CS 5806 Machine Learning II

Lecture 4 - Learning Theory 1: Bias Variance Tradeoff & PAC Learning

August 30th, 2023

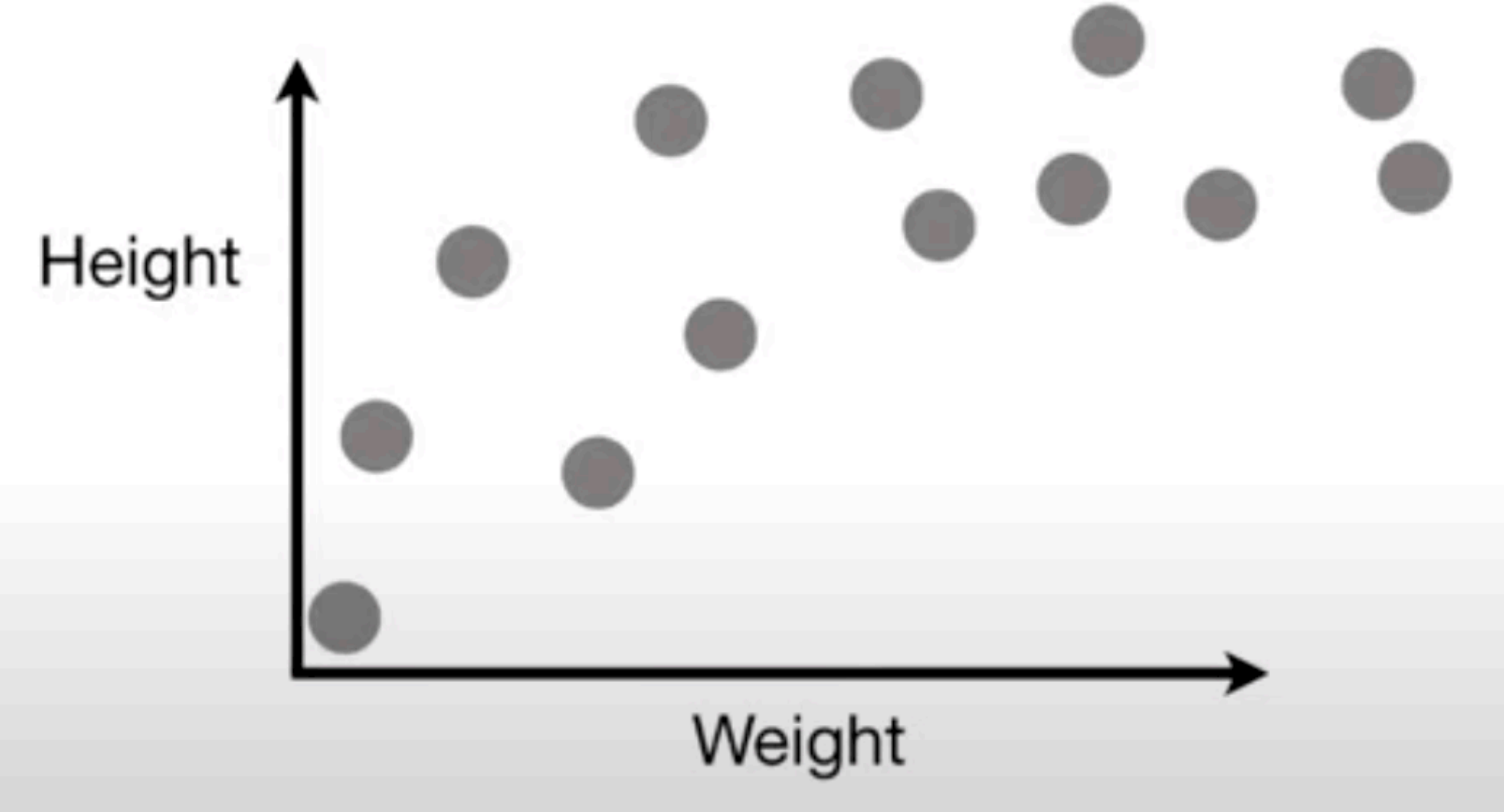
Hoda Eldardiry

Learning Objectives

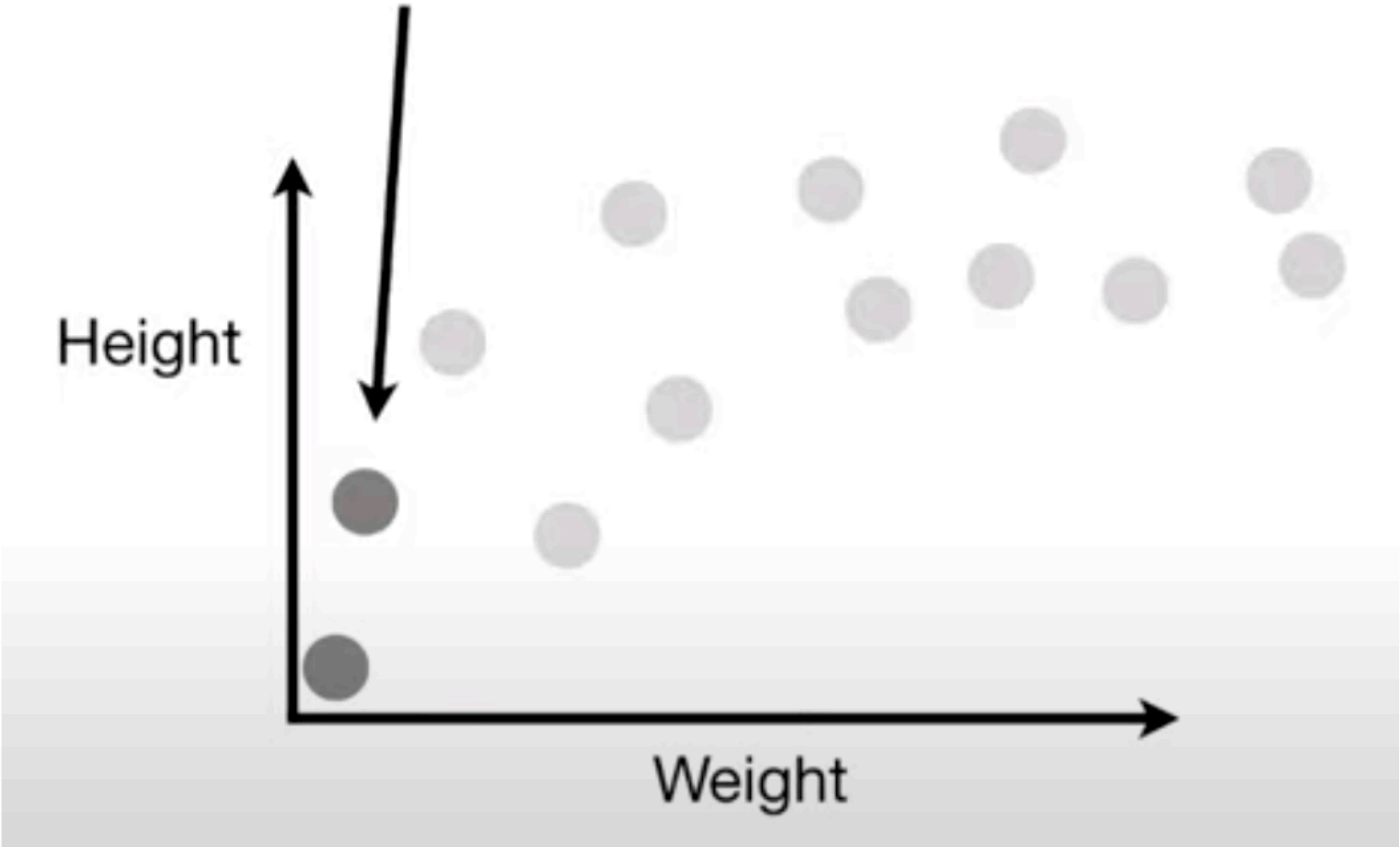
- Bias Variance Tradeoff
- PAC Learning

Bias Variance Tradeoff

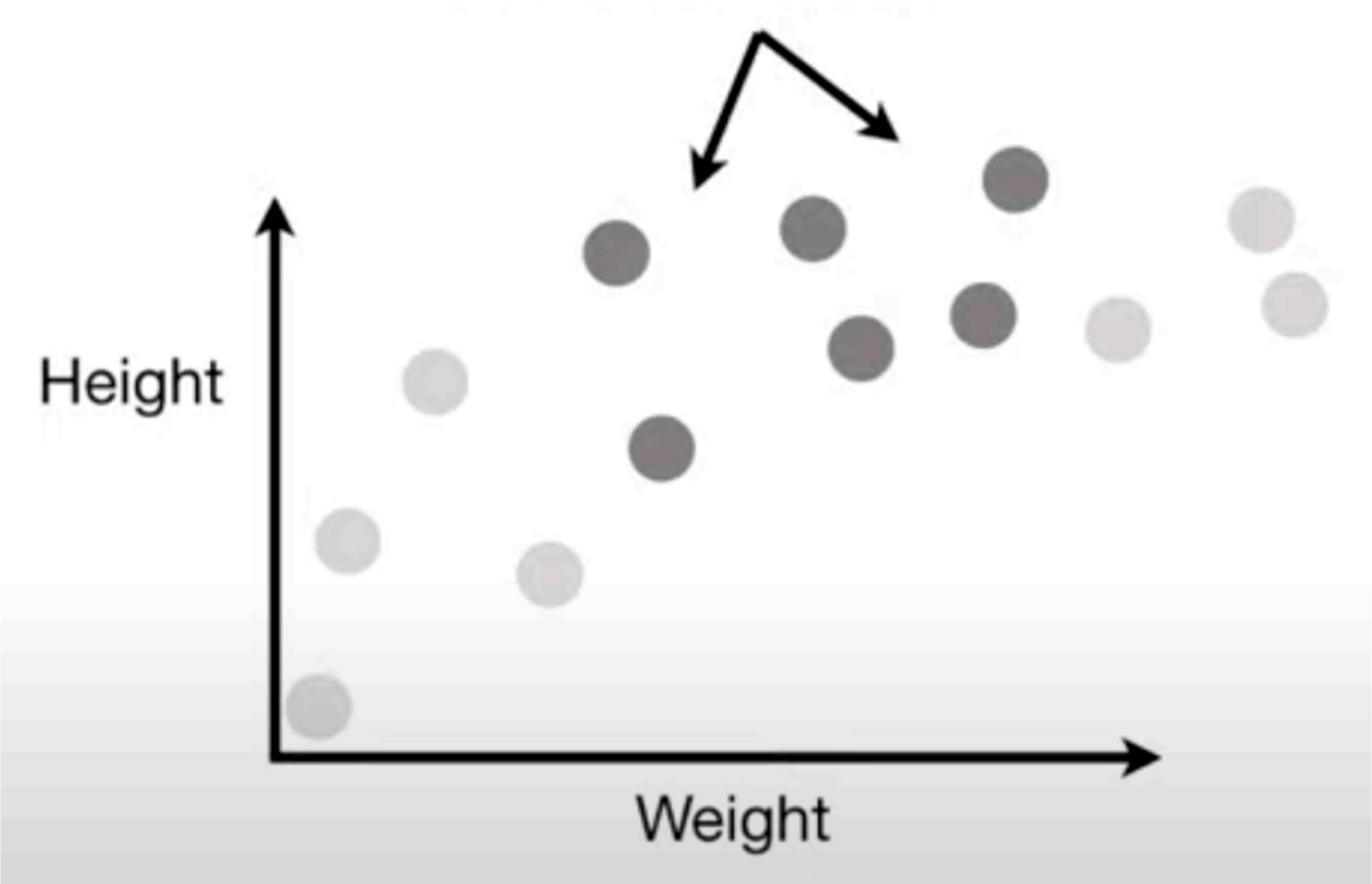
Example Graph: Weight & Height of Mice



Light mice tend to be short

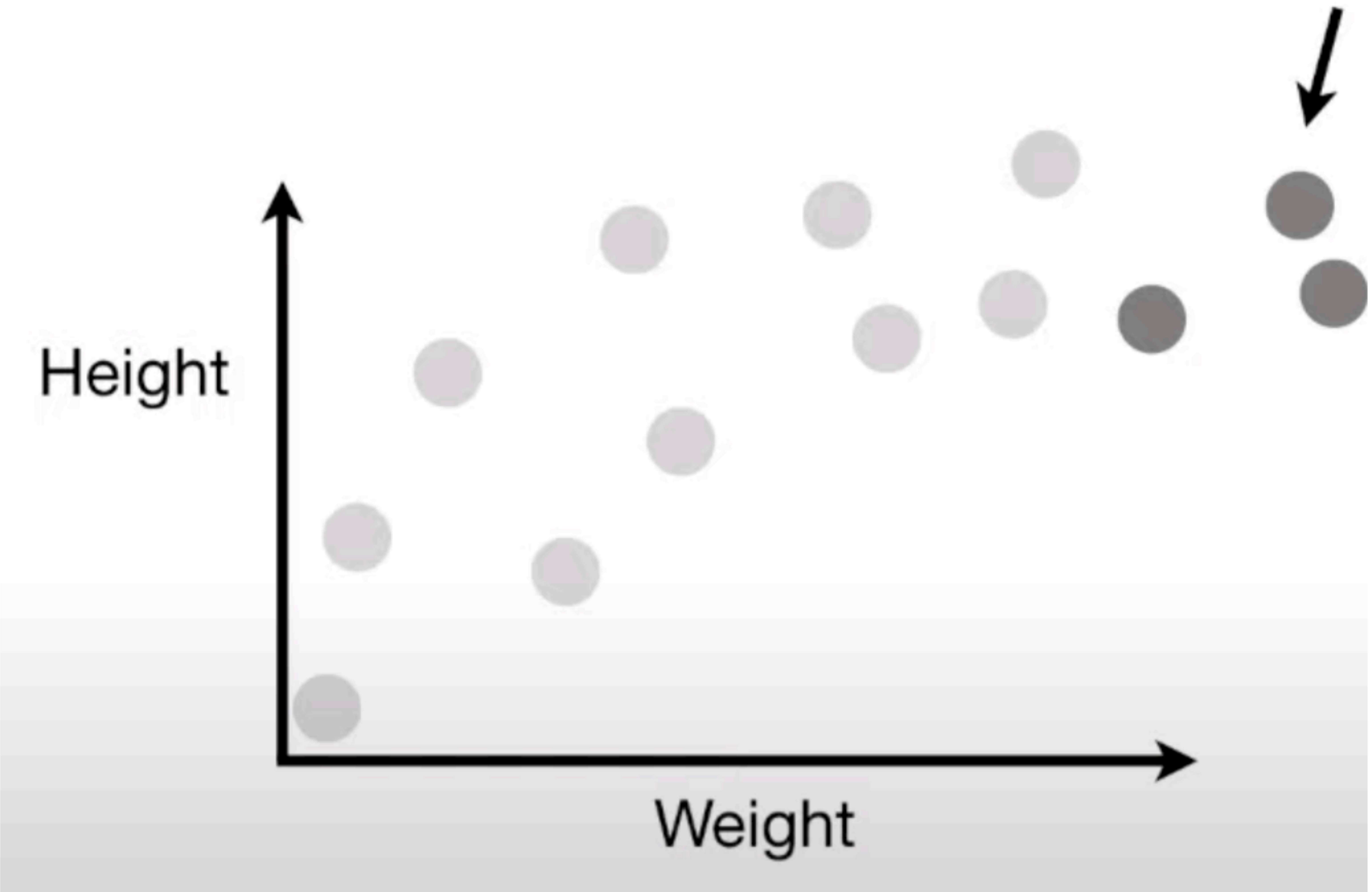


Heavier mice tend to be taller



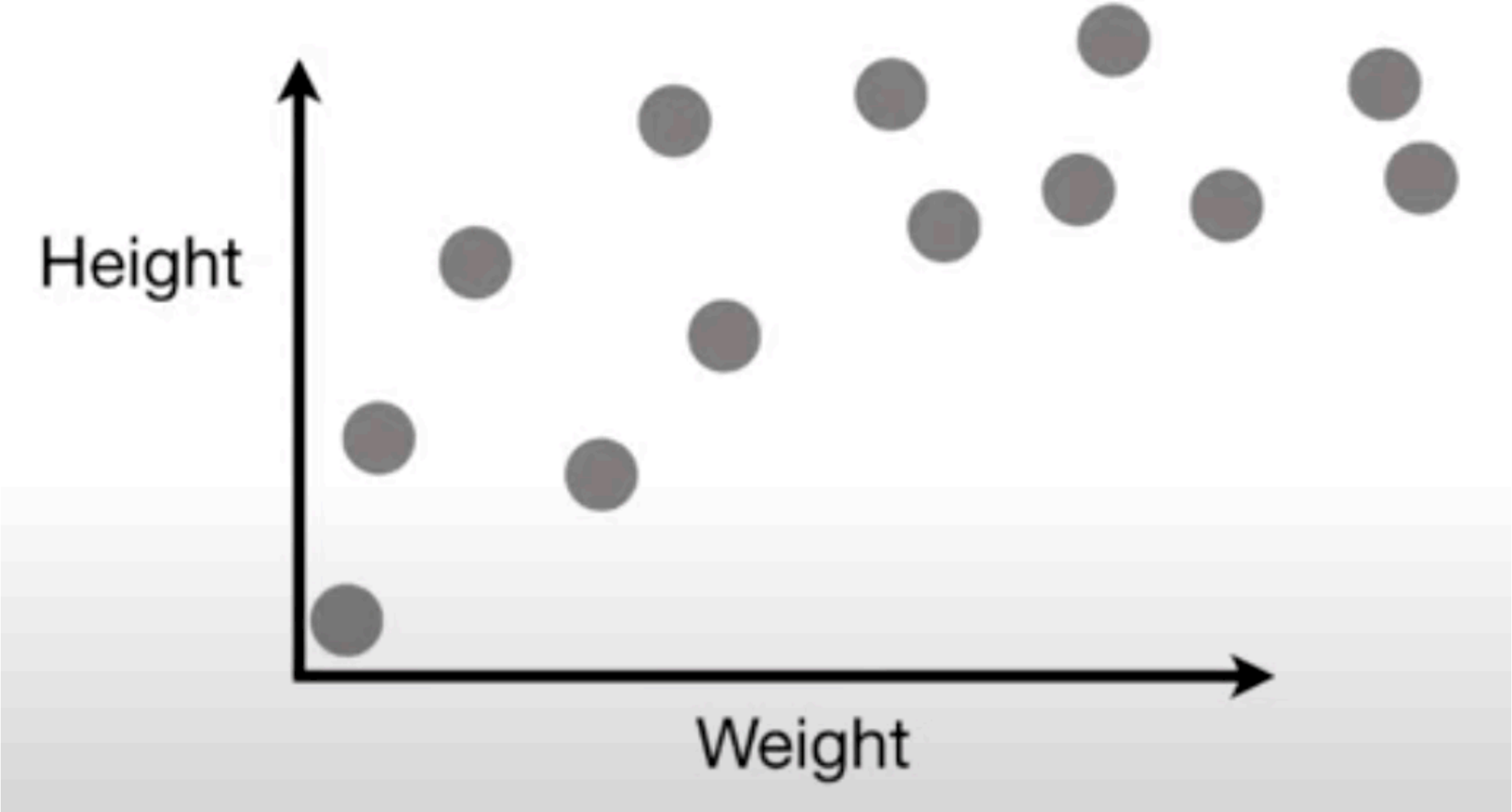
After a certain weight..

..mice don't get taller, they just get bigger



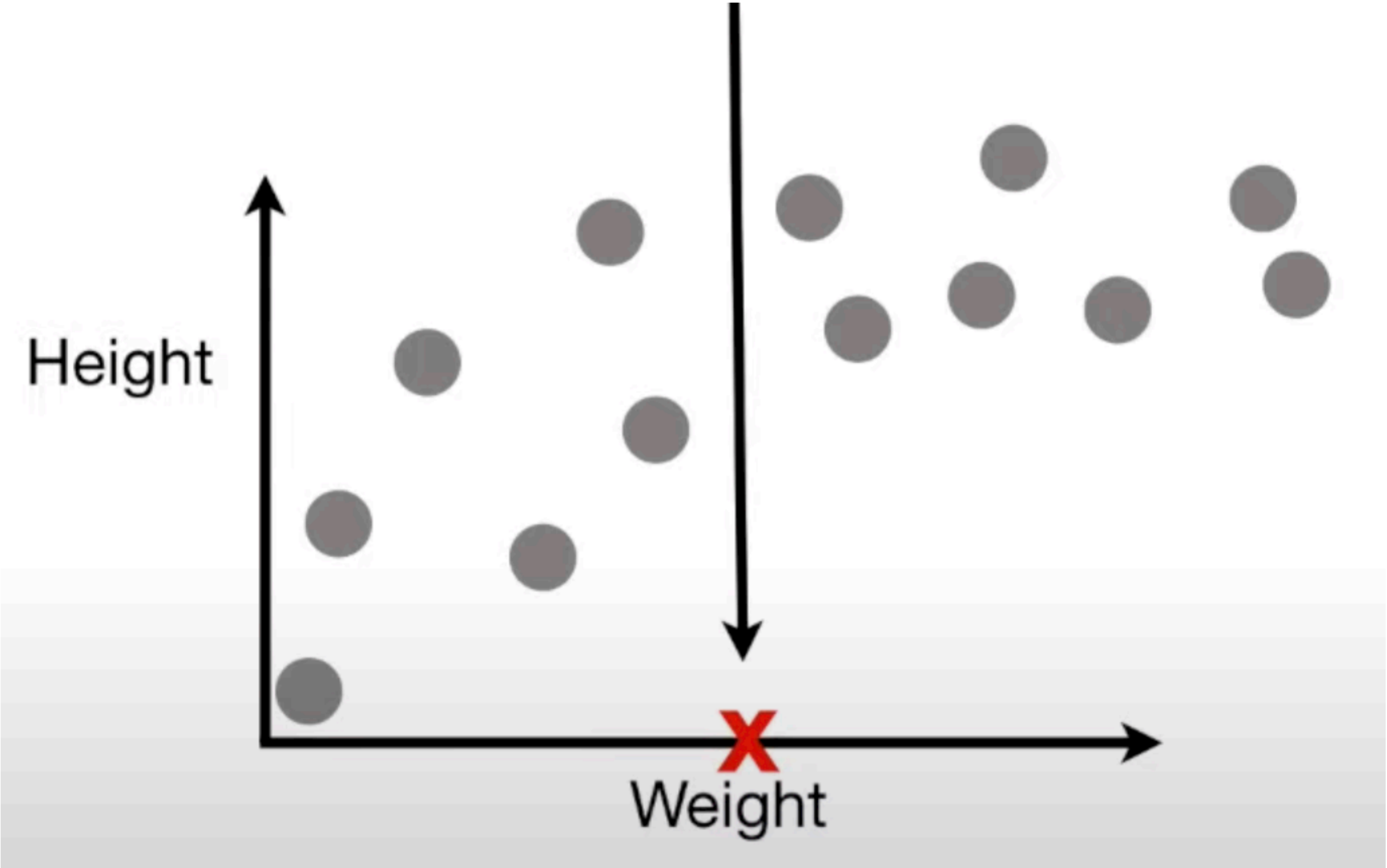
Given this data..

- Predict mouse height given its weight



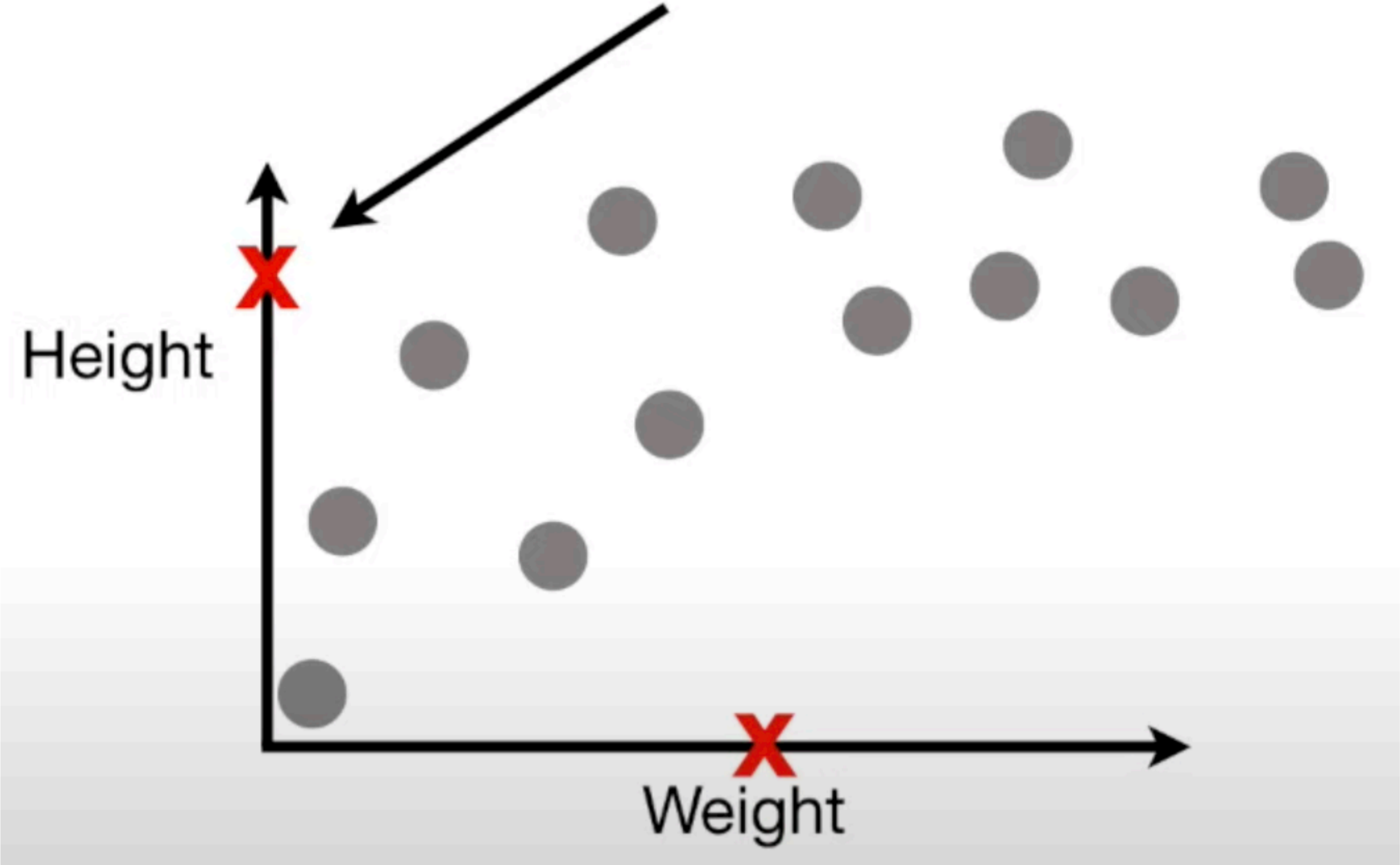
Example Prediction

If the mouse weighs this much ..



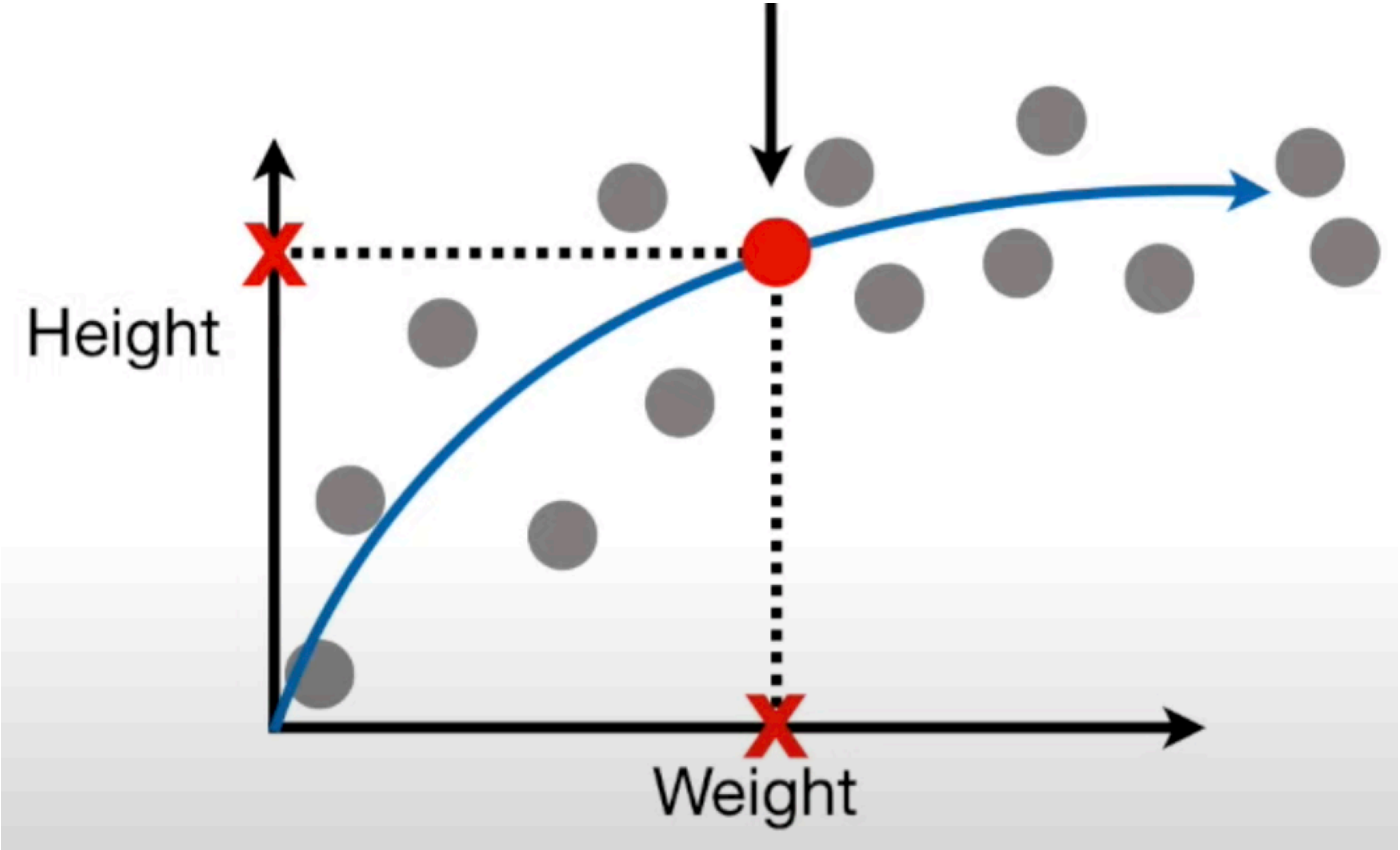
Example Prediction

We predict it’s this tall



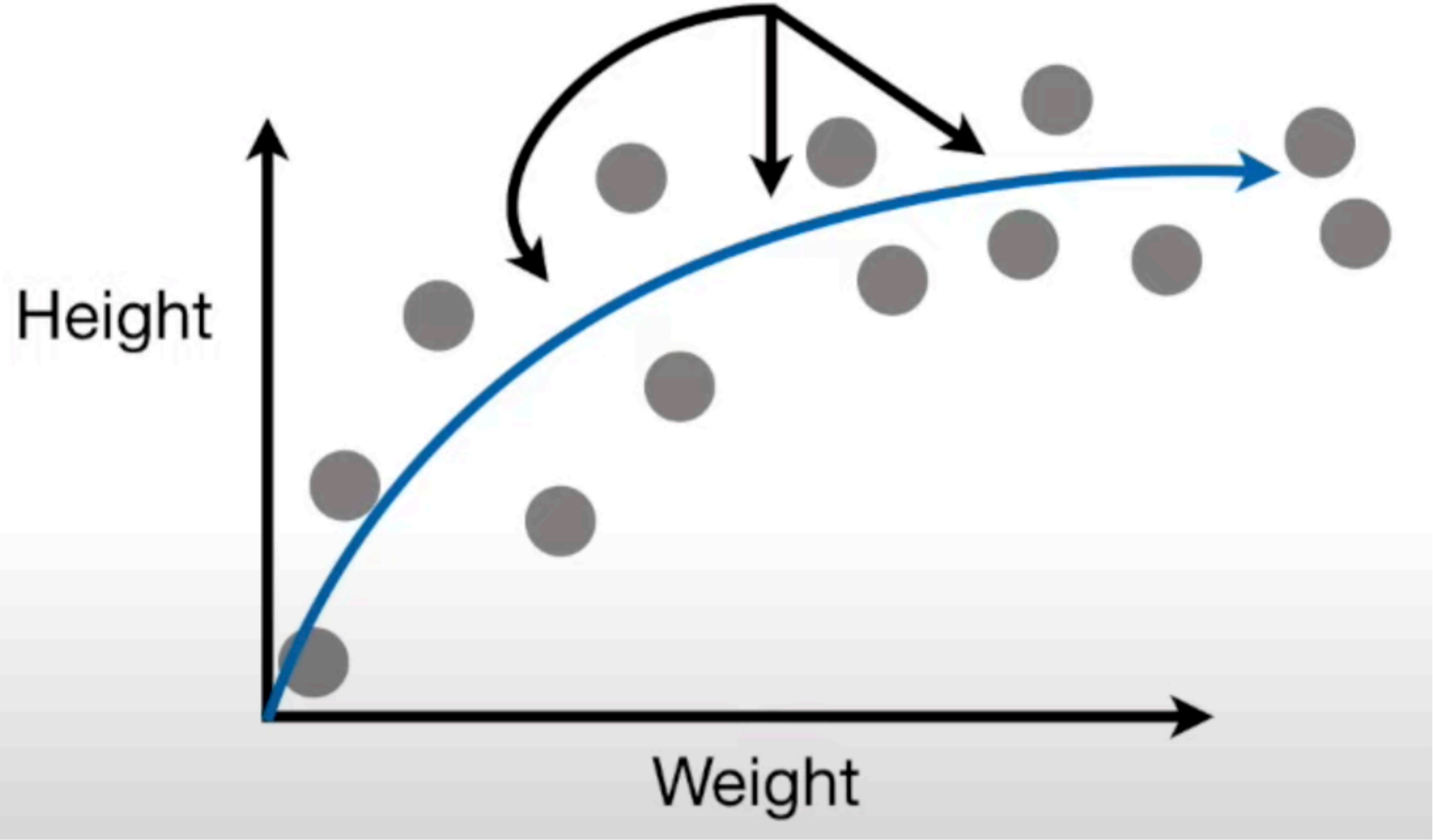
Hidden True Function

Ideally, we'd know the exact mathematical formula that describes the relationship between weight & height



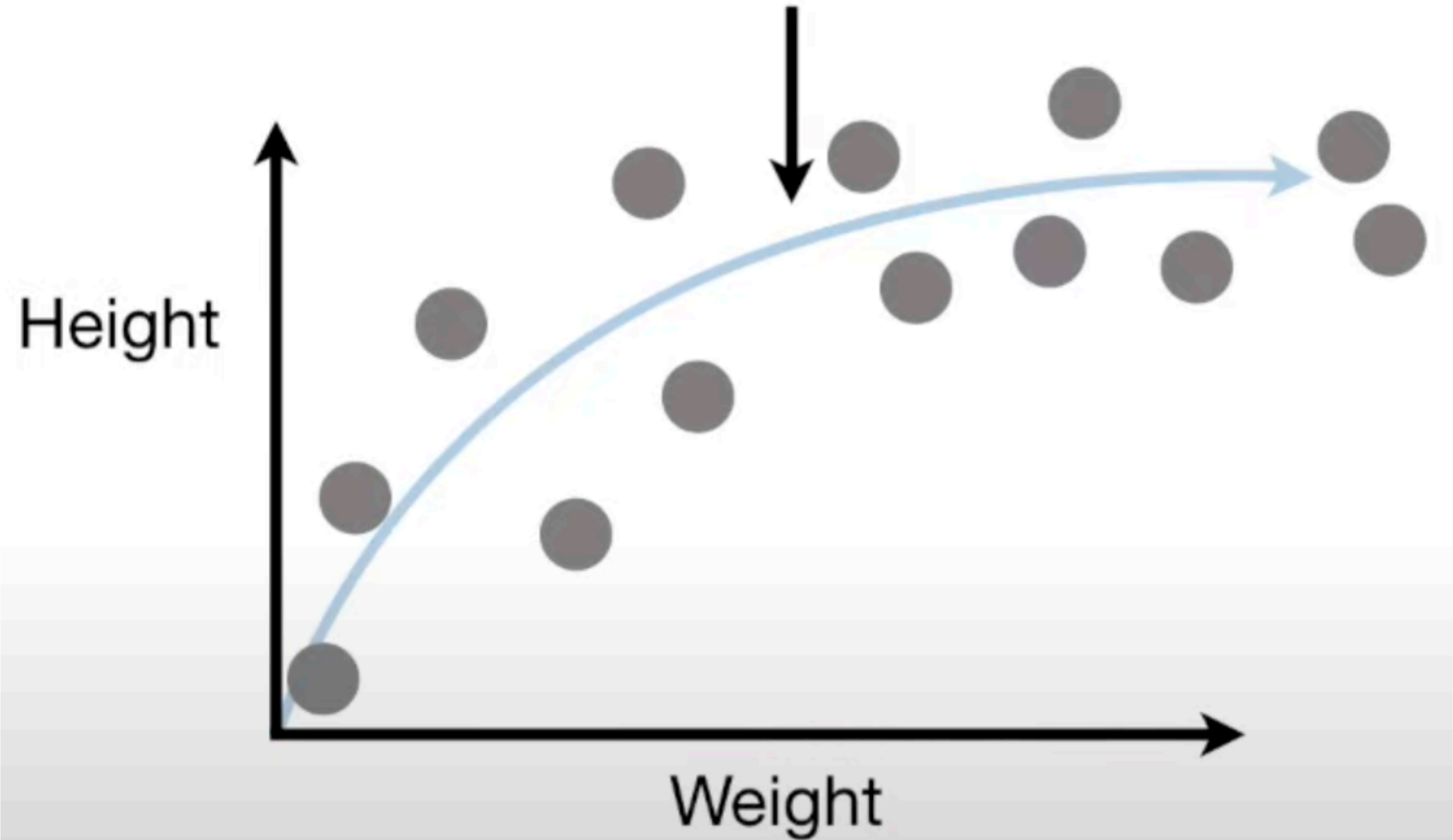
Hidden True Function

But we don’t know the formula
so we will use ML to approximate this relationship



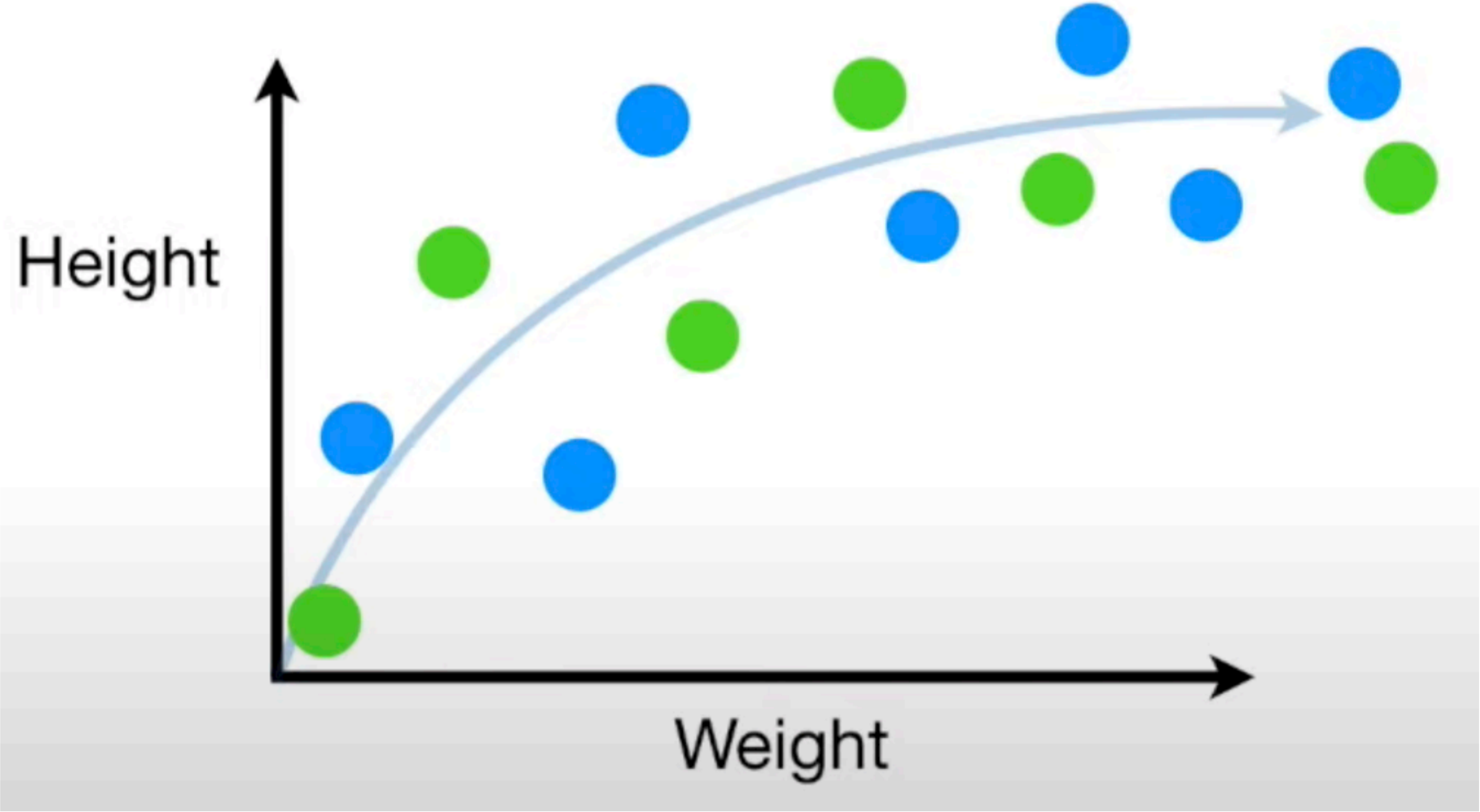
Hidden True Function

We will keep the true relationship curve for reference



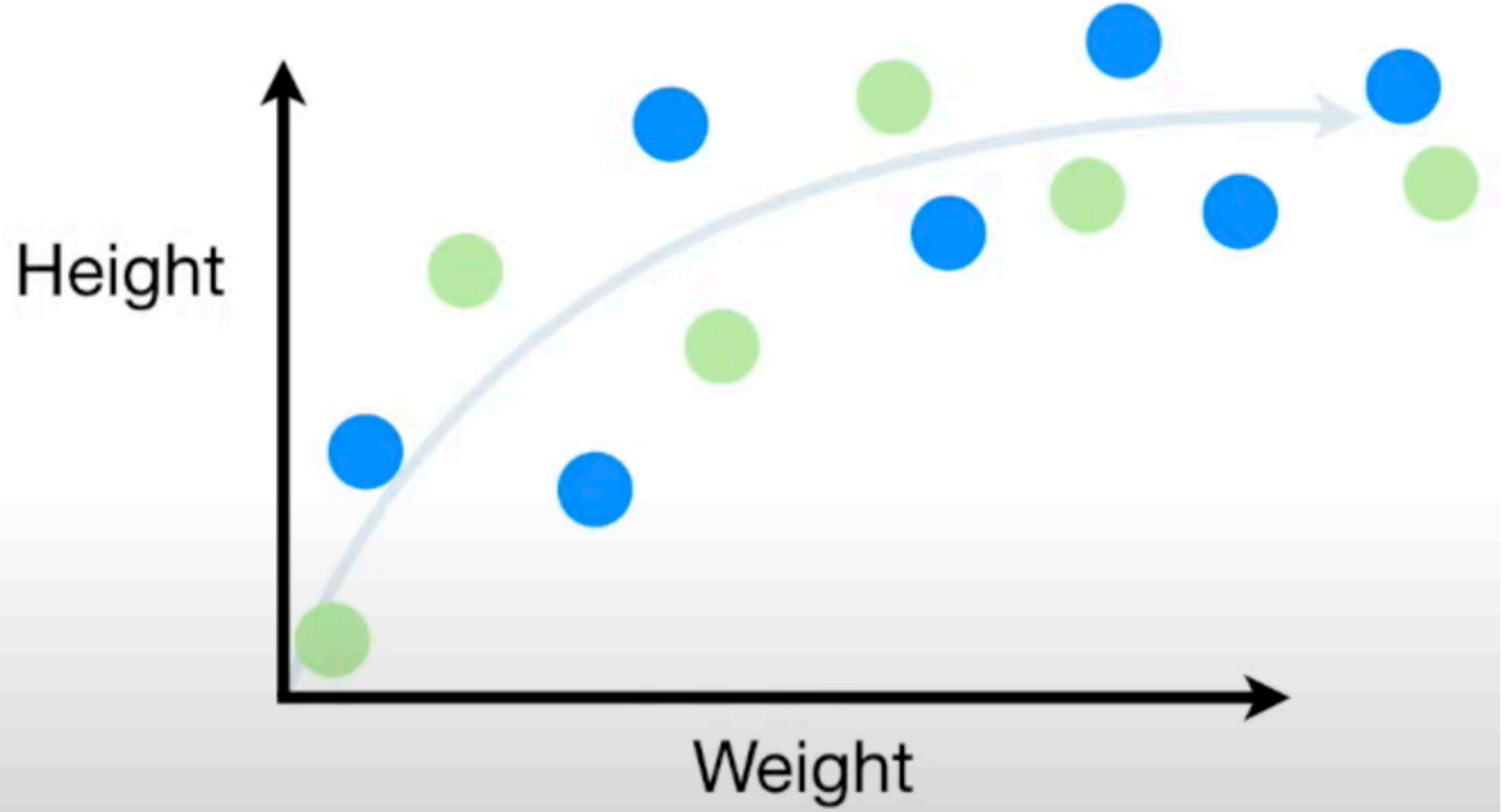
Train & Test Data

Split data into two sets:
one for training the ML algorithm & one for testing it



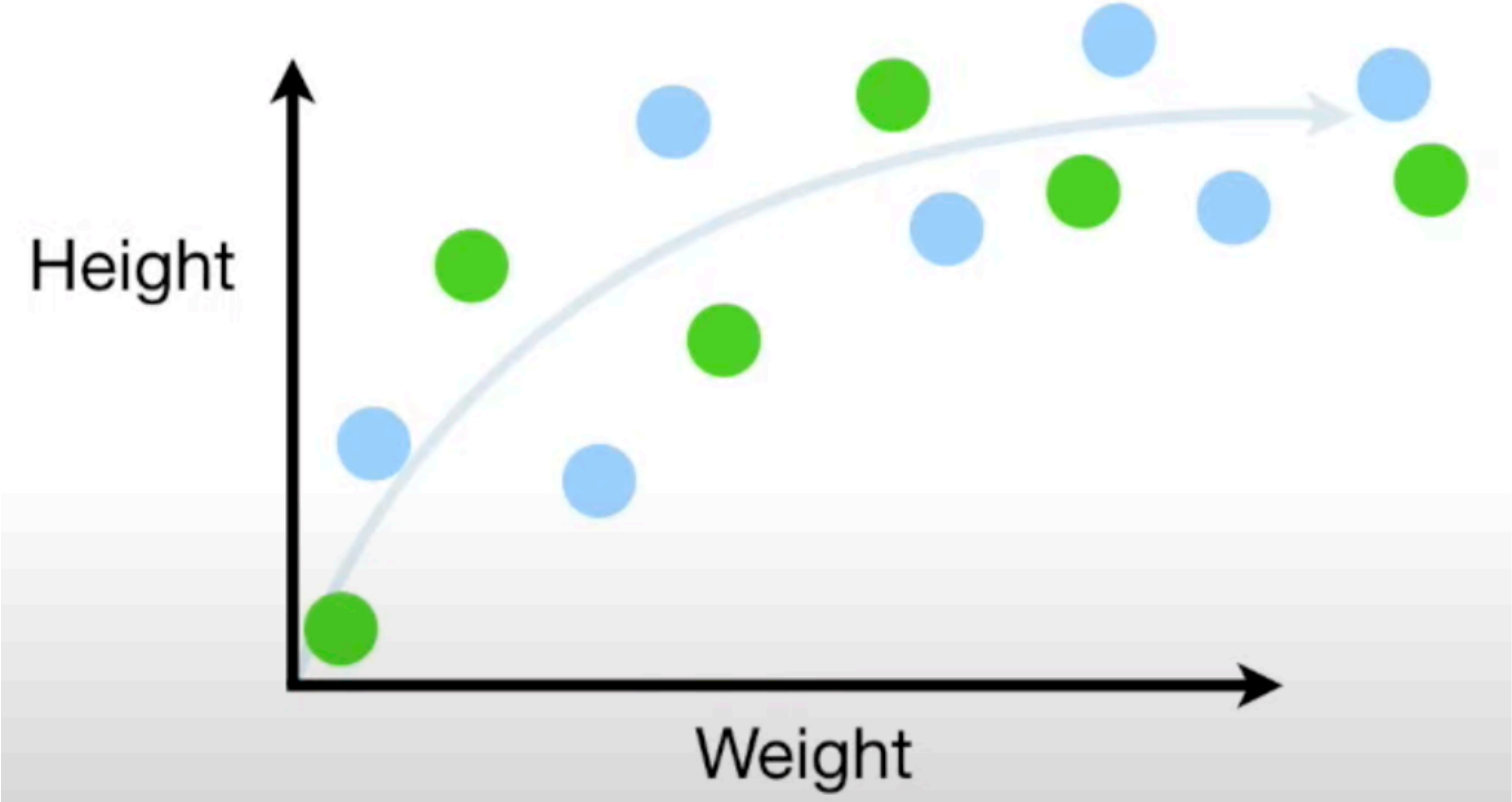
Training Data

- Blue dots

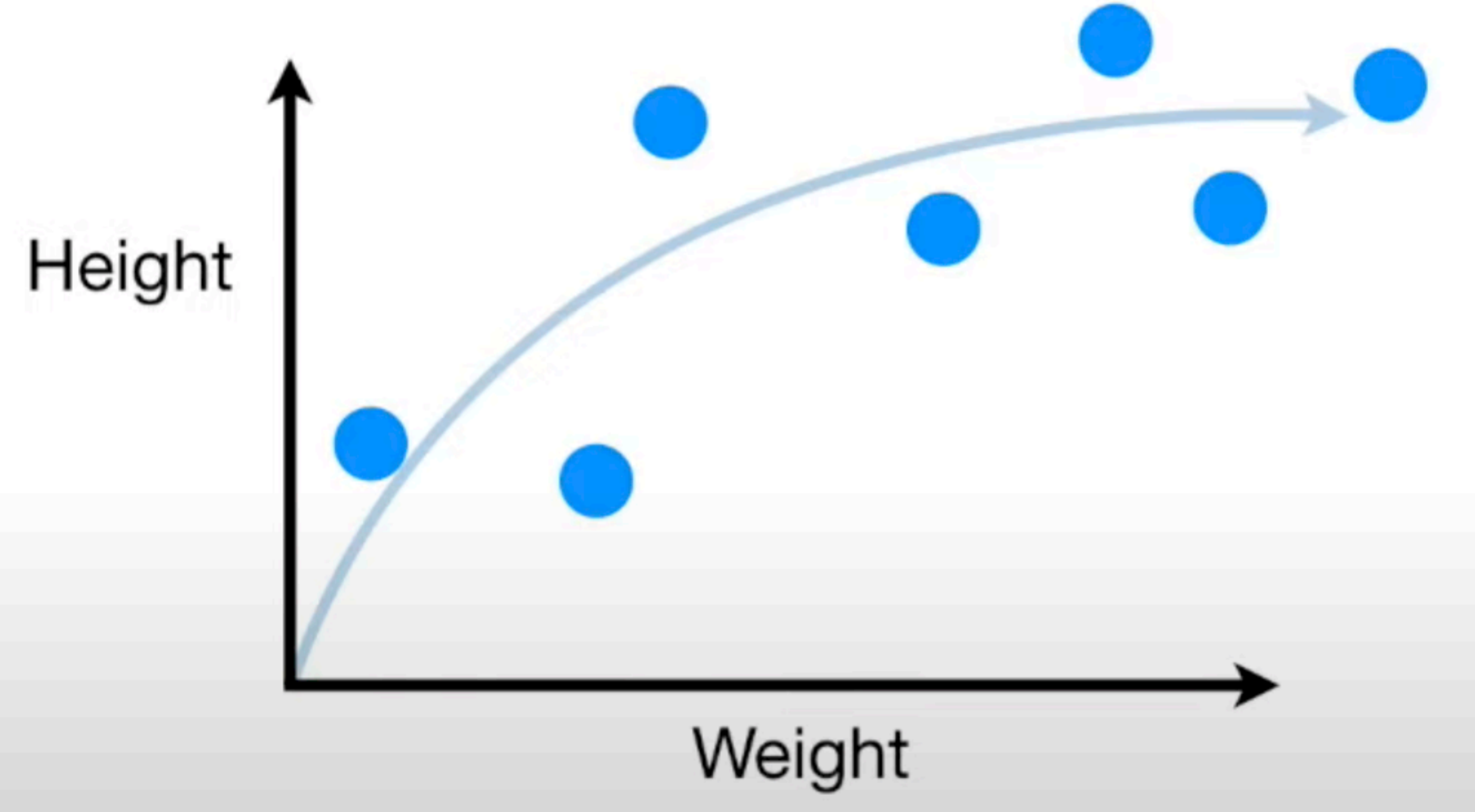


Testing Data

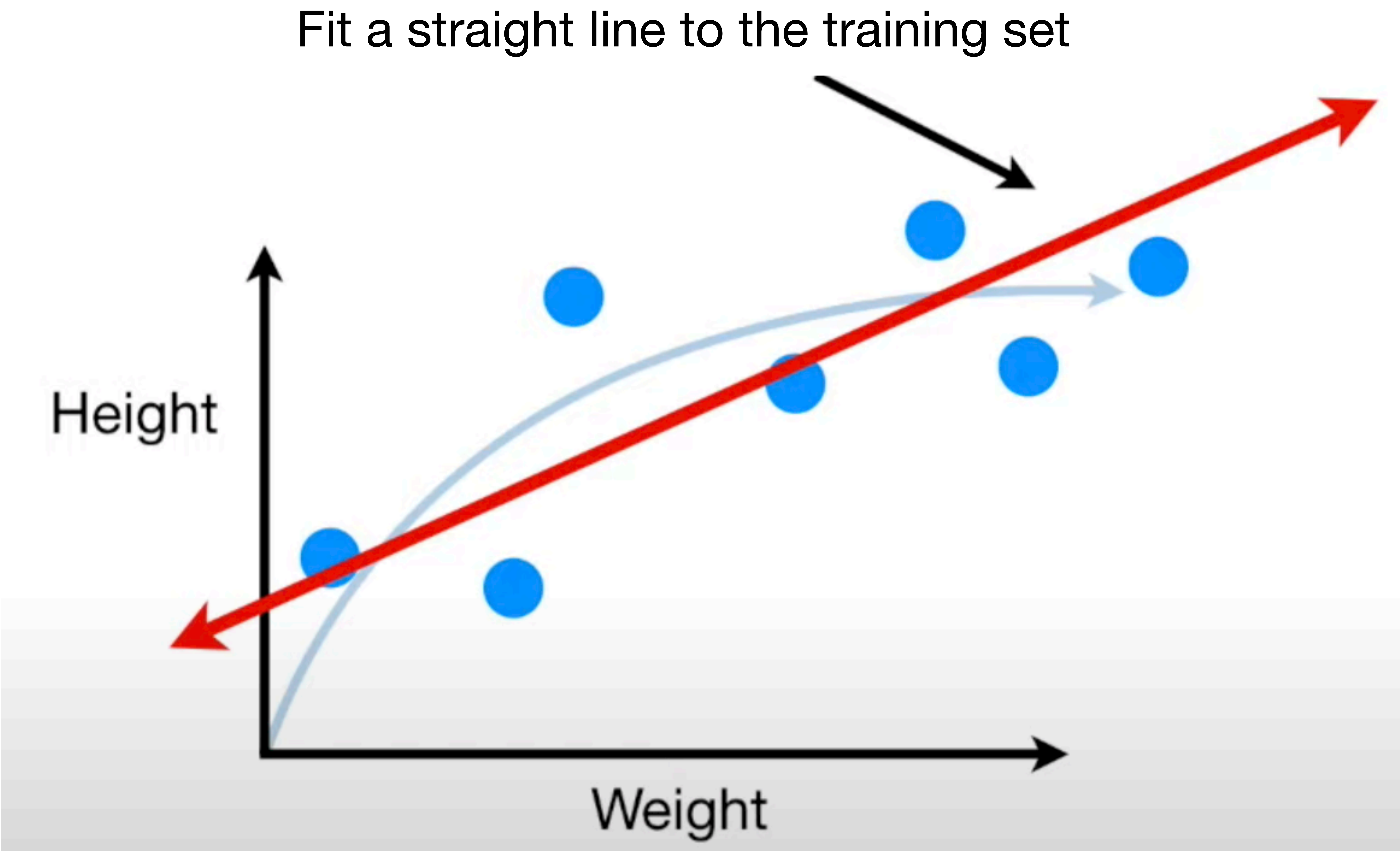
- Green dots



Training set

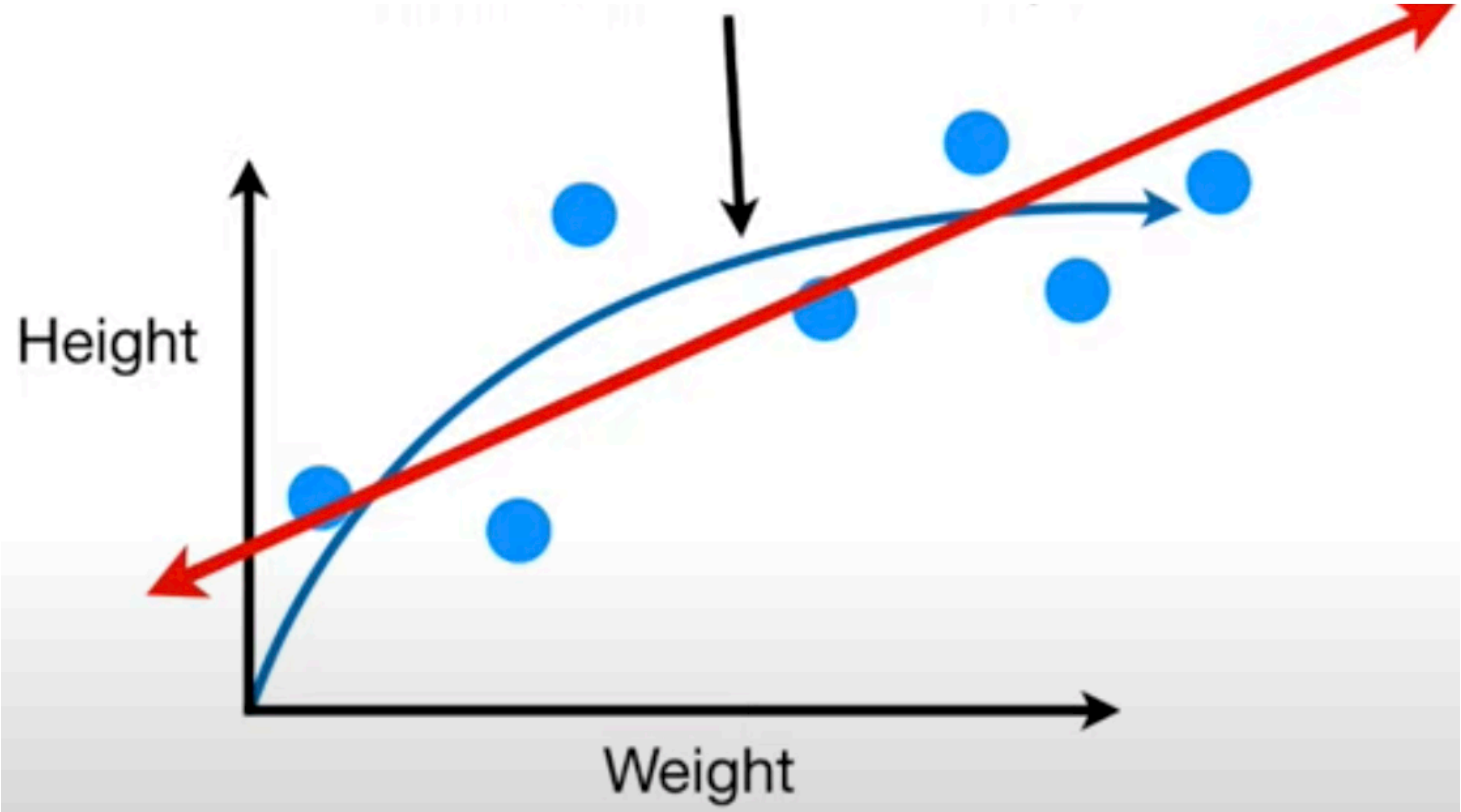


Linear Regression



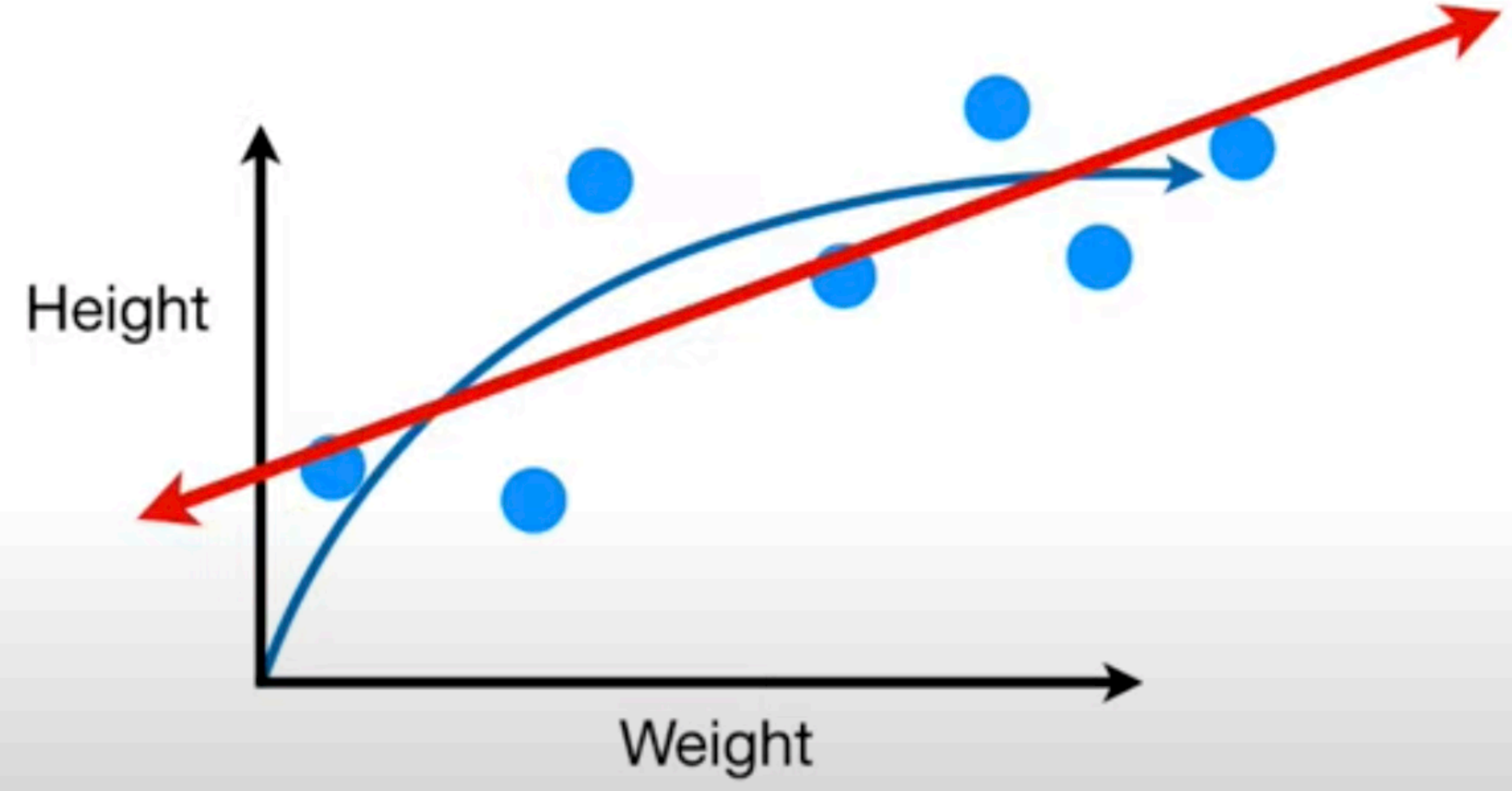
Linear Regression

The straight line is not flexible
Can’t accurately replicate the arc in the true relationship



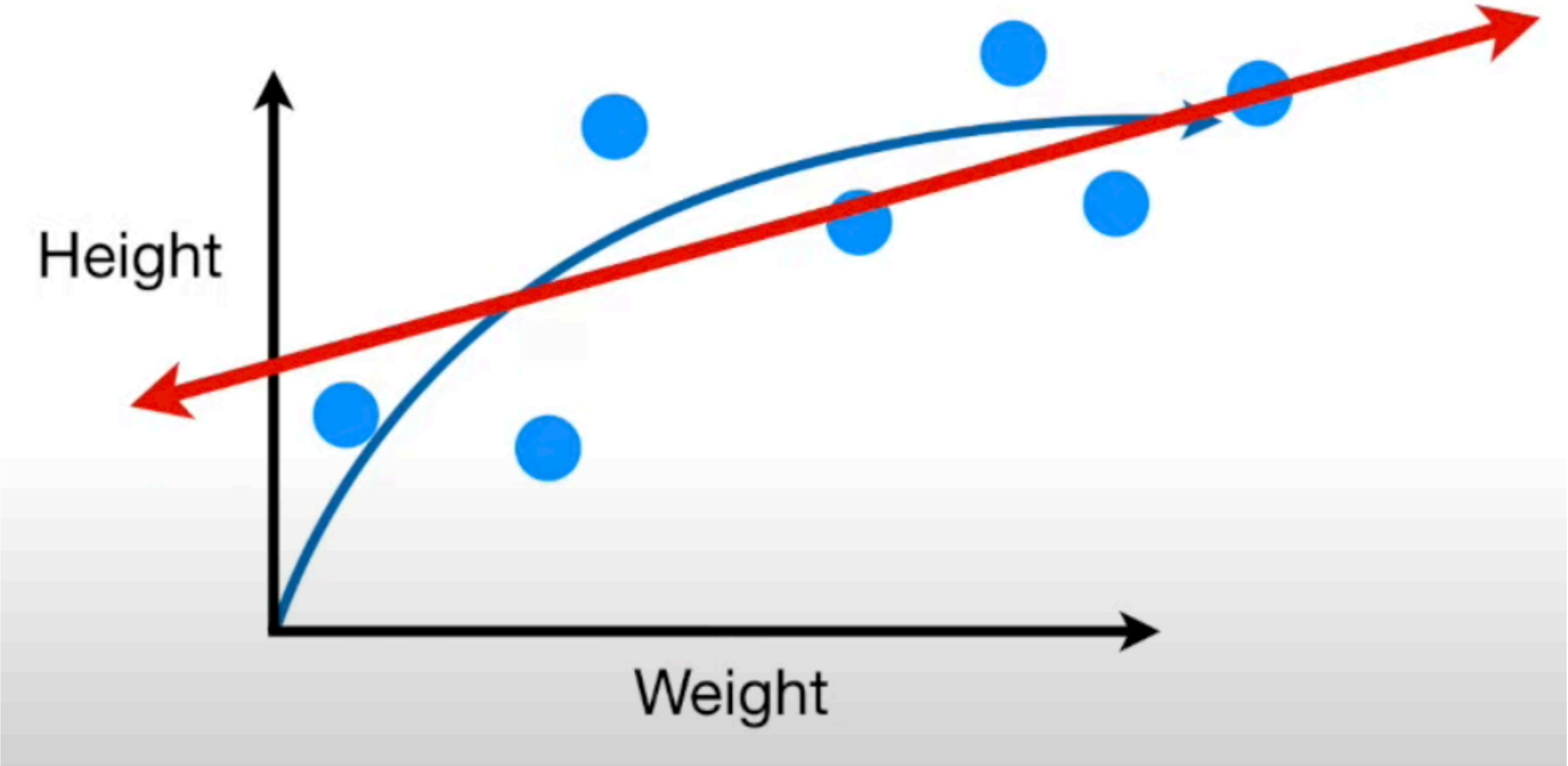
Linear Regression

No matter how we try to fit the line..
it will never curve



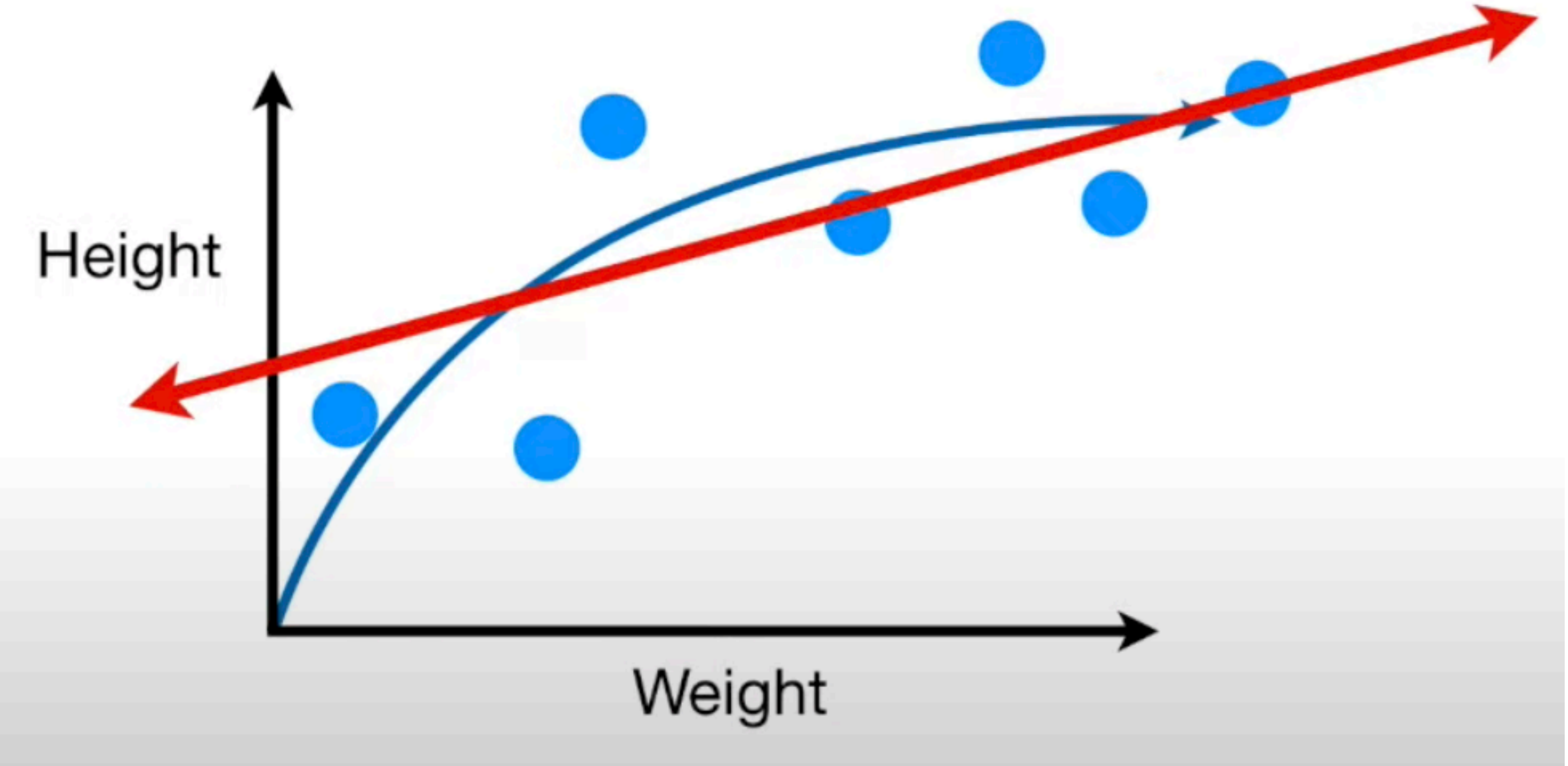
Linear Regression

The straight line will never capture the
true relationship between weight & height



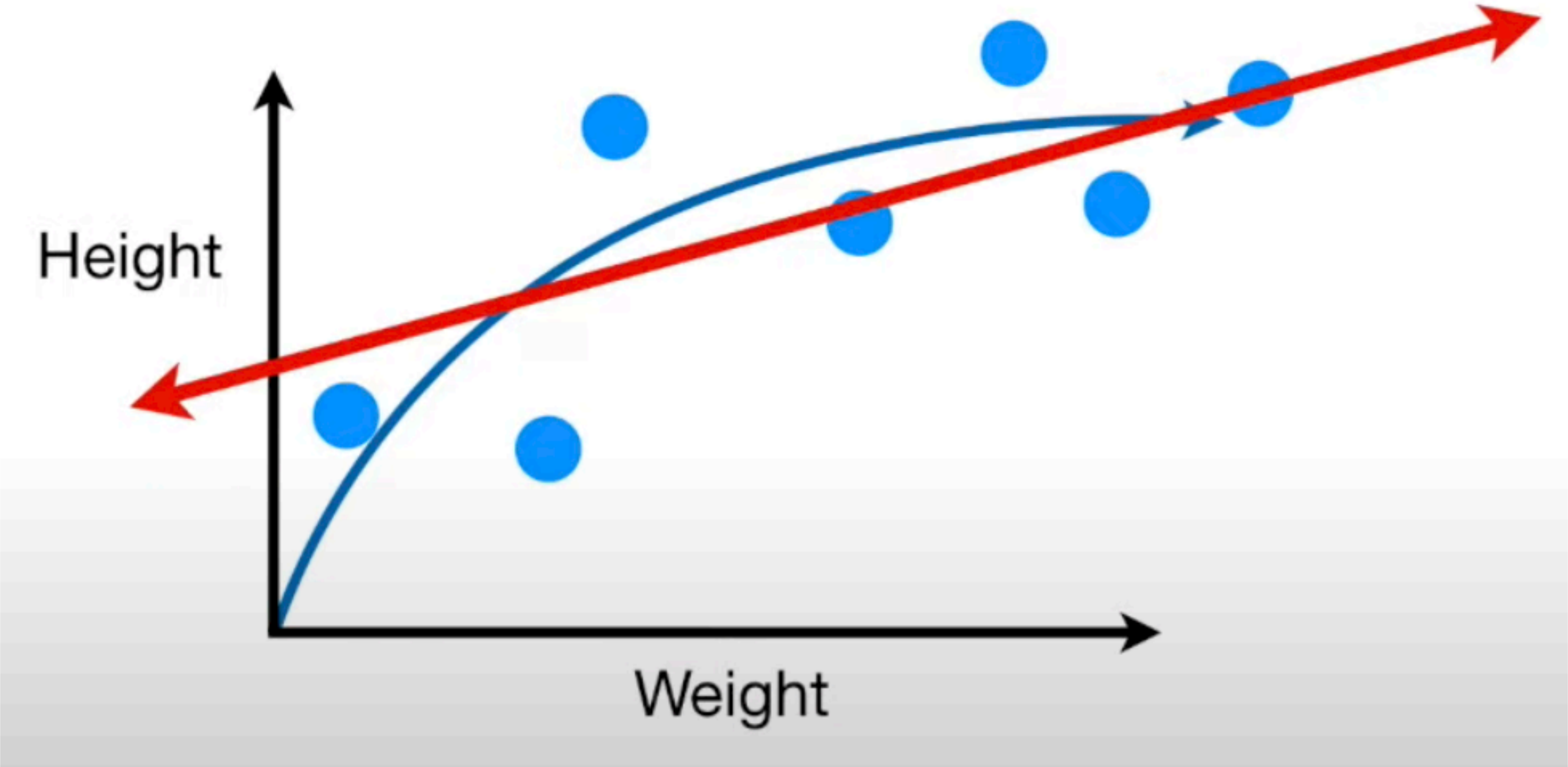
Linear Regression

The inability to capture the true relationship is called **BIAS**

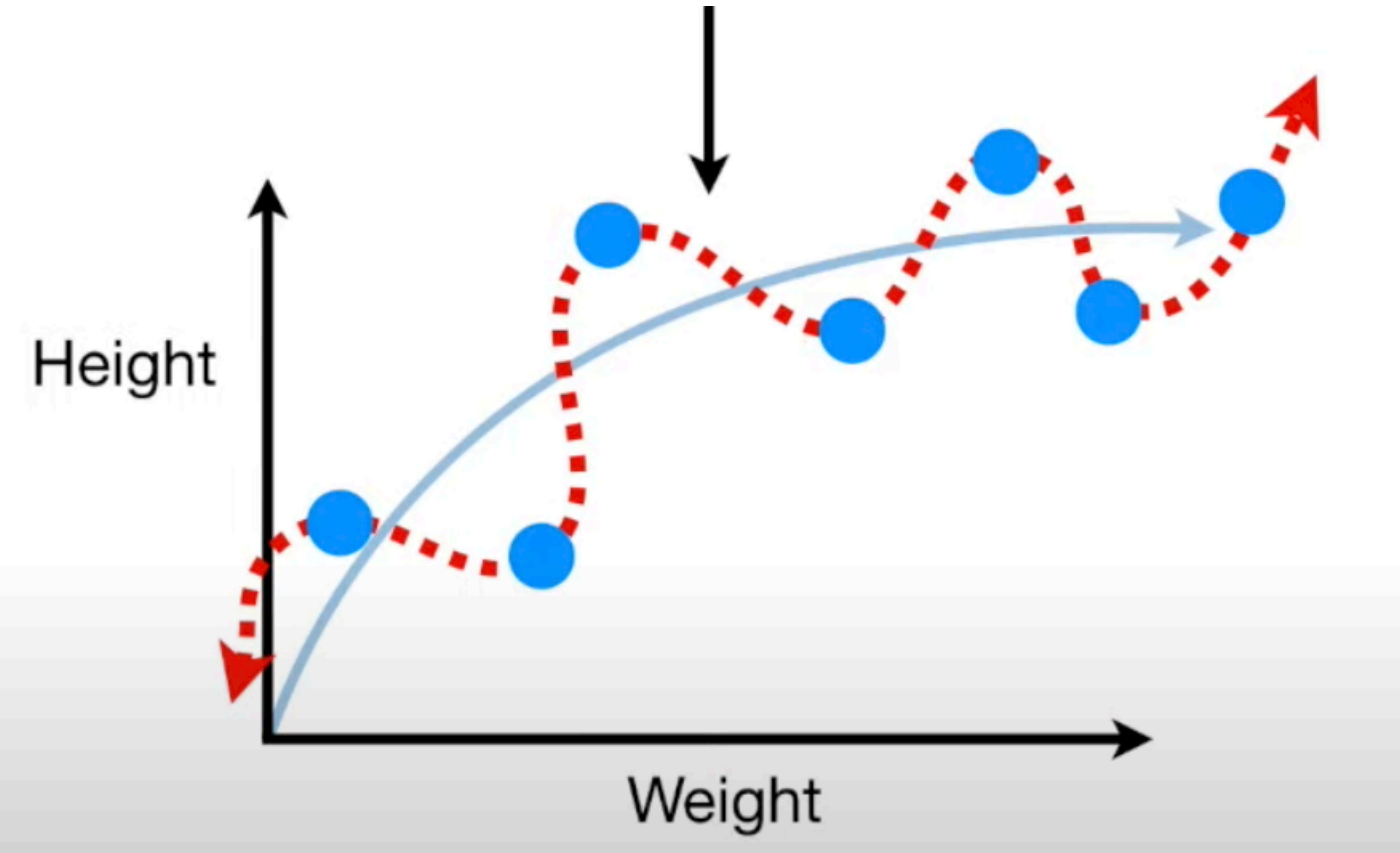


Linear Regression

The straight line can't be curved like the true relationship, so it has a large amount of **BIAS**

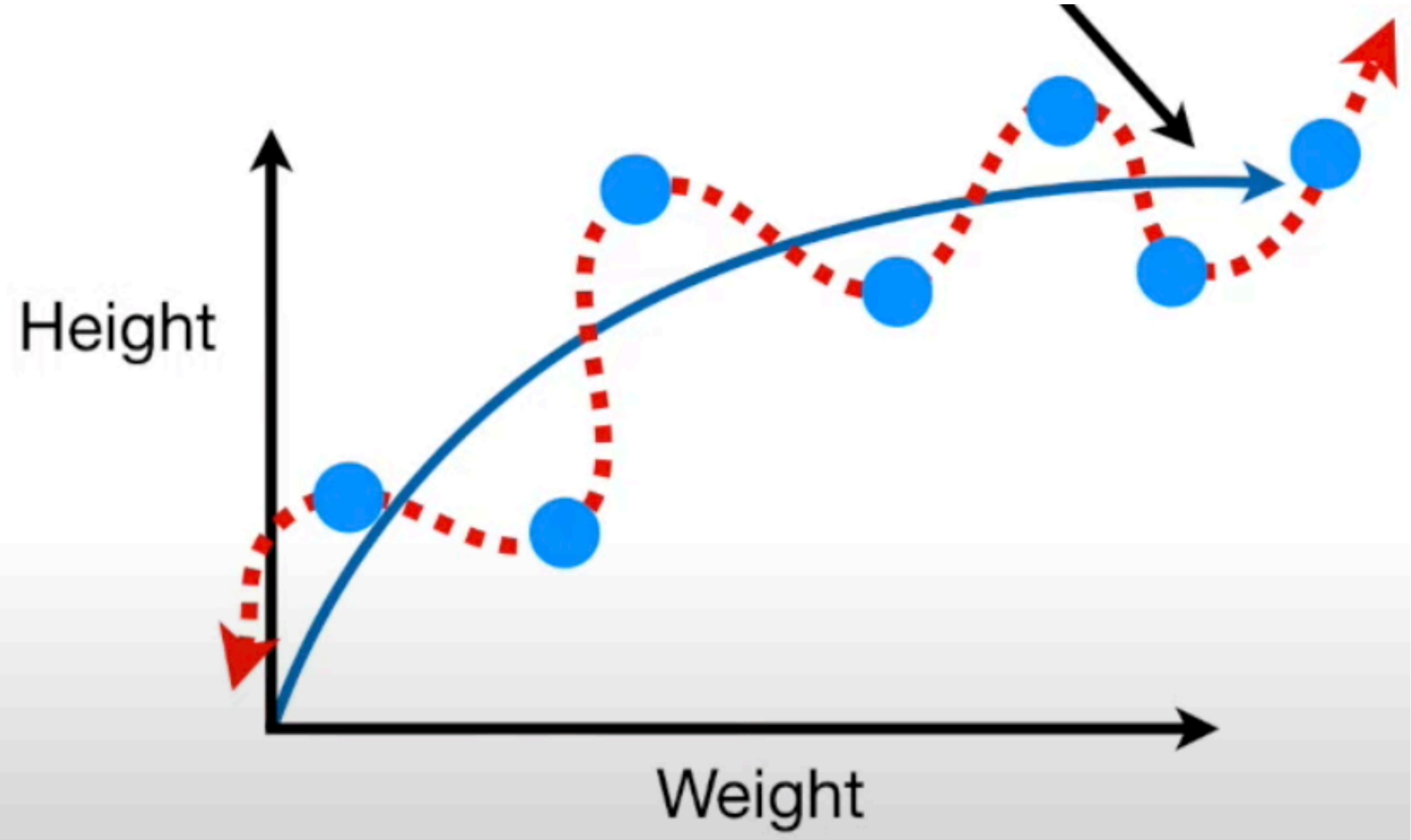


Fit a squiggly line instead?



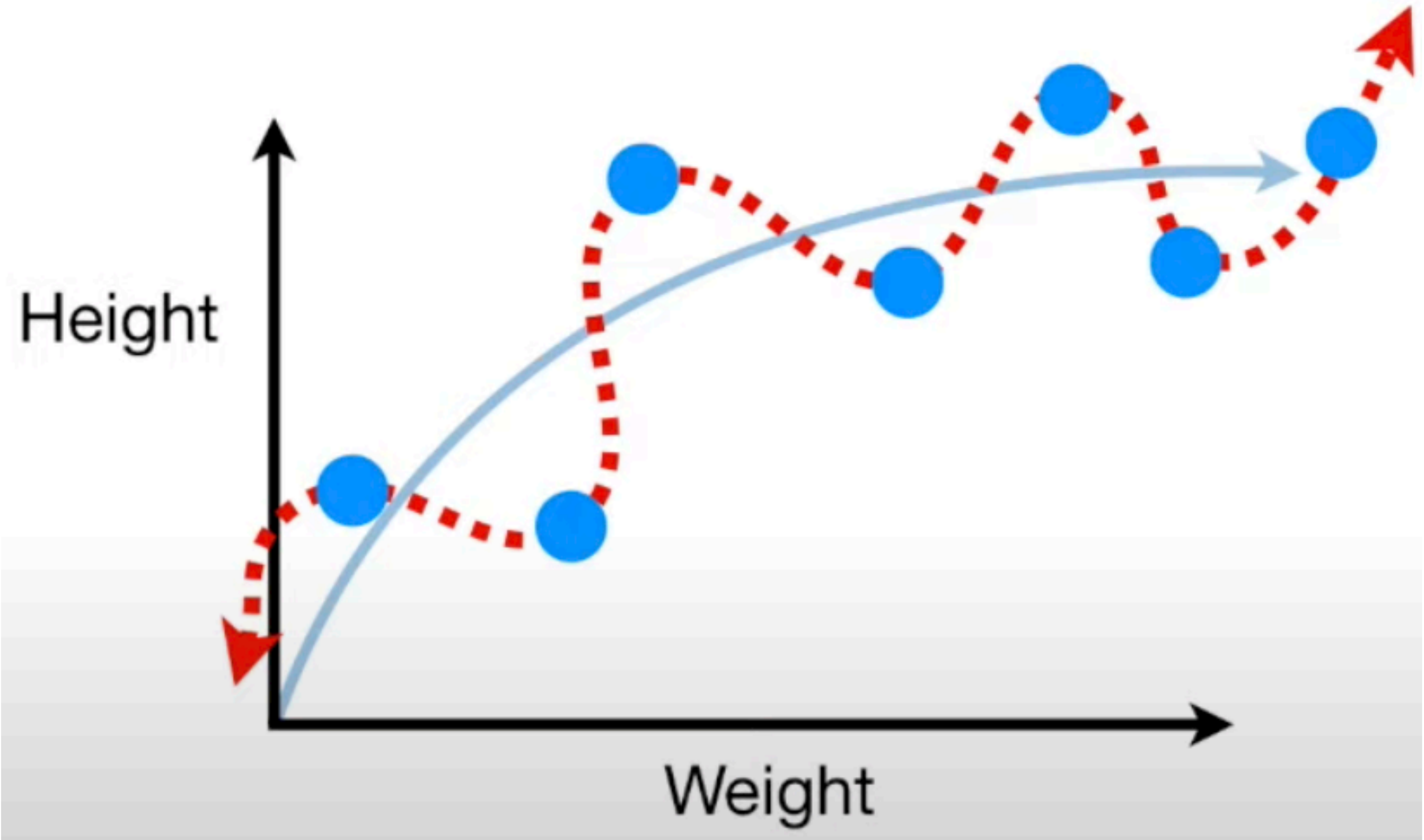
Fit a squiggly line instead?

Flexible & fits data along arc of the true relationship



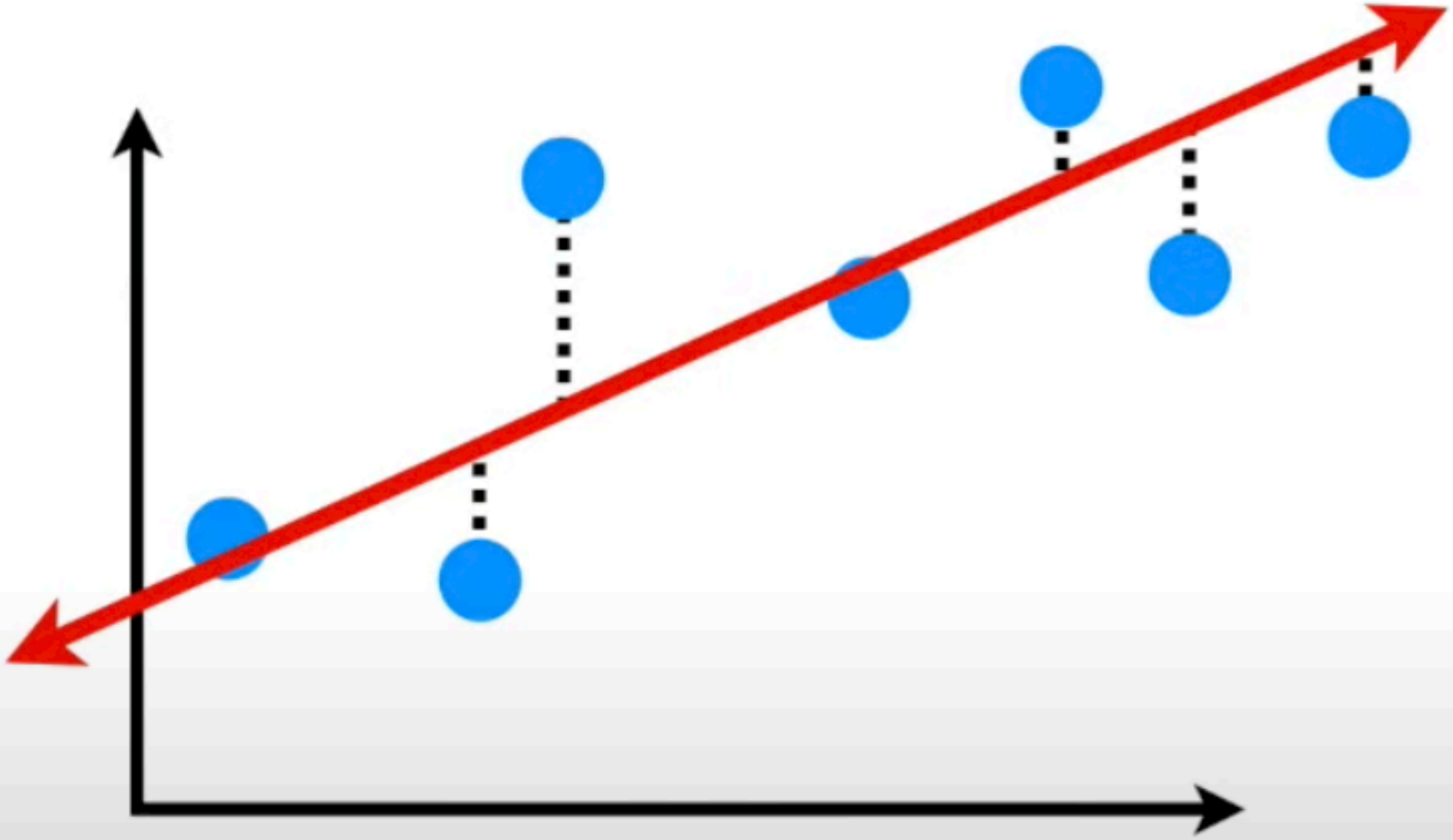
Fit a squiggly line instead?

Fits arc in true relationship between weight & height
—> very little **BIAS**

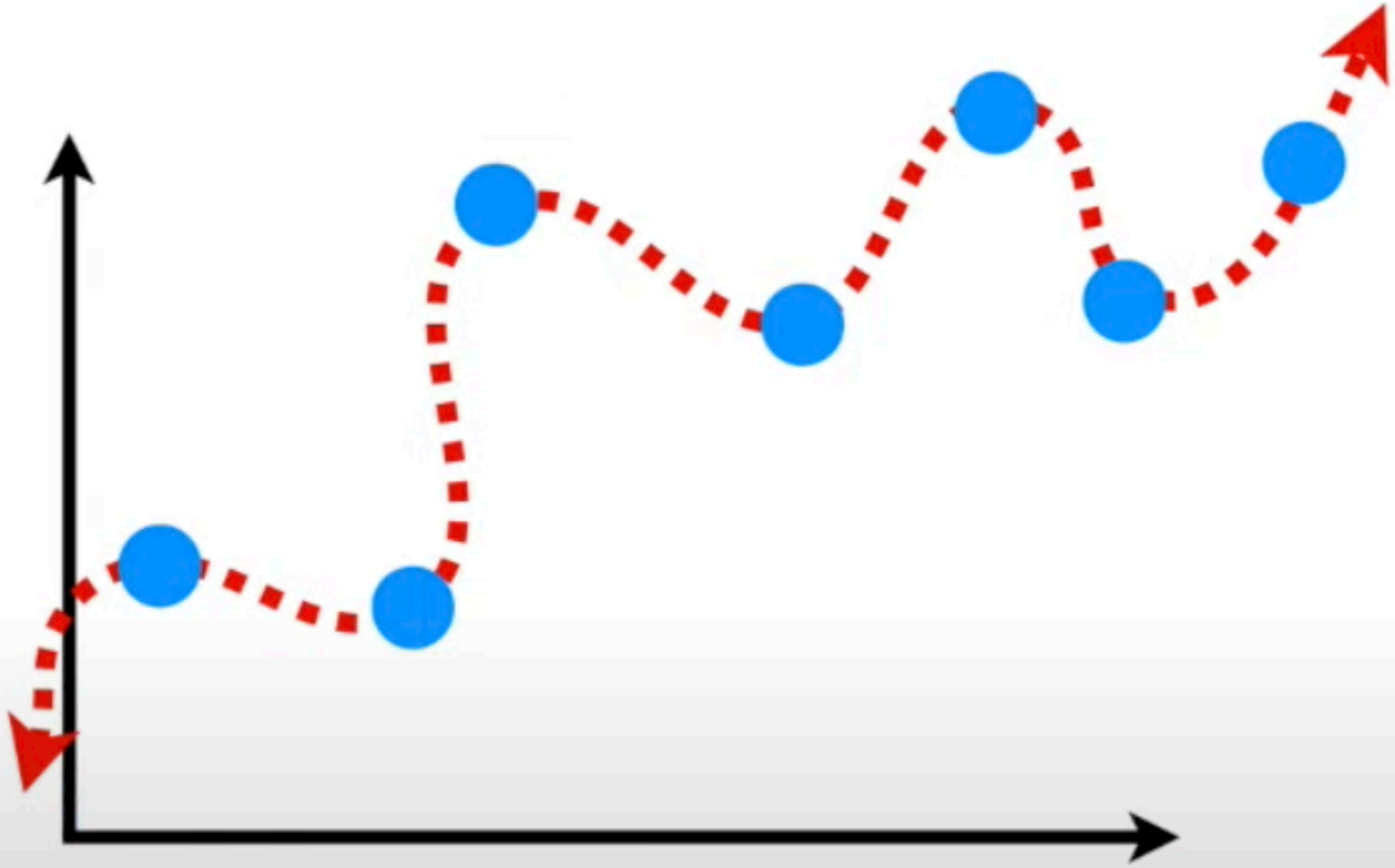


Straight vs. Squiggly

How well each fits the training set: sum of squares of distance from fit line to data



Fits **training** data well
Sum of squares is zero



What about the testing set?

Straight vs. Squiggly

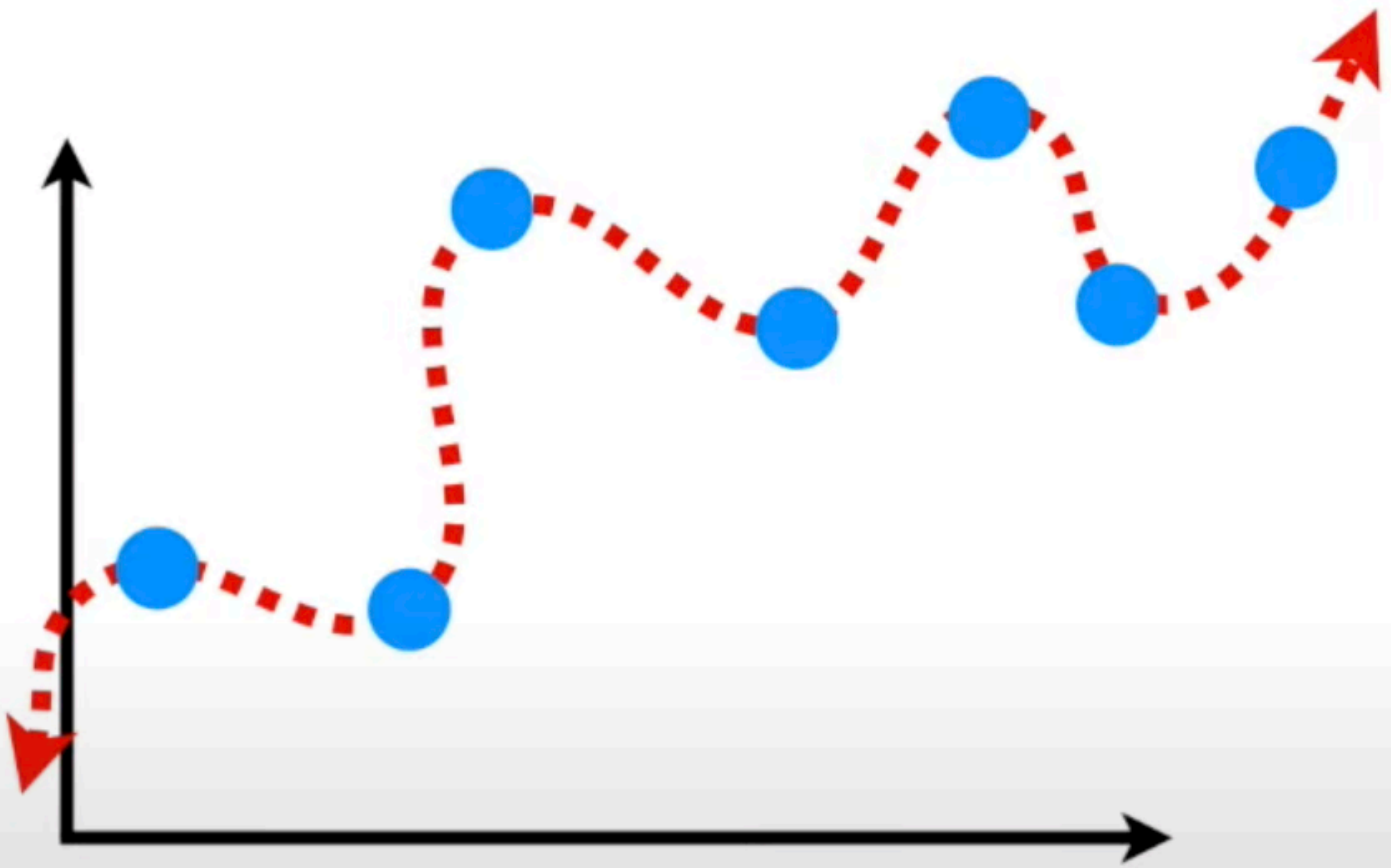
Sum of squares for testing set?

Straight vs. Squiggly

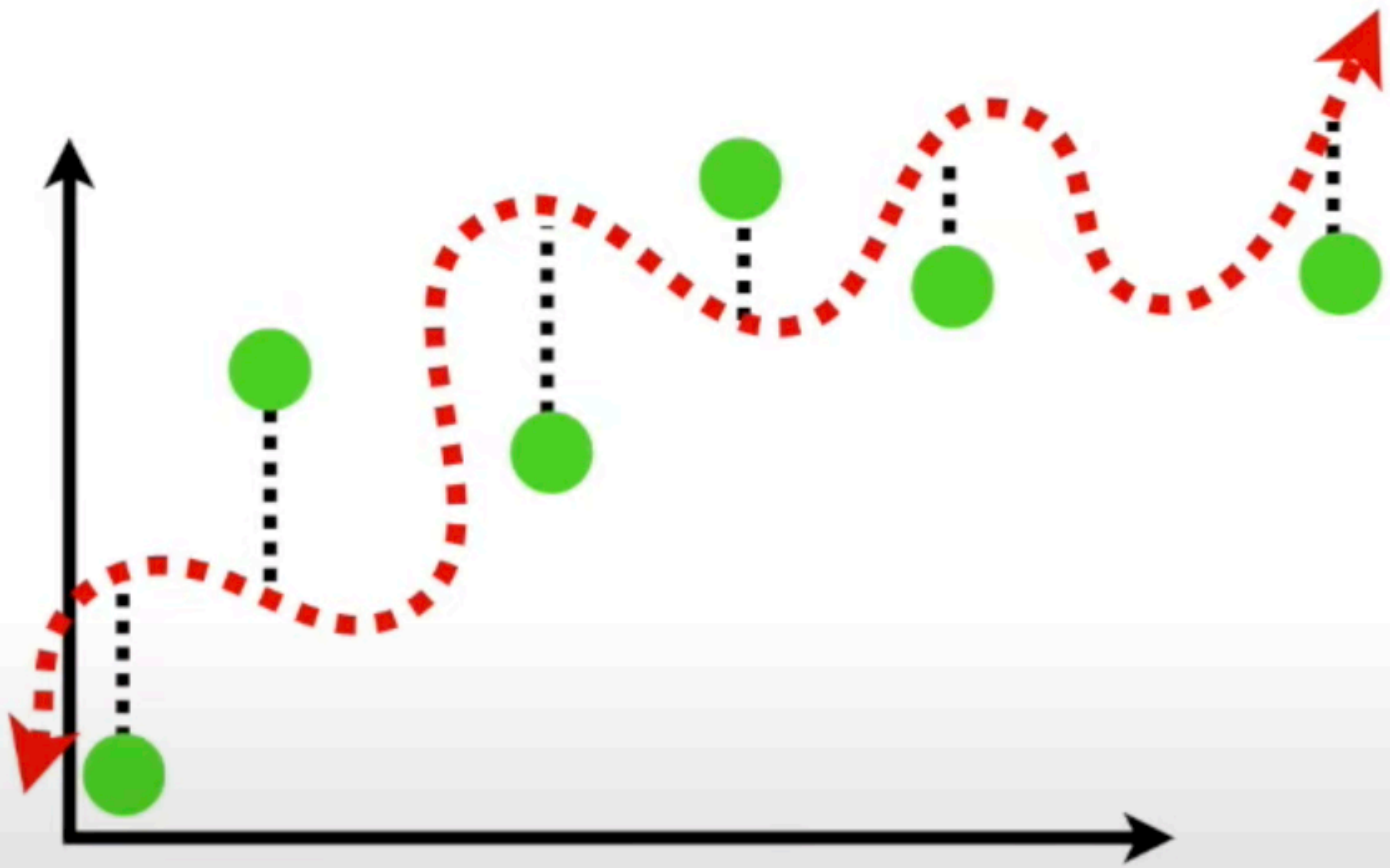
Sum of squares is smaller
Fits **testing** data better

Squiggly

Fit **train** data well

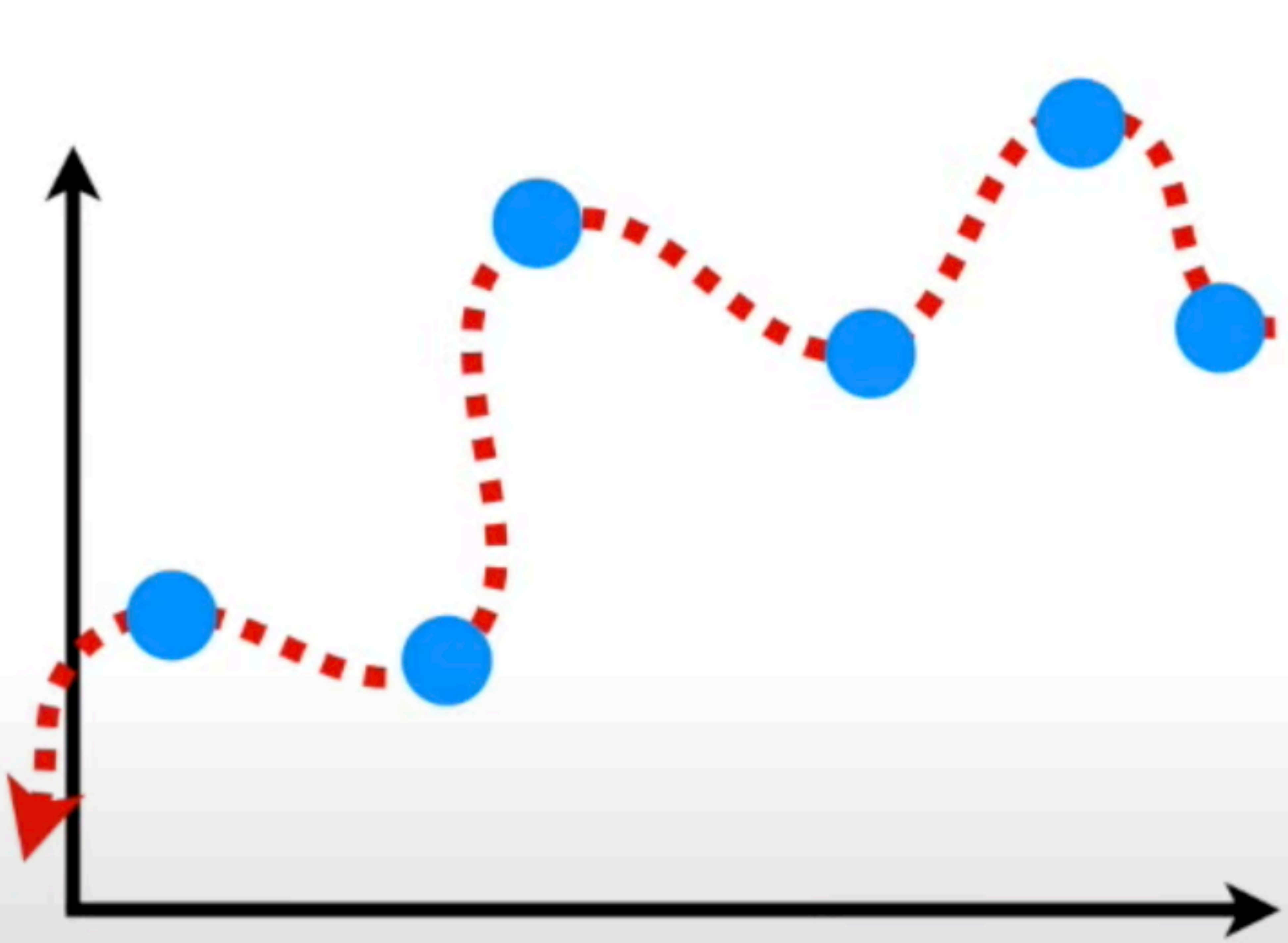


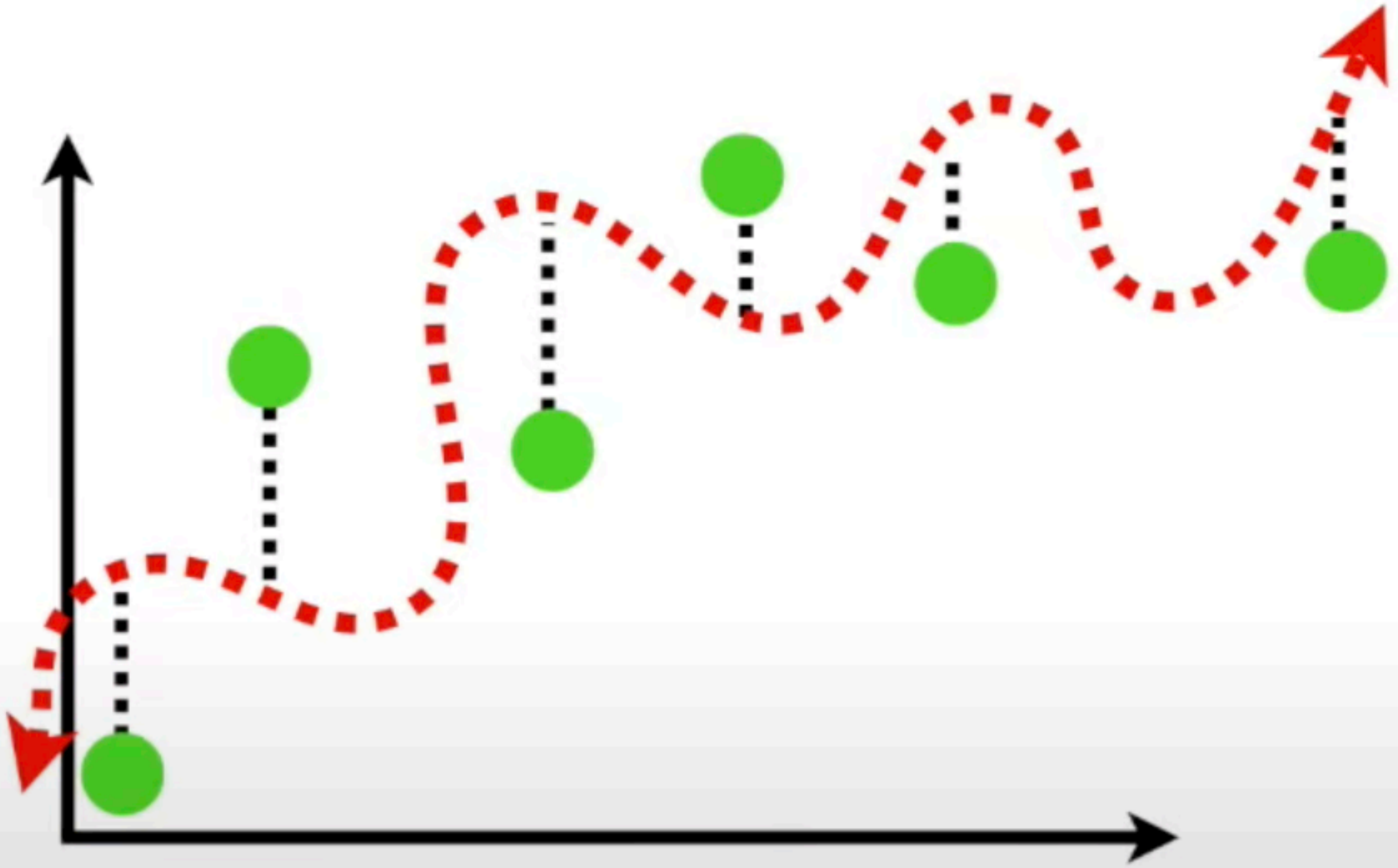
Poorly fit test data



Squiggly

Difference in fits between datasets is **VARIANCE**



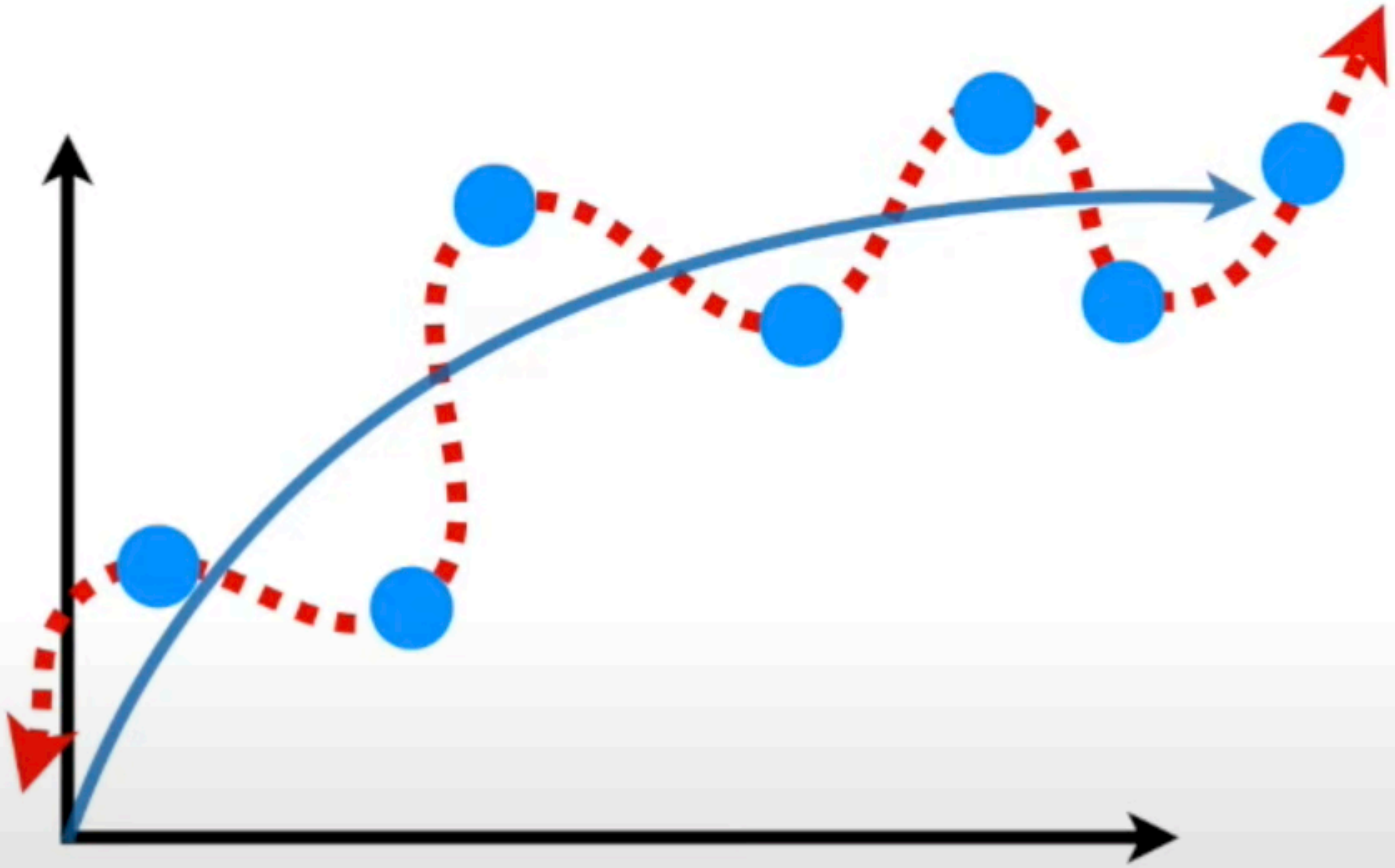


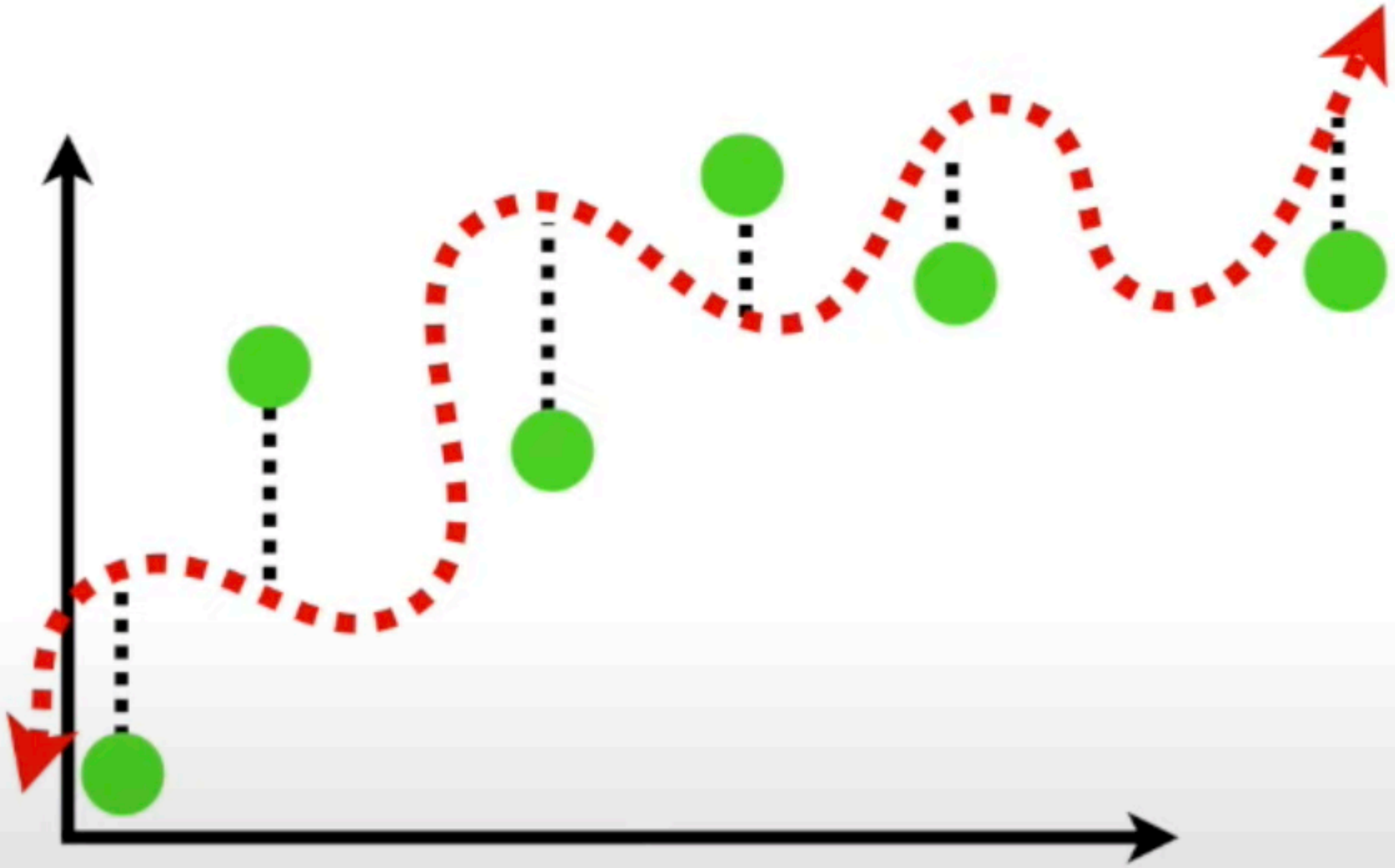
Squiggly Line: Low Bias, High Variance

Low bias: flexible, adapt to curve in relationship between weight & height

High variance: different sums of squares for different datasets

(hard to predict performance with future datasets)





Straight Line: High Bias, Low Variance

High bias: can’t capture curve in relationship between weight & height

Low variance: similar sums of squares for different datasets
(good not great performance, but consistently good)

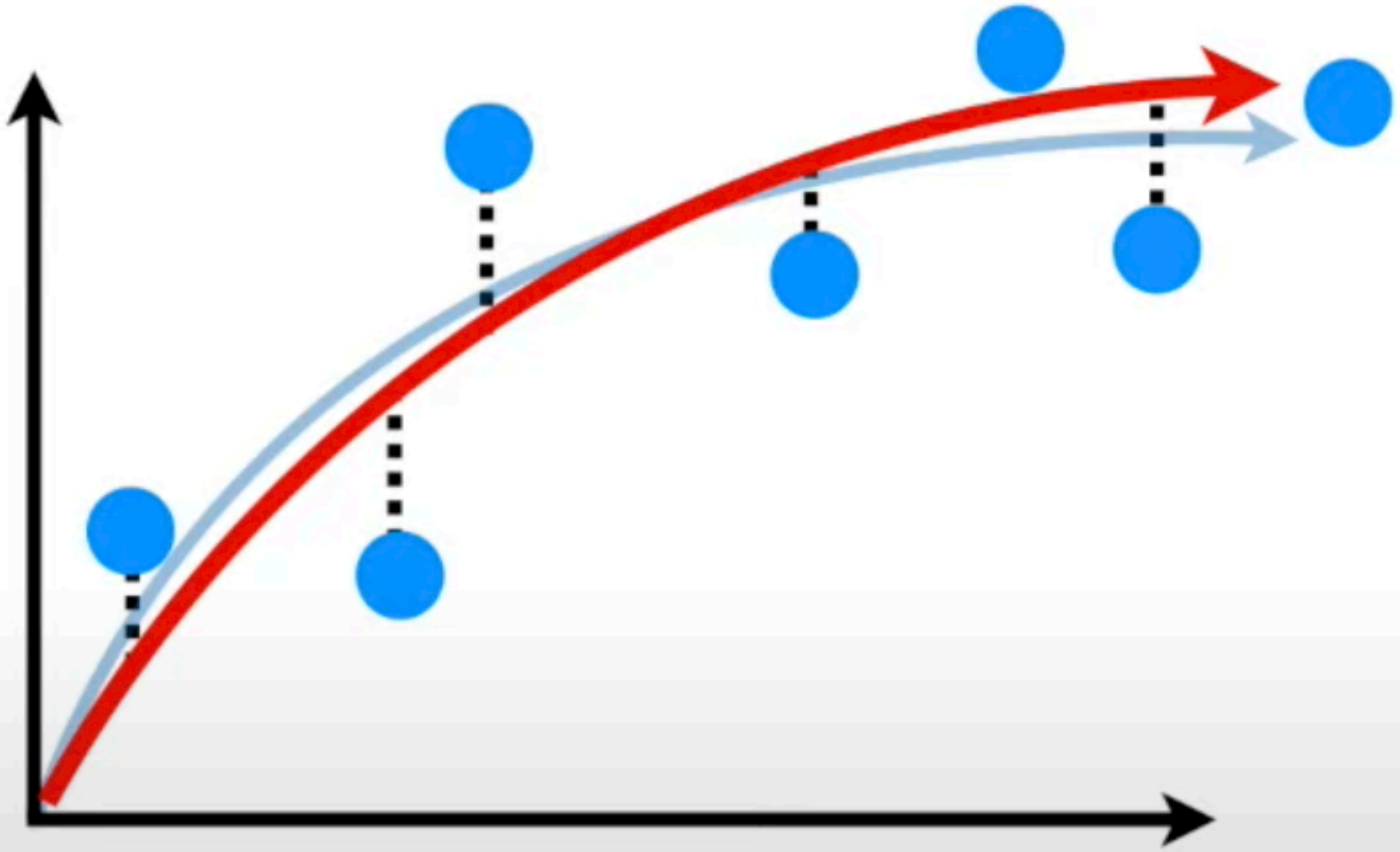
Squiggly: Overfit

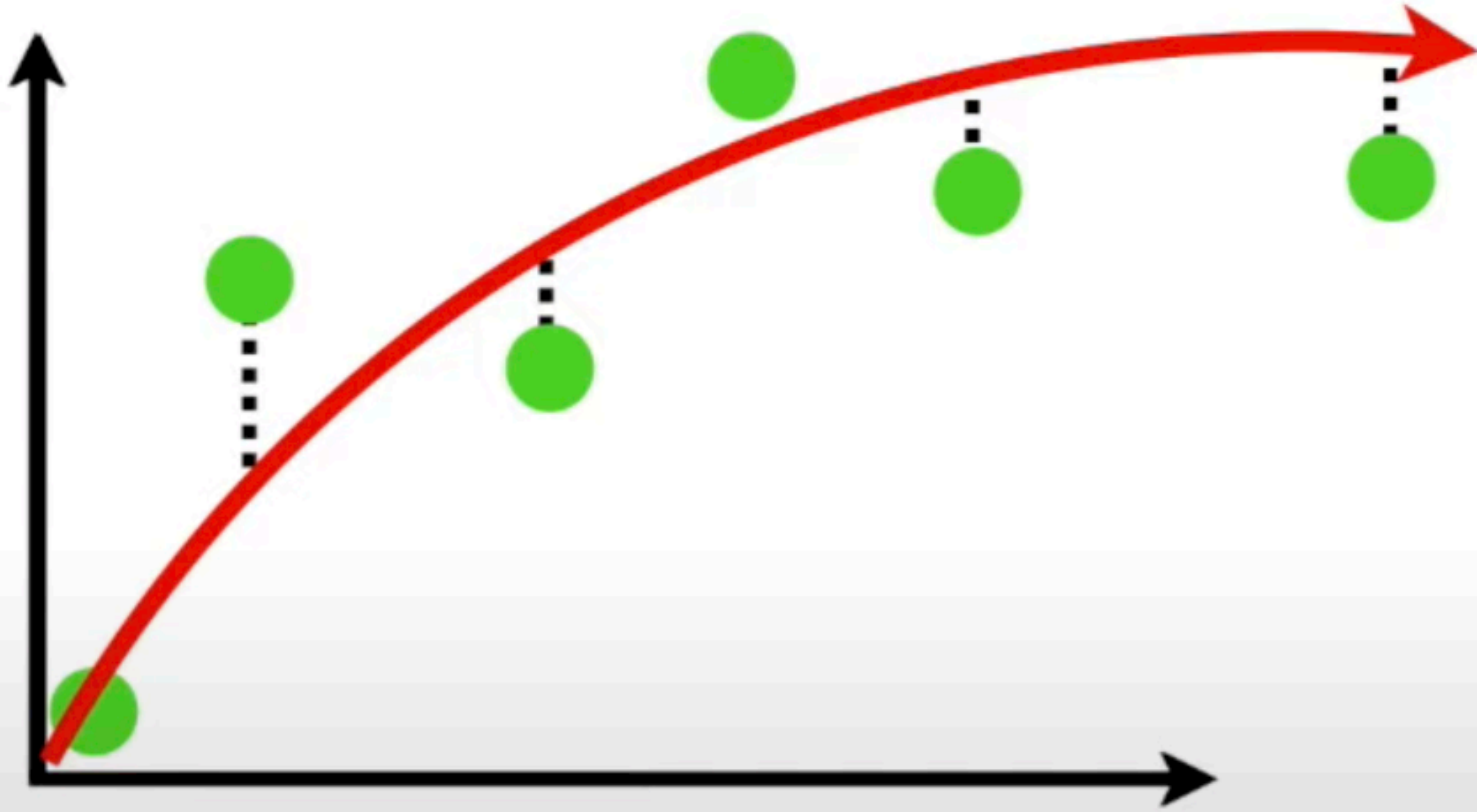
Fits the train set really well, but not the test set

Ideal Model

Low bias: can accurately capture true relationship

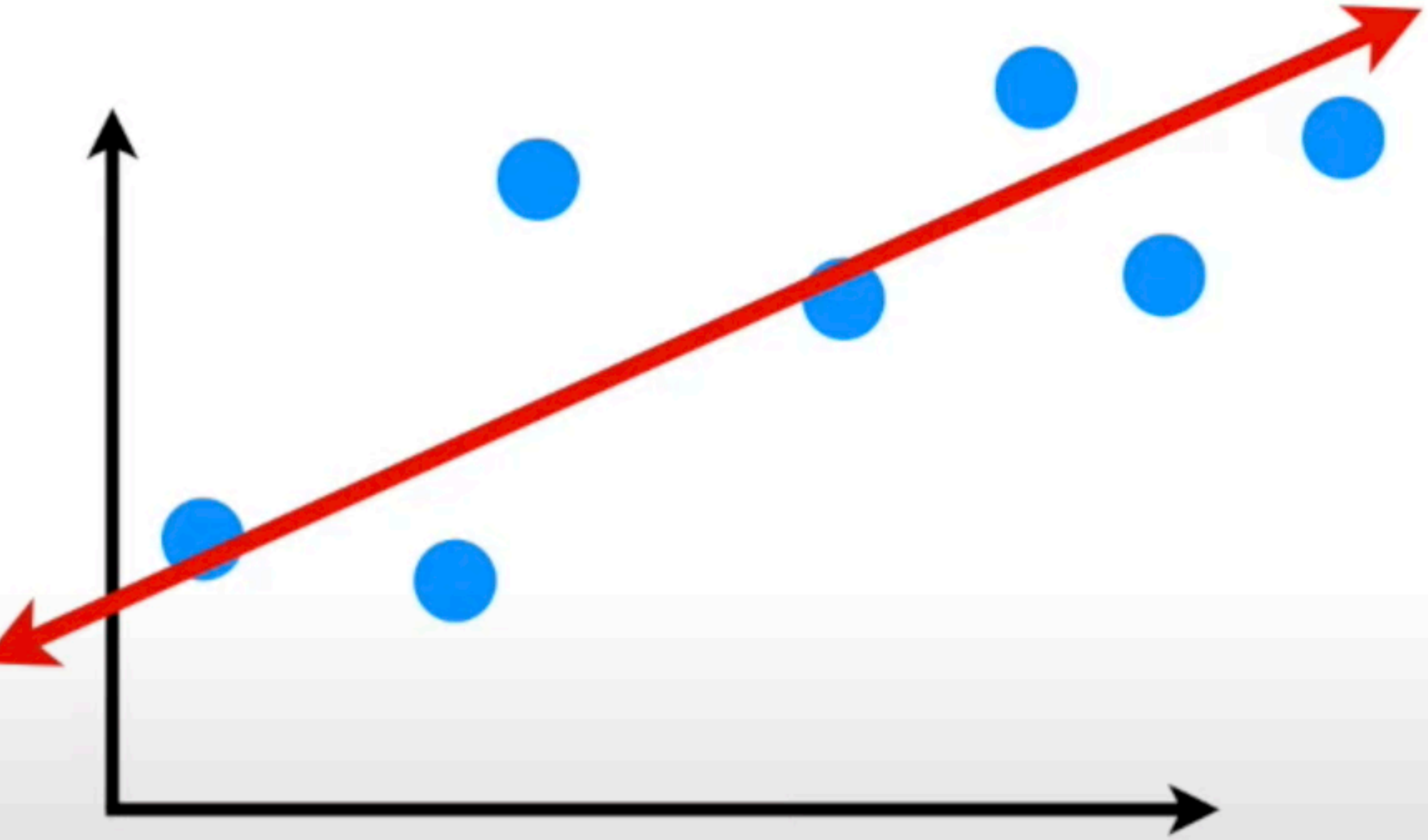
Low variance: consistent predictions across different datasets

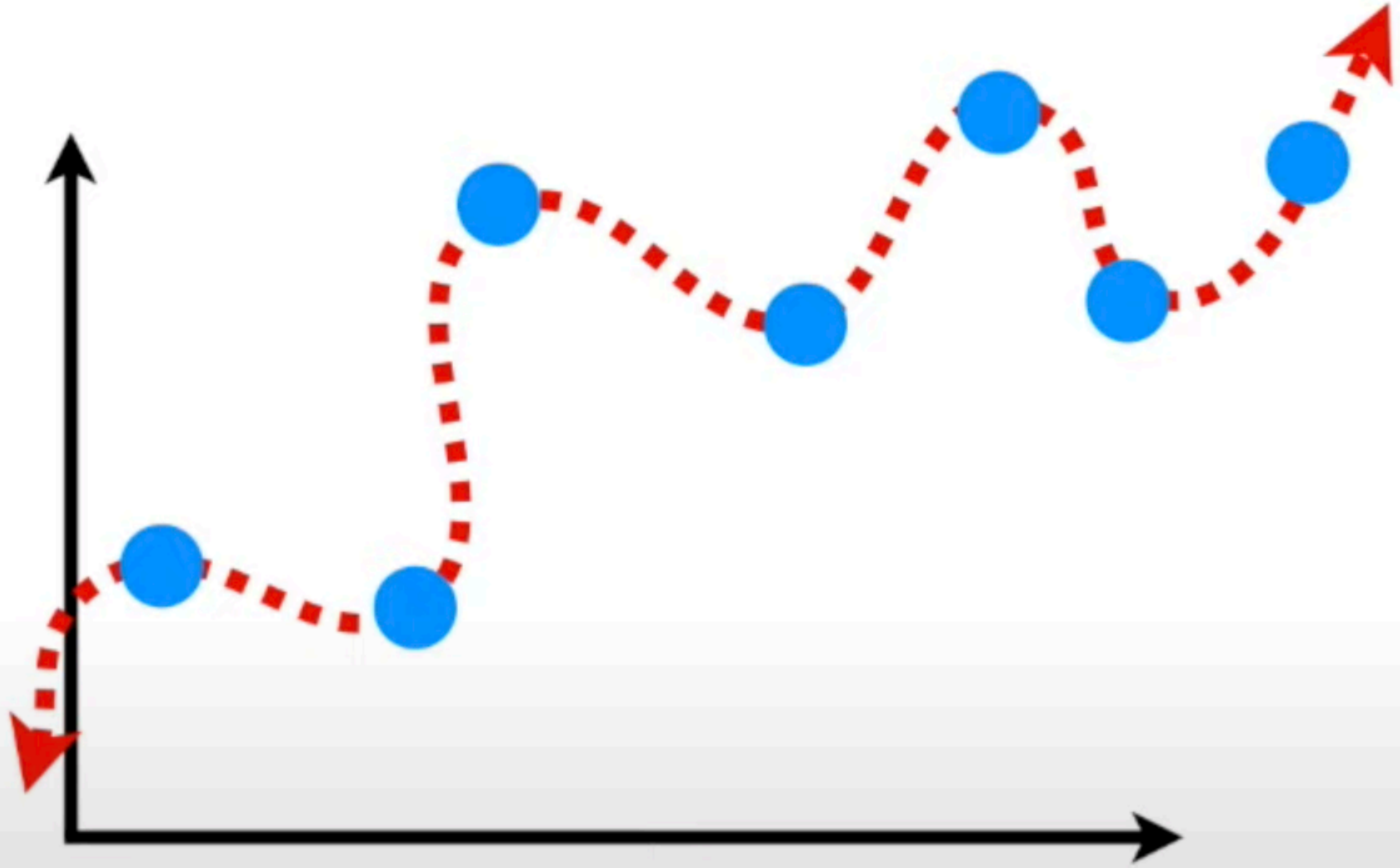




Ideal Model

Find sweet spot between a simple & a complex model





Probably Approximately Correct (PAC)

What is PAC?

- A framework for efficient & accurate computation of complex algorithms dealing with large & uncertain datasets

Perfect vs. approximate fit

- ML goal: find model that perfectly fits training data
  Learn true model from finite training data (drawn from a possibly infinite data)
- Real-world: not always possible to find perfect model due to data complexity & noise
- PAC learning addresses this by approximation

Perfect vs. approximate fit

- Instead of finding a perfect model
- PAC learning goal: Find model that is probably approximately correct
- Training: Examine data, learn an approximate model
- Goal: Learn approximation A that is very close to the true model that generated the data

Perfect vs. approximate fit

- PAC: With low probability <= p (0.001), ||concept - approximate|| <= e (3% increase in error rate)
- How: Limit our concept space
- Role of training data: falsify bad ideas, we need a lot of training data to eliminate bad ideas
- Limit concept space so there are only so many bad ideas to eliminate

More formally

- PAC meaning
  — Possible to make accurate predictions on unseen data within a certain error margin

- **Sample complexity**
  — The amount of data needed to learn a good approximate model

- PAC learning goal: find models with low sample complexity
  (models that can learn from a smaller amount of data)

Power of PAC Learning

- The concepts of **approximation** and **sample complexity** are essential for understanding the limitations and potential of ML algorithms

- PAC learning
  — Considers inherent uncertainty & variability in real-world data
  — Allows for more realistic & efficient approach to learning from data
  — Allows for more robust & efficient models that perform well in real-world

Next Lectures

- Monday is a holiday!

- Lecture 5:
  — Design & implementation of ML methods
  — Using ML toolboxes & platforms

- Lecture 6:
  — VC: A numeric representation of how limited our concept space is for PAC learning