrics and error function

- However, all the previous metrics have an important drawback!
- What is it?
- They do not allow the assessment of quality in absolute terms, as they depend on the units of measurement.
- In other words, you can only compare two different models to each other in terms of quality, but you cannot say whether they are good models overall or not.

Quality metrics and error function

- However, all the previous metrics have an important drawback!
- What is it?
- They do not allow the assessment of quality in absolute terms, as they depend on the units of measurement.
- In other words, you can only compare two different models to each other in terms of quality, but you cannot say whether they are good models overall or not.
- Which metrics solve this problem?

Quality metrics and error function

- Coefficient of determination (R^2):

$$1 - \frac{\sum_i^n (\widetilde{y_i} - y_i)^2}{\sum_i^n (\overline{y_i} - y_i)^2} \in (-\infty, 1]$$

- Mean Absolute Percentage Error (MAPE):

$$100 \cdot \frac{1}{n} \sum_i^n \left| \frac{y_i - \widetilde{y_i}}{y_i} \right| \in [0, +\infty)$$

Quality metrics and error function

- Coefficient of determination (R^2):

$$1 - \frac{\sum_i^n (\widetilde{y_i} - y_i)^2}{\sum_i^n (\overline{y_i} - y_i)^2} \in (-\infty, 1]$$

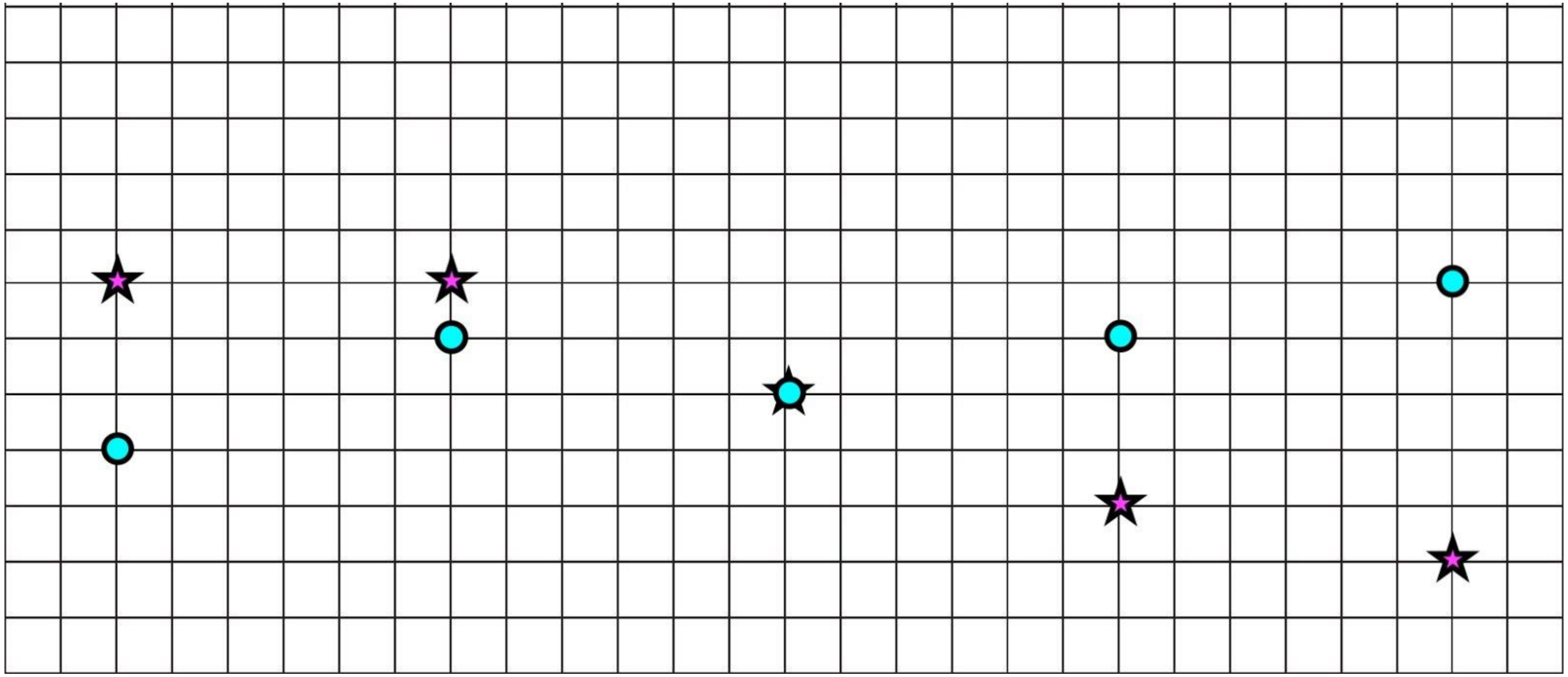
- Mean Absolute Percentage Error (MAPE):

$$100 \cdot \frac{1}{n} \sum_i^n \left| \frac{y_i - \widetilde{y_i}}{y_i} \right| \in [0, +\infty)$$

What values of these metrics will the best model have? How are these metrics better than the previous ones?

Quality metrics and error function

- Let's practice with them too!



KNN Algorithm

KNN Algorithm

- Let's now return to the algorithm we once considered – the K-nearest neighbors algorithm.
- We now know all the necessary mathematics to discuss it fully!

KNN Algorithm

- Let's now return to the algorithm we once considered – the K-nearest neighbors algorithm.
- We now know all the necessary mathematics to discuss it fully!
- Let's recall the main idea.

KNN Algorithm

- The input is a vector representing the feature description of some object.

KNN Algorithm

- The input is a vector representing the feature description of some object.
- K nearest vectors are found, for which the response is known.

KNN Algorithm

- The input is a vector representing the feature description of some object.
- K nearest vectors are found, for which the response is known.

This is where we had a major snag earlier! We'll get back to it very soon!

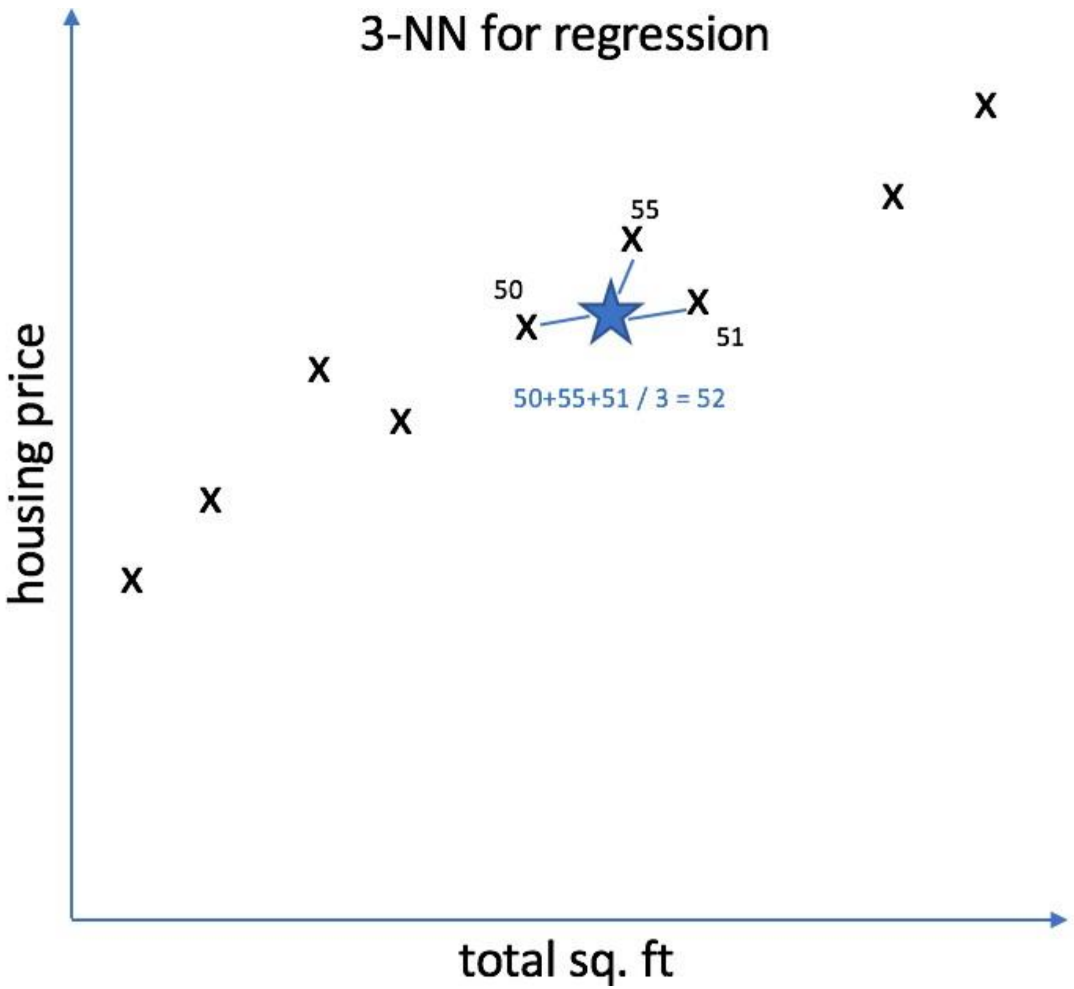
KNN Algorithm

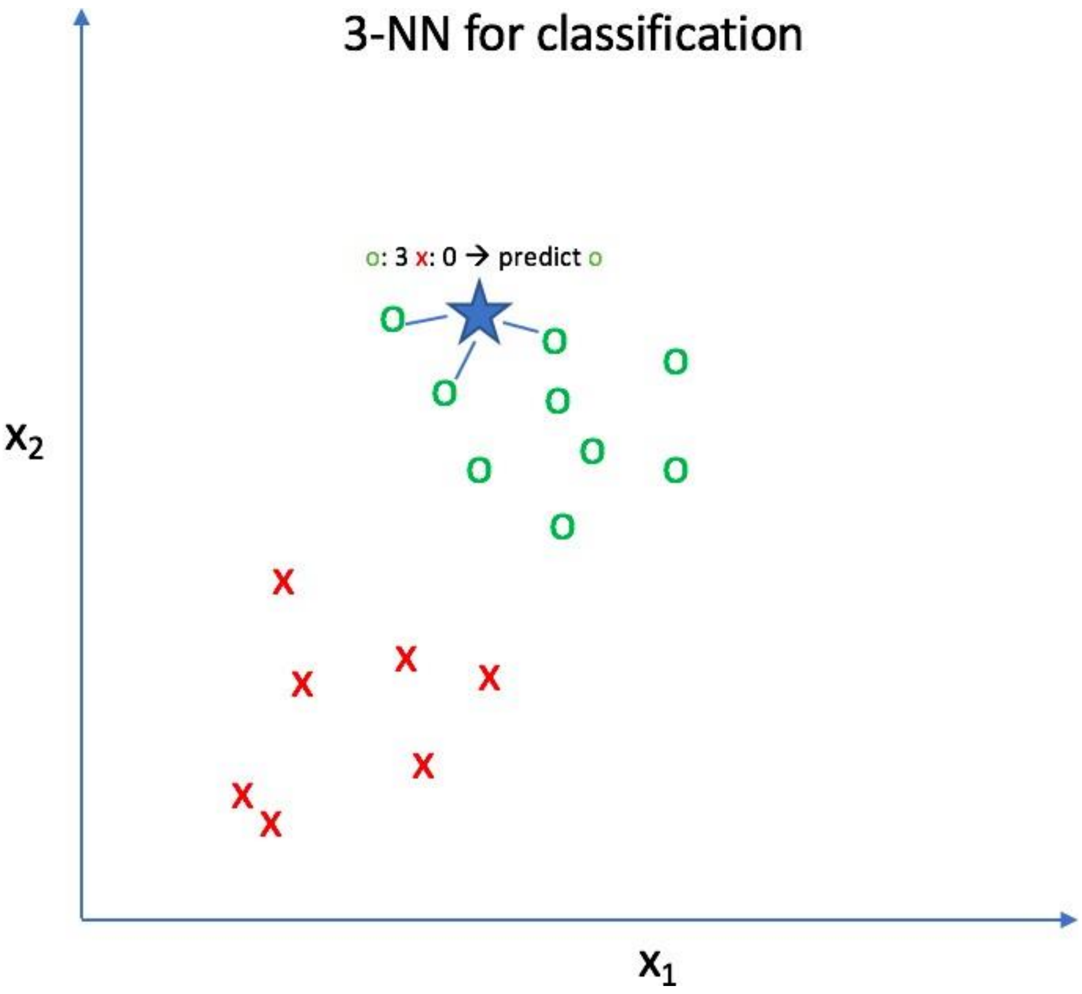
- The input is a vector representing the feature description of some object.
- K nearest vectors are found, for which the response is known.
- The response for the new object is selected using:
  - Averaging, in the case of regression
  - Voting, in the case of classification

KNN Algorithm

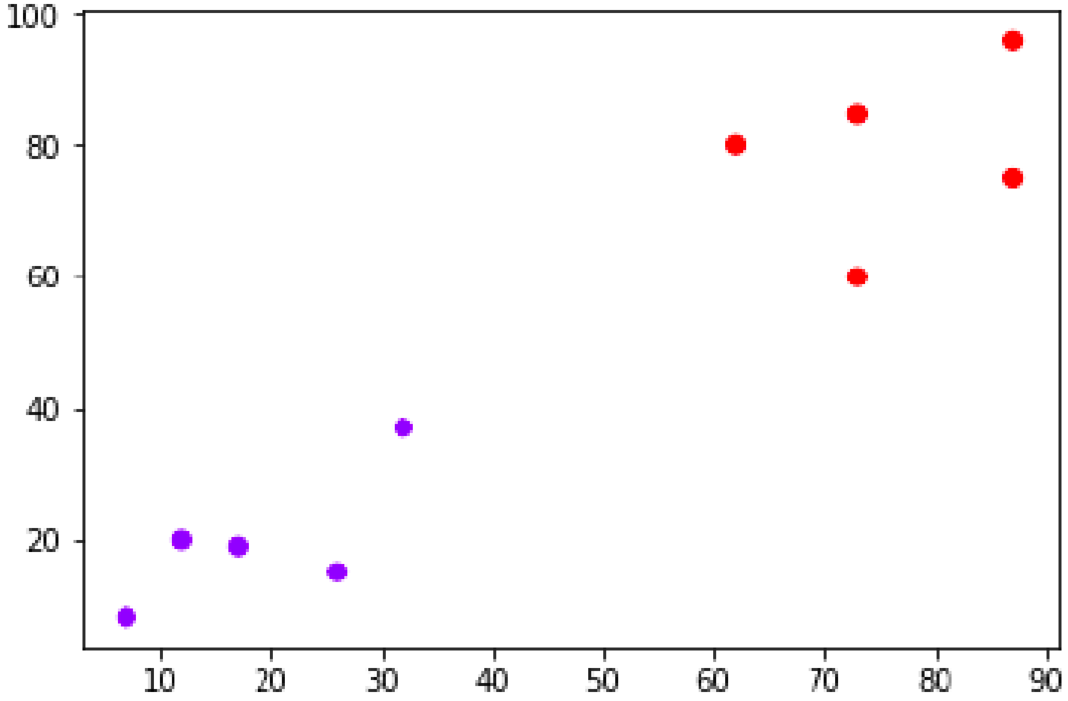
- The input is a vector representing the feature description of some object.
- K nearest vectors are found, for which the response is known.
- The response for the new object is selected using:
  - Averaging, in the case of regression
  - Voting, in the case of classification
- It's also possible to use weighted averaging/voting and many other modifications of the standard algorithm.

KNN Algorithm

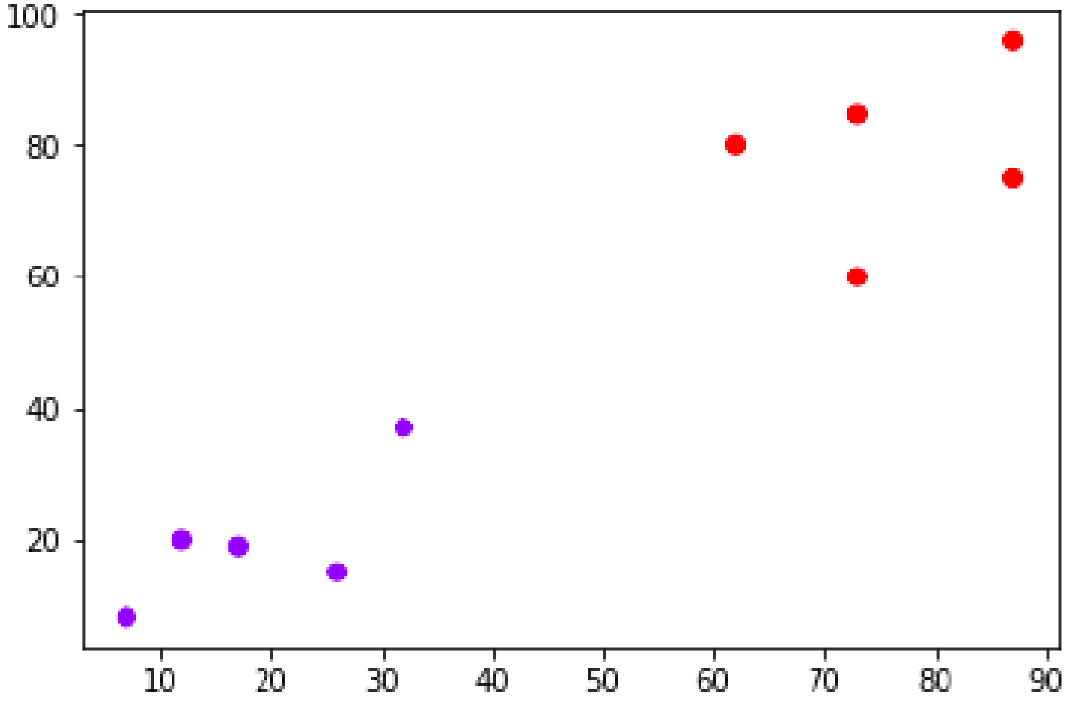


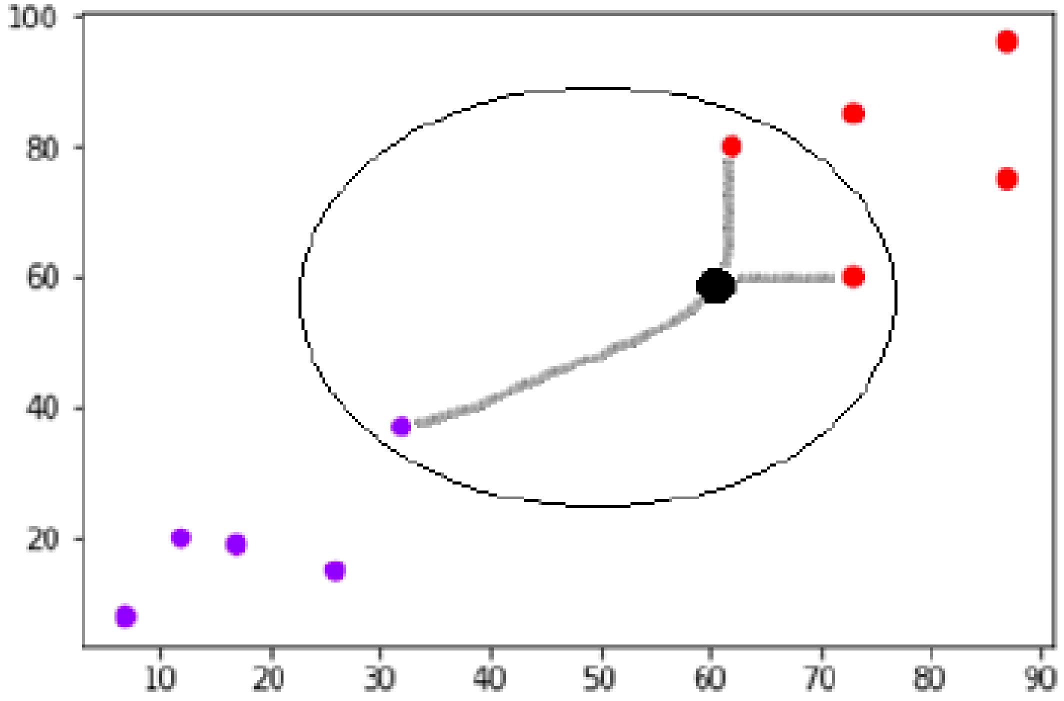


KNN Algorithm

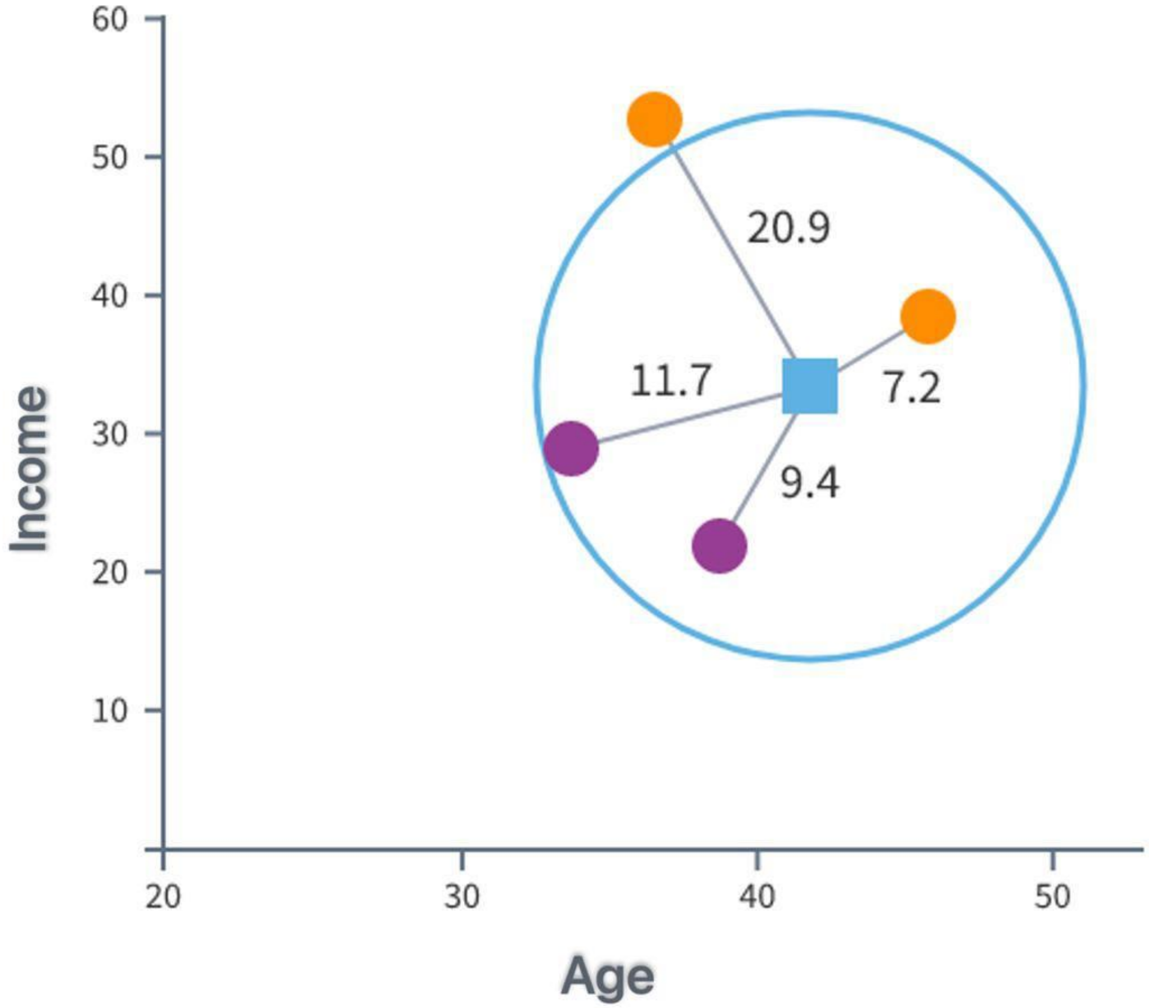


KNN Algorithm

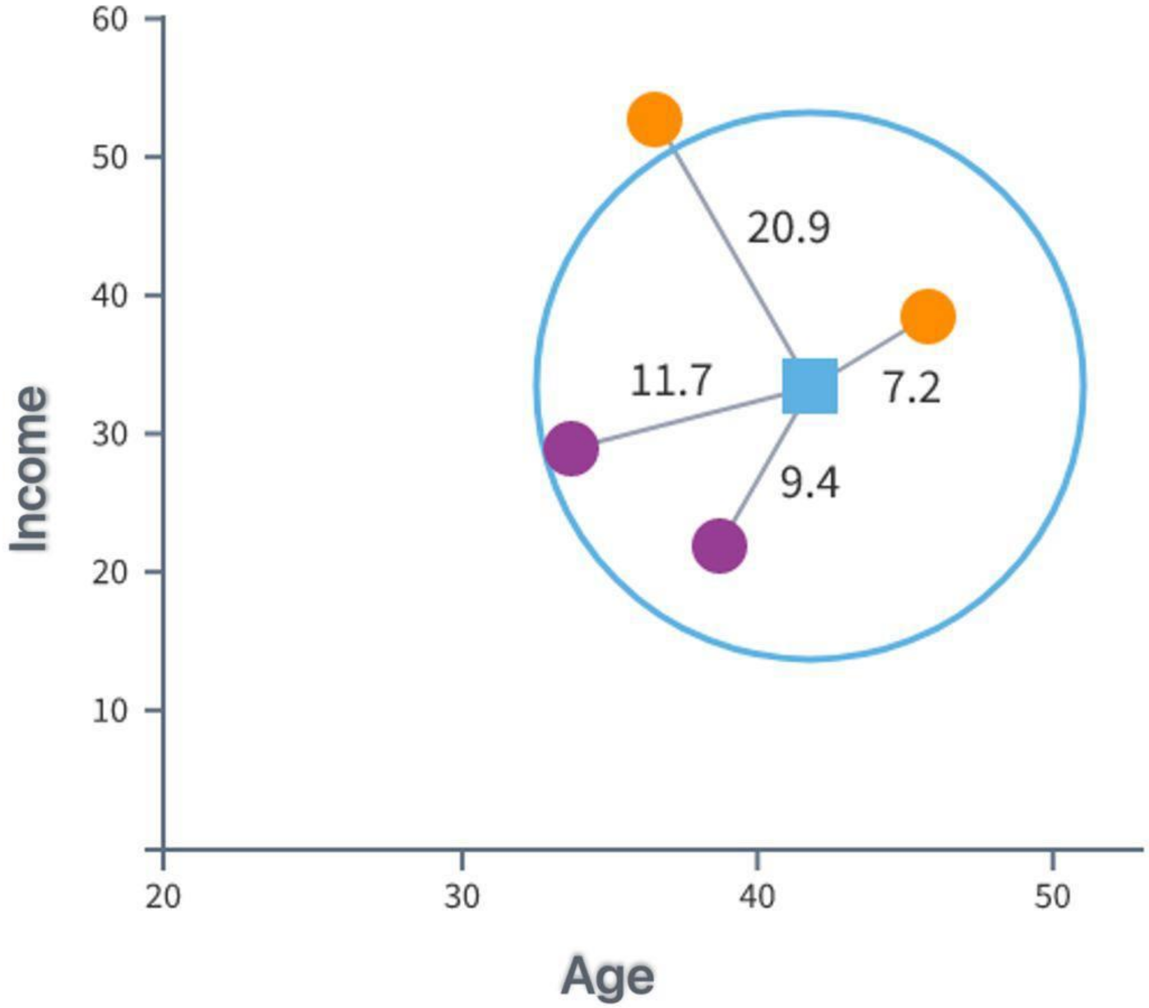




KNN Algorithm

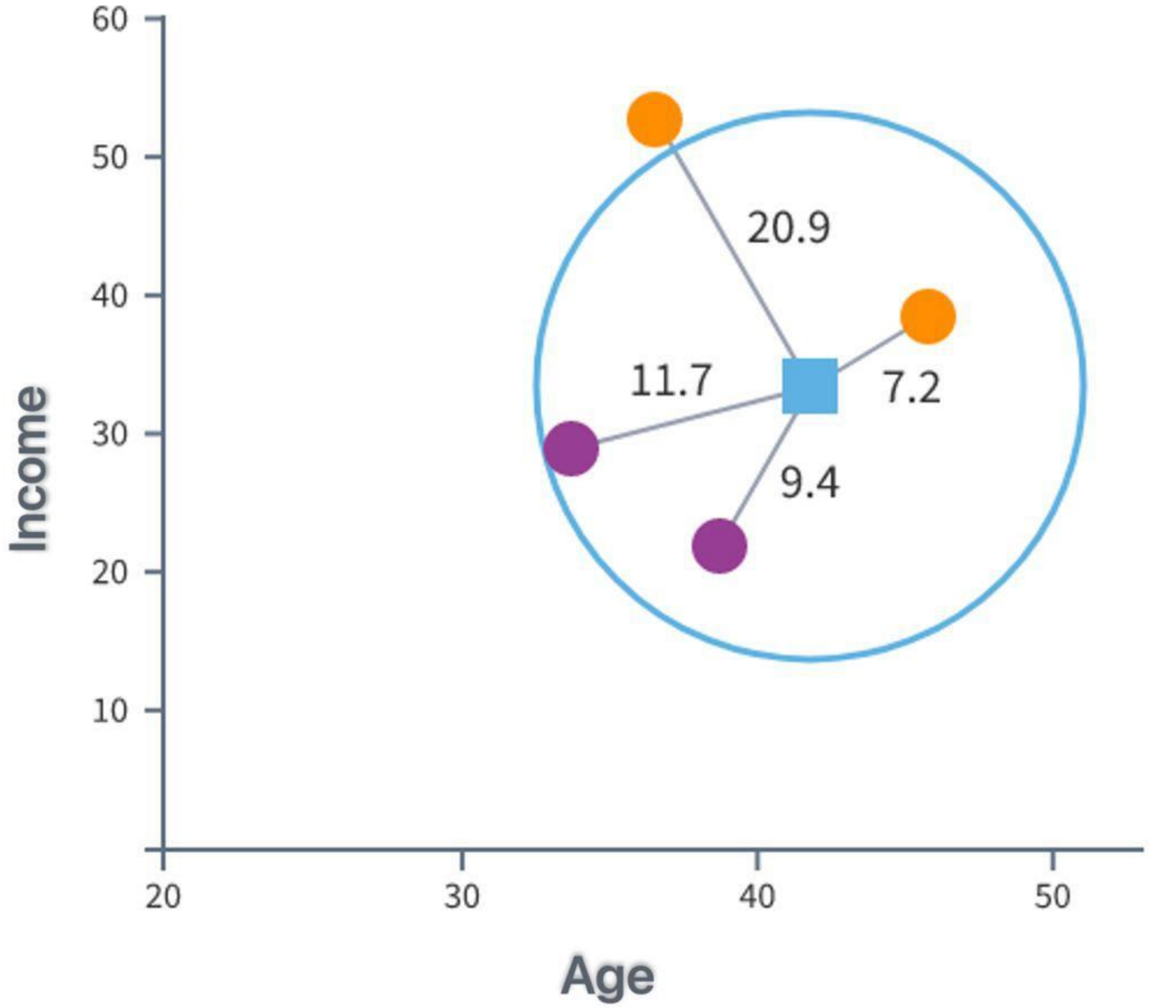


KNN Algorithm



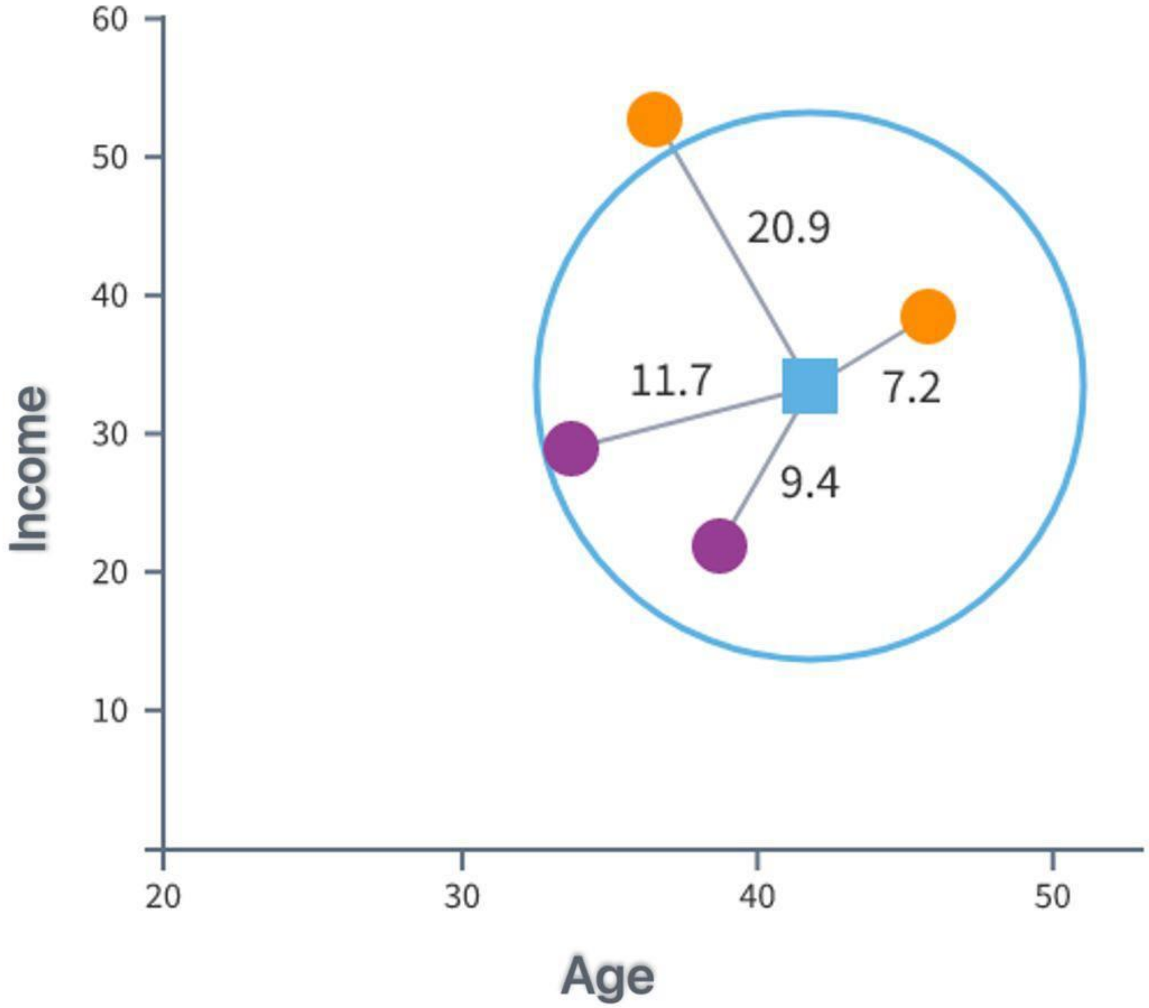
Comprehension Check #1: Is this depicting a regression problem or a classification problem?

KNN Algorithm



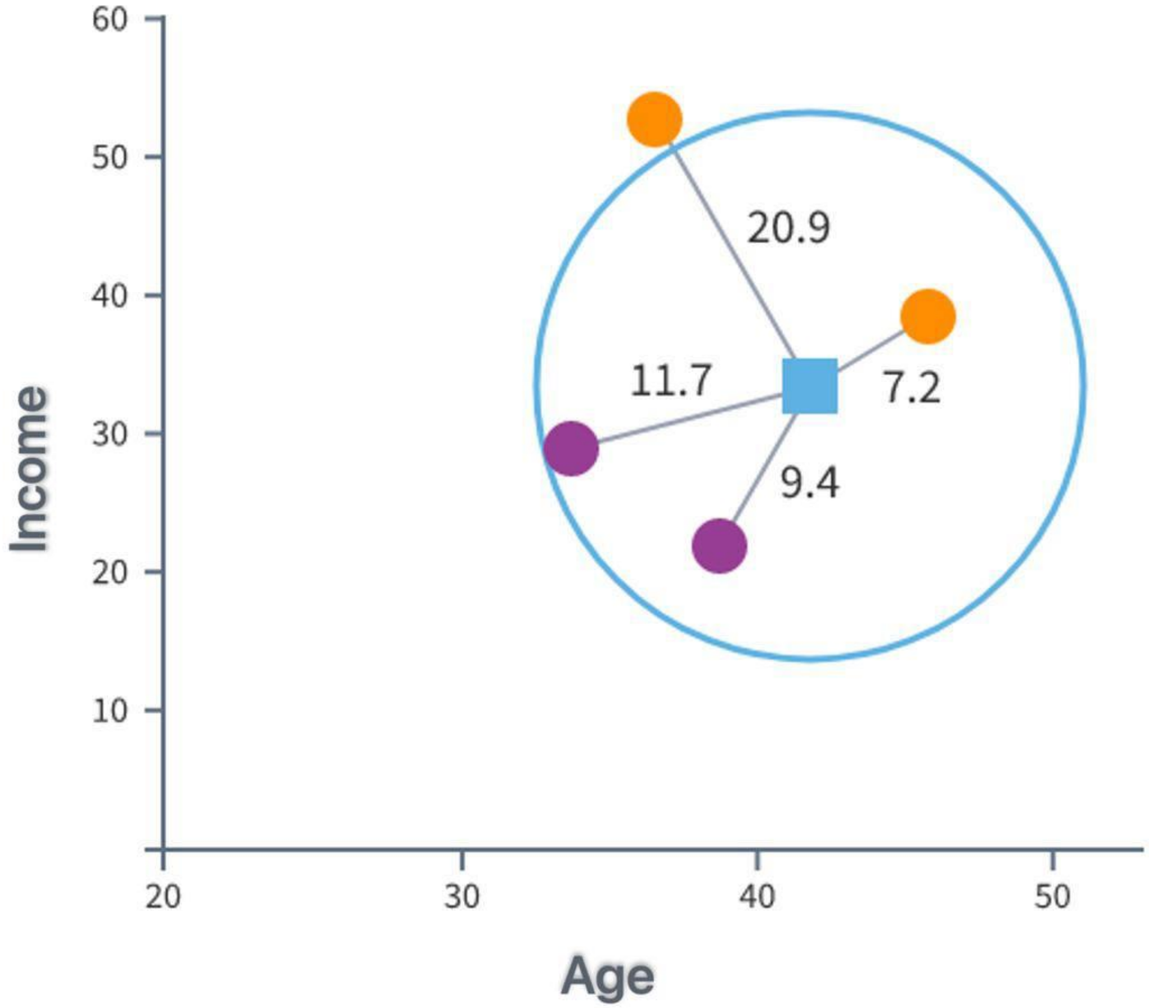
Comprehension check #1: Both are problems!

KNN Algorithm



Comprehension Check #2: Since this is a regression, how many traits is it made from? From one? From two? From three? From more than one?

KNN Algorithm



Comprehension check #2: Here it is already from two, not one!
The target variable (its value) is indicated next to the points simply as a label (color or number)

Metric spaces

- Now, as we intended, let's return to the main question: how do we find the distance between "similar" points (vectors)?

Metric spaces

- Now, as we intended, let's return to the main question: how do we find the distance between "similar" points (vectors)?
- For this, we need to introduce an important concept – a metric space!

Metric spaces

- Now, as we intended, let's return to the main question: how do we find the distance between "similar" points (vectors)?
- For this, we need to introduce an important concept – a metric space!
- A metric space is a space in which a certain metric (function) is defined, allowing us to calculate the distance between any two points in this space. This metric is called a distance metric (or distance function).

Metric spaces

- Give examples of some metric spaces

Metric spaces

- Give examples of some metric spaces
- The simplest and most familiar example is a map. For instance, in Yandex.Maps.

Metric spaces

- Give examples of some metric spaces
- The simplest and most familiar example is a map. For instance, in Yandex.Maps.
- Questions to ponder:
  - How do you calculate the distance between the starting point and the endpoint of a route on maps?
  - How do you do this on a piece of paper between two points?
  - How do you do this in our three-dimensional space?


Middle Earth
the Sundering Seas
Forodwaith
Northern Waste
Forochel
Arnor
Angmar
Grey Mountains
Withered Heath
Ettenmoors
Misty Mountains
Forlindon
Blue Mountains
Harlindon
the Shire
Eriador
Minhiriath
Enedwaith
Dunland
Fangorn Forest
Rhovanion
Mirkwood
Erebor
Iron Hills
Rhûn
The Brown Lands
Dagorlad
Ash Mountains
Rohan
White Mountains
Anórien
Gondor
Ithilien
Mount Doom
Barad-dûr
Gorgoroth
Mordor
Mountains of Shadow
Mountains of the East
Bay of Belfalas
Hârondor
Khând
Near Hârad
Hârâdwâith
Umbar
N
W
E
S
pikabu.ru

Distance metrics

- The standard, familiar distance metric, proven through the Pythagorean theorem and extended to the multi-dimensional case
- Is called the Euclidean metric.

Distance metrics

- The standard, familiar distance metric, proven through the Pythagorean theorem and extended to the multi-dimensional case
- Is called the Euclidean metric.

$$\rho(x, y) = \sqrt{\sum_i (x_i - y_i)^2}$$

Distance metrics

- The standard, familiar distance metric, proven through the Pythagorean theorem and extended to the multi-dimensional case
- Is called the Euclidean metric.

$$\rho(x, y) = \sqrt{\sum_i (x_i - y_i)^2}$$

Comprehension question: what does i represent in this formula?

Distance metrics

- A metric that often proves more effective than the Euclidean due to its structure, which aligns closely with many domains (such as maps)
- Manhattan metric. :)

Distance metrics

- A metric that often proves more effective than the Euclidean due to its structure, which aligns closely with many domains (such as maps)
- Manhattan metric. :)

$$\rho(x, y) = \sum_{i} |x_i - y_i|$$

Distance metrics

- A metric that often proves more effective than the Euclidean due to its structure, which aligns closely with many domains (such as maps)
- Manhattan metric. :)

$$\rho(x, y) = \sum_i |x_i - y_i|$$

What is the advantage of such a metric and what does Manhattan have to do with it? :)

Distance metrics

- The general case is the Minkowski metric

Distance metrics

- The general case is the Minkowski metric

$$\rho(x, y) = \sqrt[p]{\sum_i |x_i - y_i|^p}$$

KNN Algorithm

- Let's return to the setup of the KNN problem.
- Suppose we have data recorded in a familiar format, as a feature matrix of size $M \times N$, where M is the number of objects, and N is the number of features.

KNN Algorithm

- Let's return to the setup of the KNN problem.
- Suppose we have data recorded in a familiar format, as a feature matrix of size $M \times N$, where M is the number of objects, and N is the number of features.
- We assume that all N features are in the same metric space. If this is the case, each of the M objects is simply a point in an N-dimensional metric space, which means that we can calculate the distance between any two points.

KNN Algorithm

- Let's assume that all N features are in the same metric space. If so, each of the M objects is simply a point in an N-dimensional metric space, meaning we can calculate the distance between any two points.
- Then, if we select any point, we can determine which of the $M-1$ remaining points will be the closest to the point under consideration—such a point we call the nearest neighbor.
- Similarly, we can identify the second closest neighbor, and so on.

KNN Algorithm

- Now, since we've established that our space is metric, we can assert that points which are close to each other in distance will be similar, while those that are farther apart will be dissimilar.

KNN Algorithm

- Now, since we've established that our space is metric, we can assert that points which are close to each other in distance will be similar, while those that are farther apart will be dissimilar.
- This means the KNN algorithm is justified and applicable!
- Hooray!

KNN Algorithm

- An important remark (which we've indirectly touched on before)
- With the KNN algorithm, there isn't "training" in the traditional sense. Essentially, all you need to do is "remember" the locations of the points in space, as well as their targets.

KNN Algorithm

- An important remark (which we've indirectly touched on before)
- With the KNN algorithm, there isn't "training" in the traditional sense. Essentially, all you need to do is "remember" the locations of the points in space, as well as their targets.
- Later, predictions are made simply by averaging the target of the nearest neighbors stored in memory from the training set.

KNN Algorithm

- An important remark (which we've indirectly touched on before)
- With the KNN algorithm, there isn't "training" in the traditional sense. Essentially, all you need to do is "remember" the locations of the points in space, as well as their targets.
- Later, predictions are made simply by averaging the target of the nearest neighbors stored in memory from the training set.
- This characteristic makes the KNN algorithm non-parametric!

KNN Algorithm

- An important remark (which we've indirectly touched on before)
- With the KNN algorithm, there isn't "training" in the traditional sense. Essentially, all you need to do is "remember" the locations of the points in space, as well as their targets.
- Later, predictions are made simply by averaging the target of the nearest neighbors stored in memory from the training set.
- This characteristic makes the KNN algorithm non-parametric!
  - However, the algorithm does indeed have hyperparameters, the most important of which is K — the number of neighbors to include in the calculation.

KNN Algorithm

A logically arising and yet important clarification: what is the difference between parameters and hyperparameters of an algorithm in general?

- An important remark (which we've indirectly touched on before)
- With the KNN algorithm, there isn't "training" in the traditional sense. Essentially, all you need to do is "remember" the locations of the points in space, as well as their targets.
- Later, predictions are made simply by averaging the target of the nearest neighbors stored in memory from the training set.
- This characteristic makes the KNN algorithm non-parametric!
  - However, the algorithm does indeed have hyperparameters, the most important of which is K — the number of neighbors to include in the calculation.