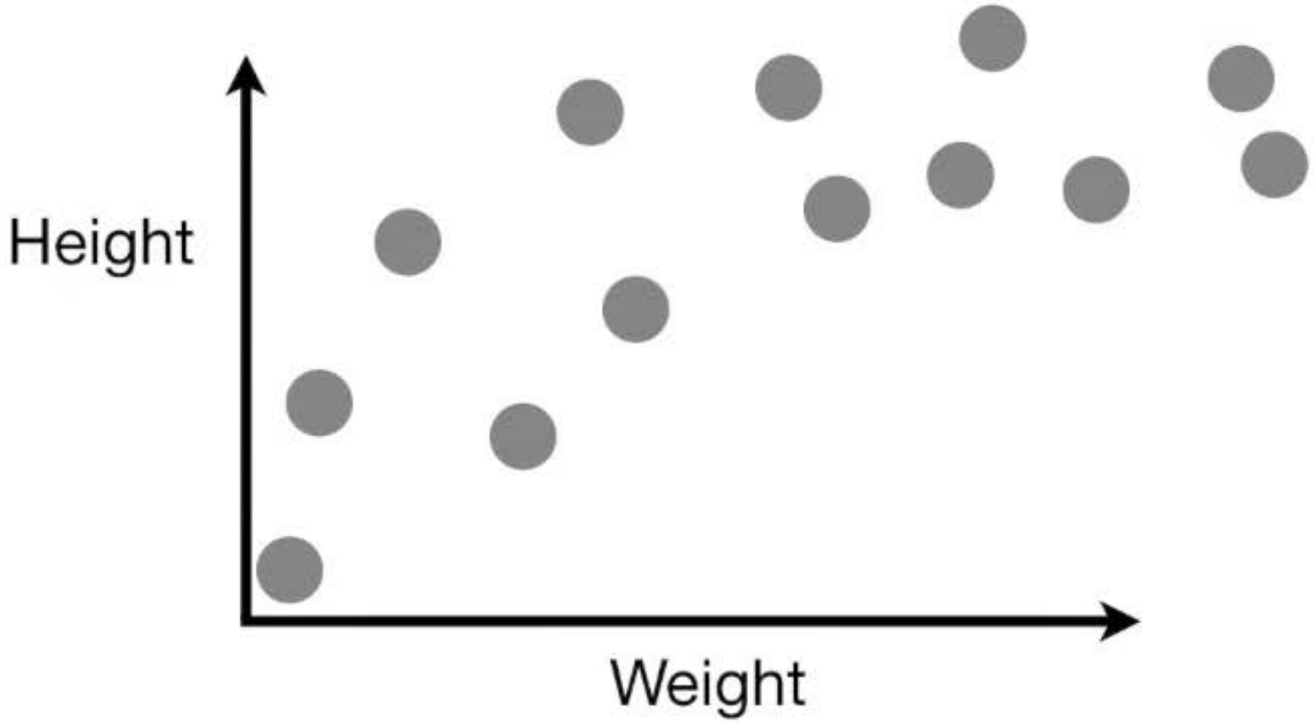
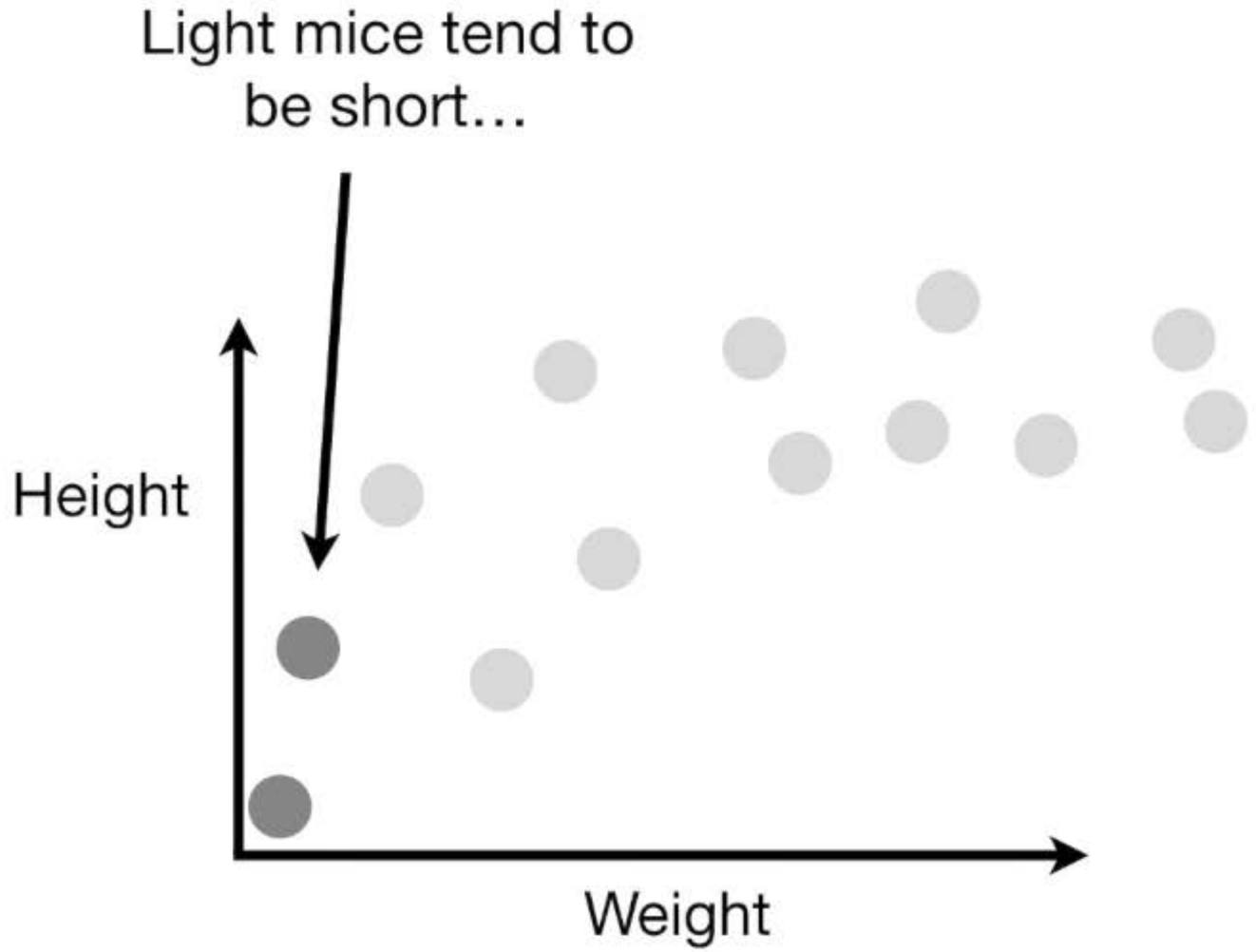
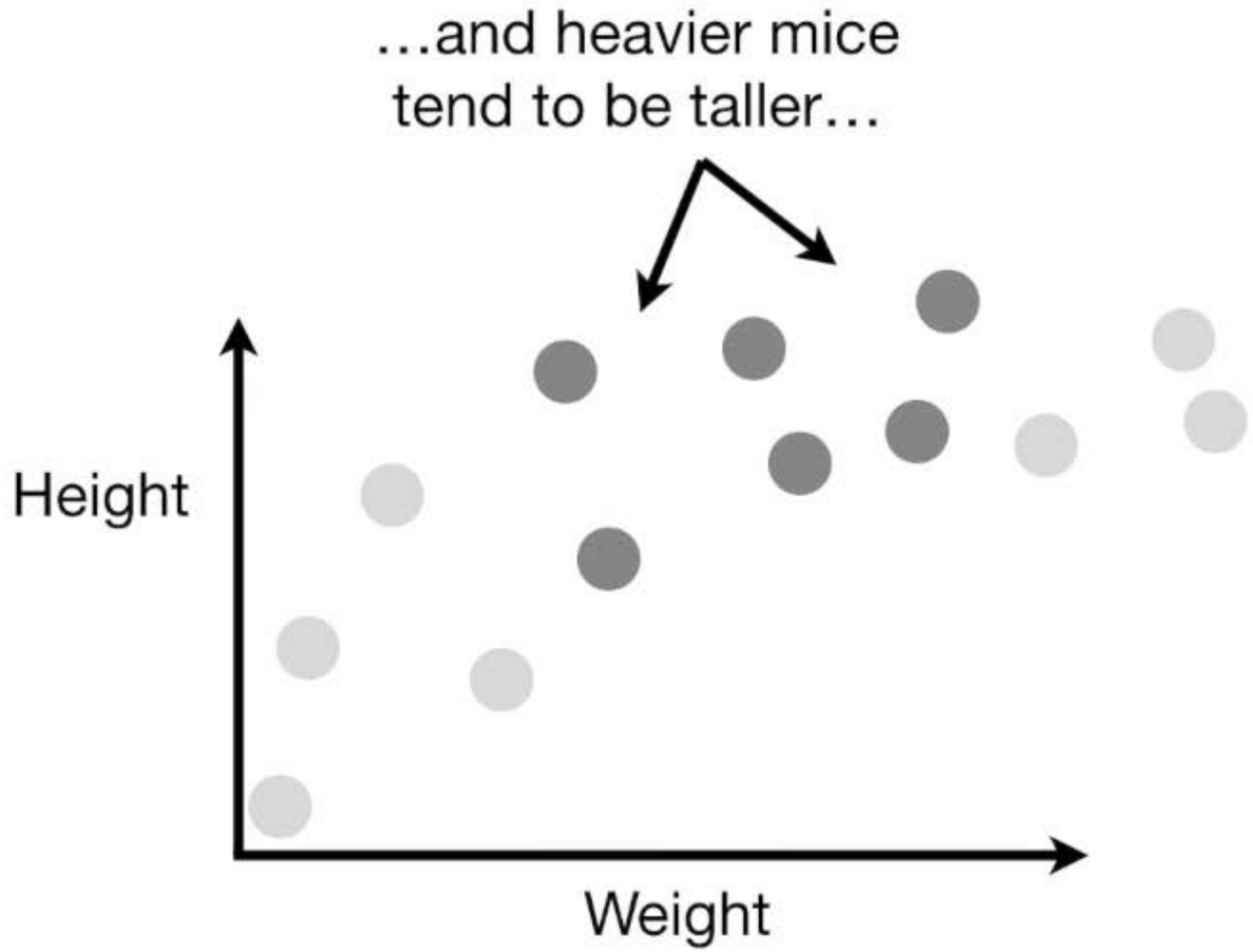
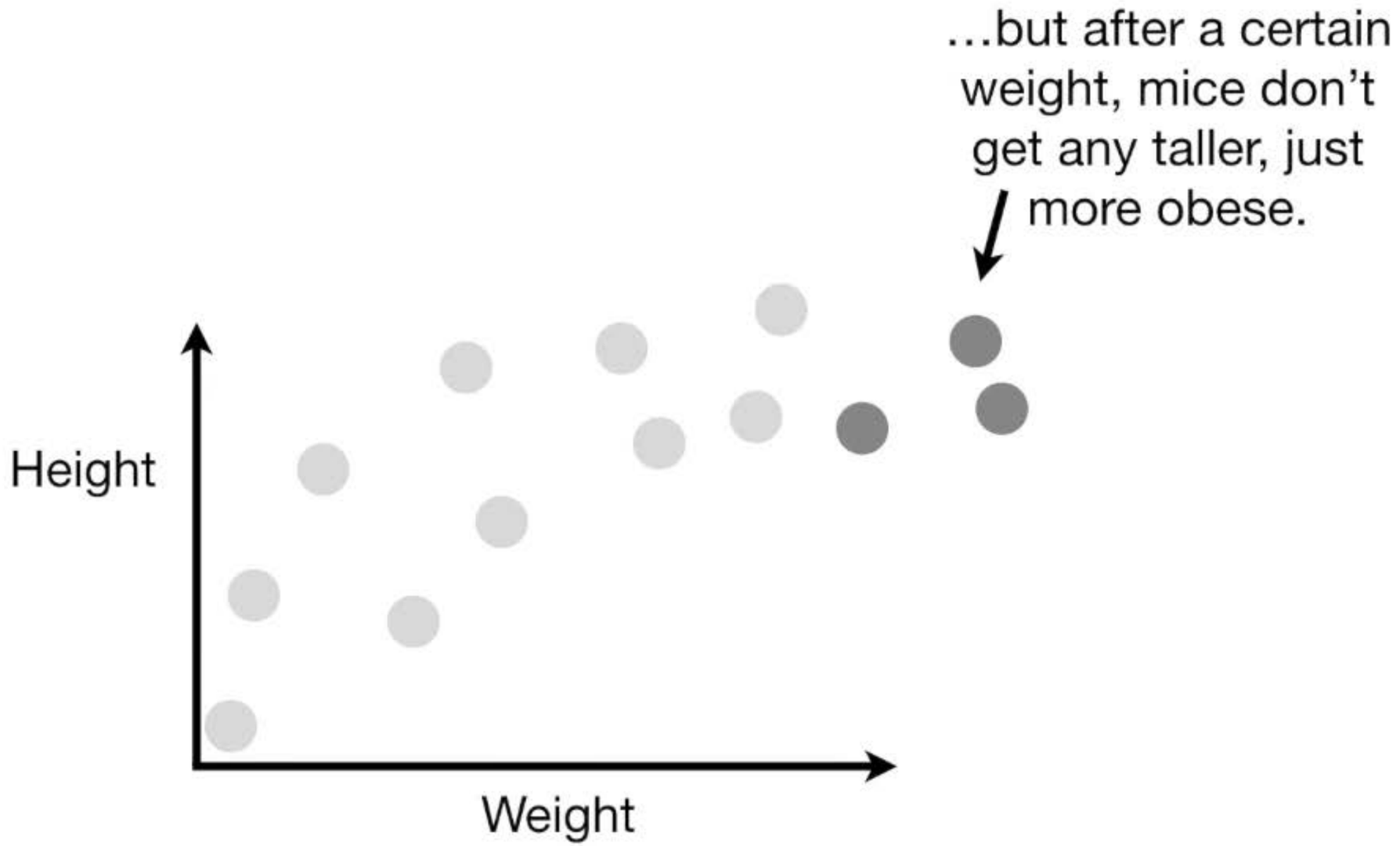
Imagine we measured the weight and height of a bunch of mice and plotted the data on a graph…

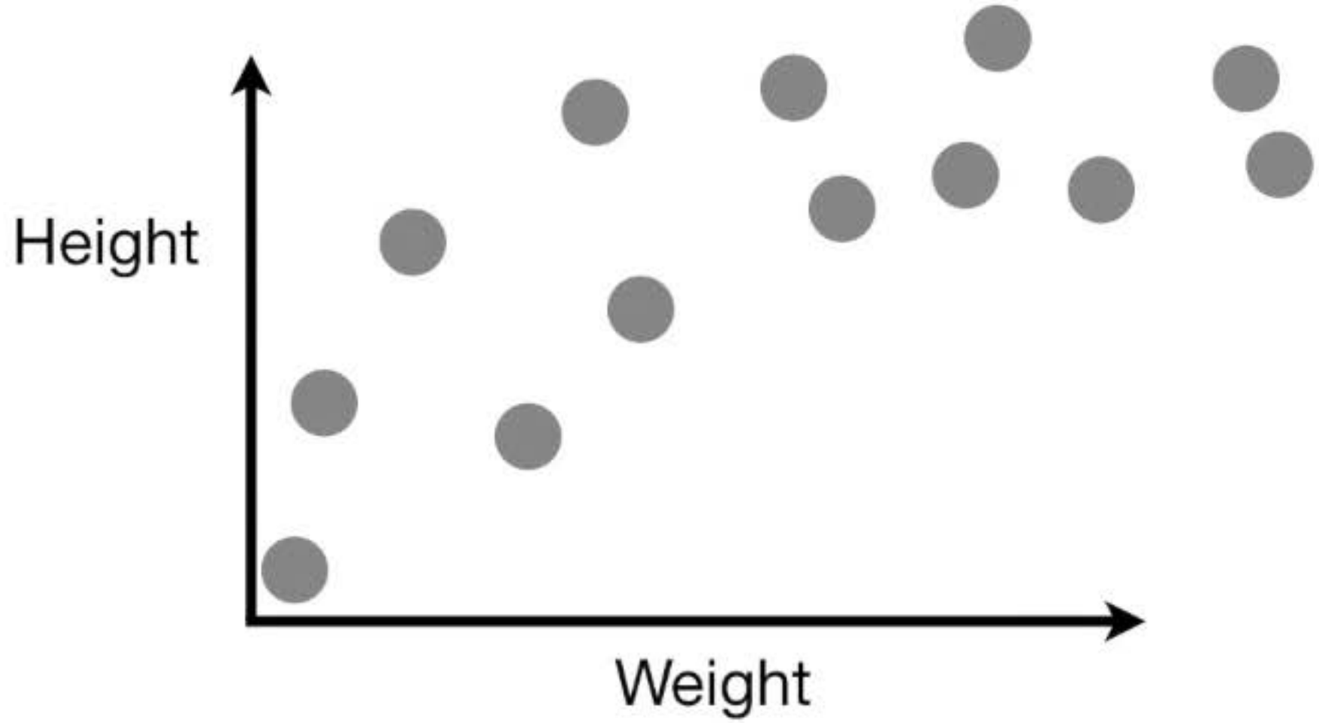


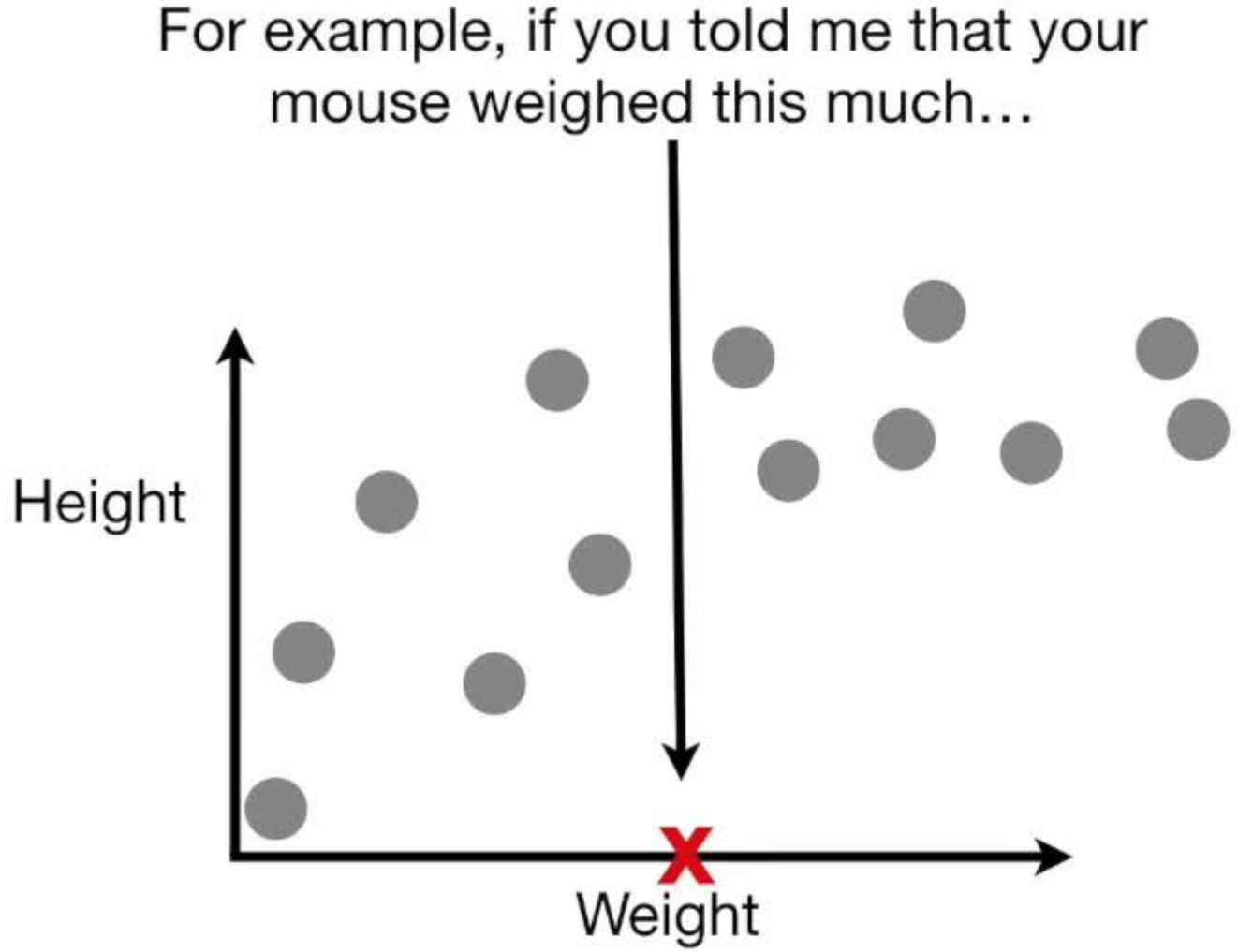

Light mice tend to
be short…
Height
Weight


…and heavier mice
tend to be taller…
Height
Weight

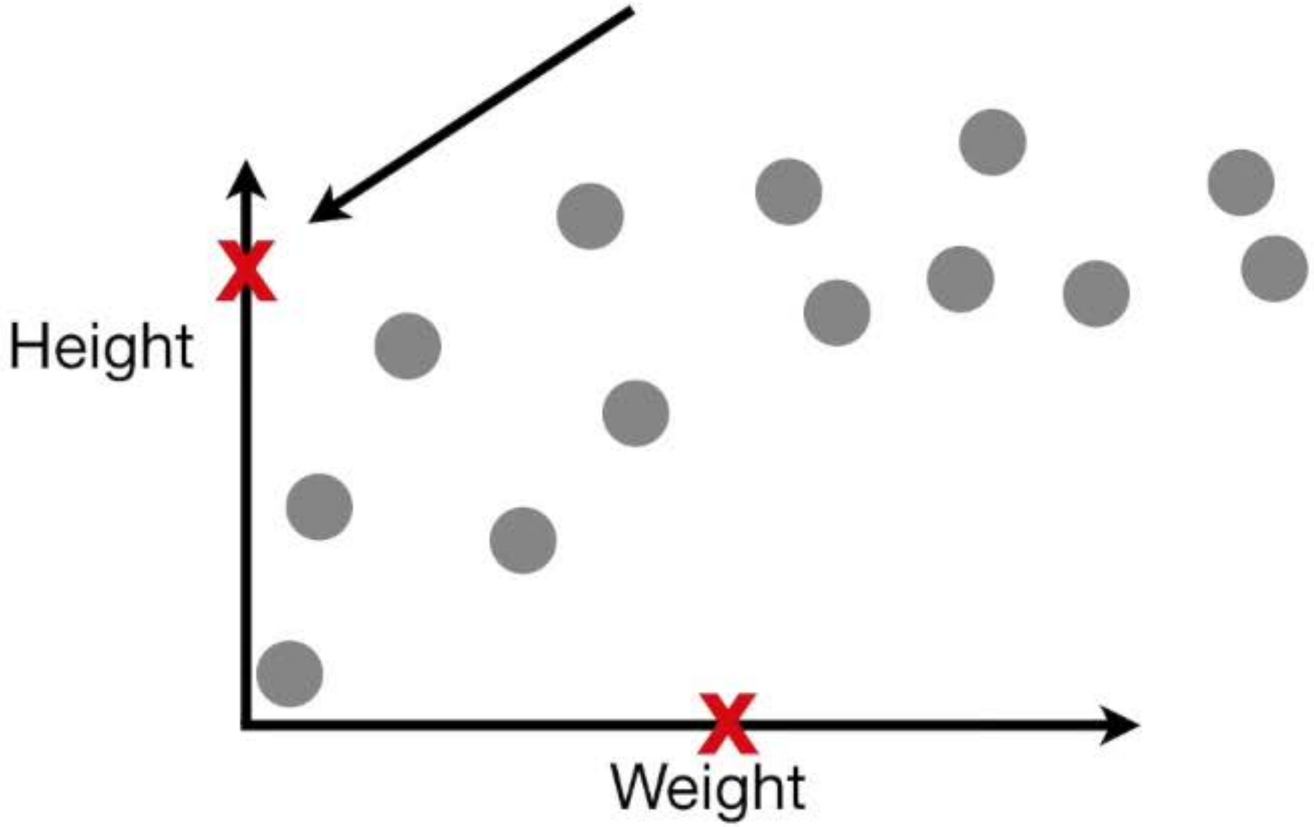

…but after a certain weight, mice don't get any taller, just more obese.
Height
Weight

Given this data, we would like to predict mouse height given its weight.

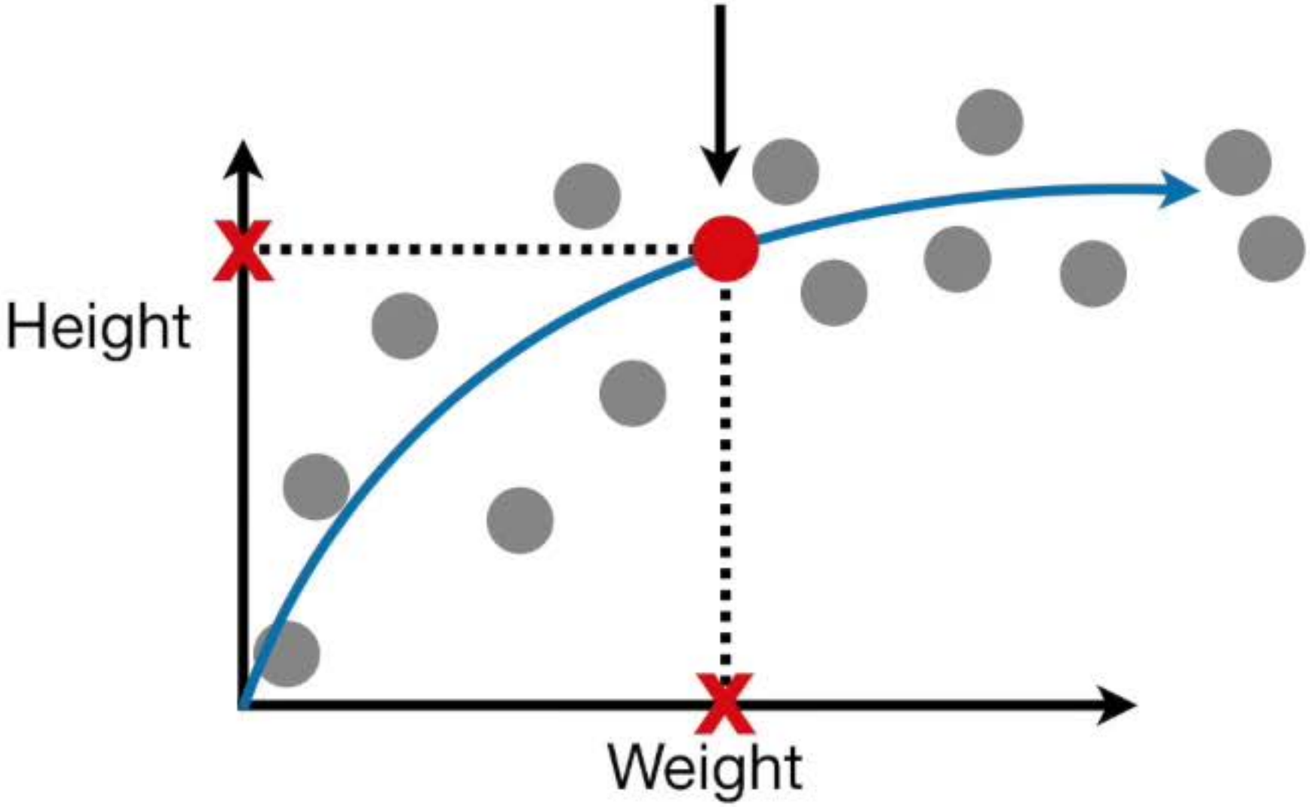



For example, if you told me that your mouse weighed this much…
Height
Weight

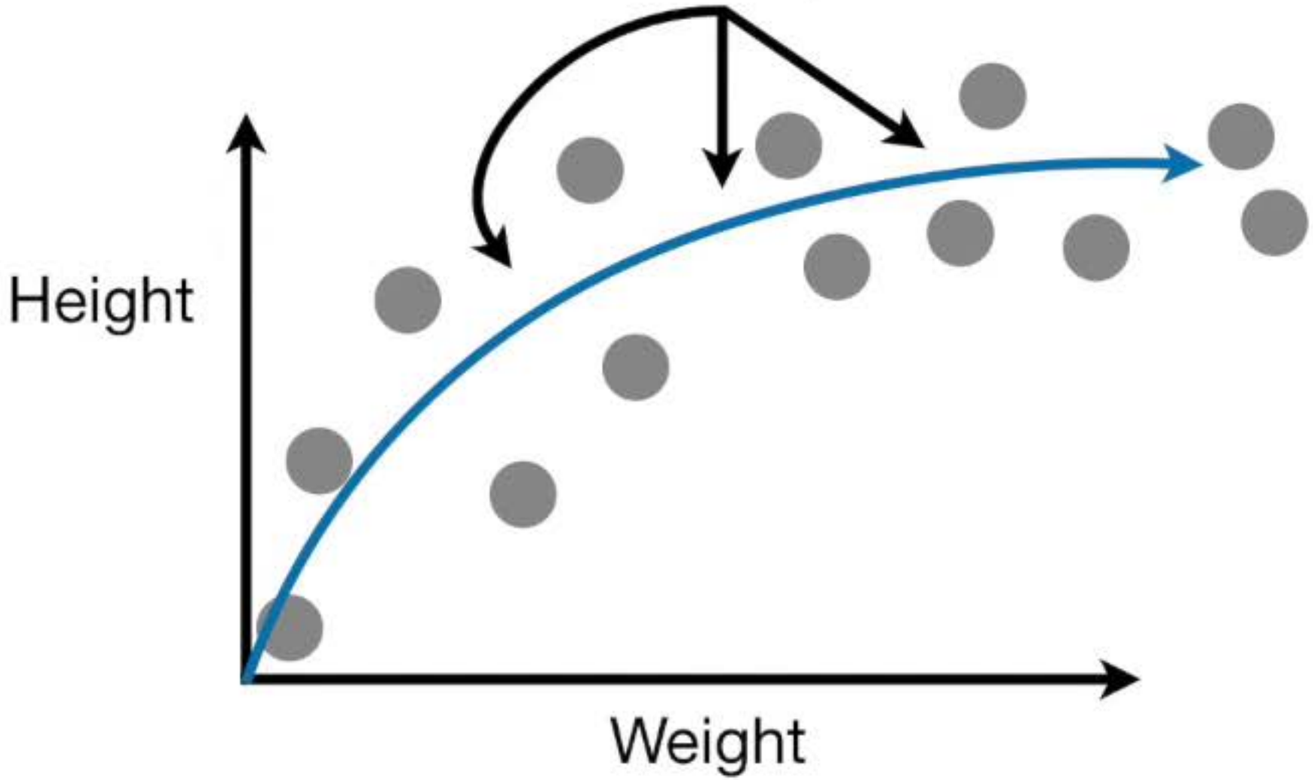
…then we might predict that the mouse is this tall….



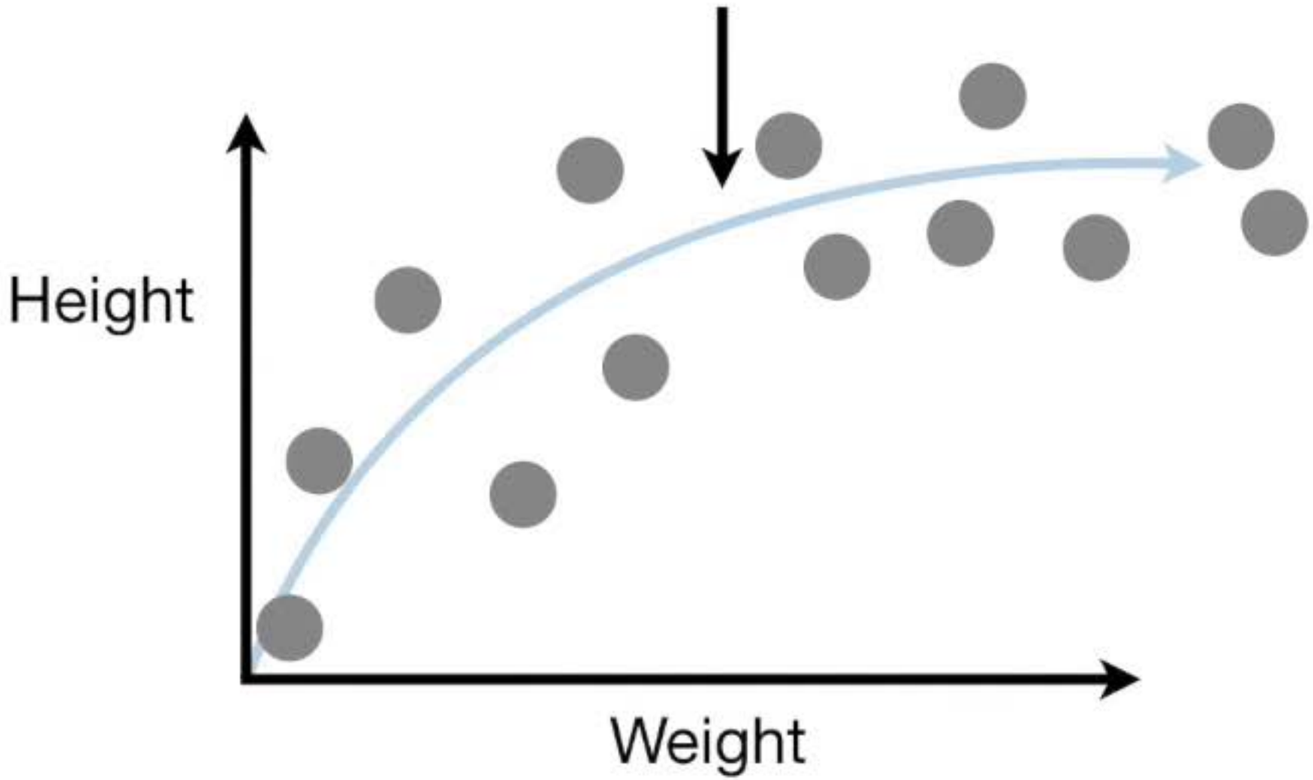
Ideally, we would know the exact mathematical formula that describes the relationship between weight and height…



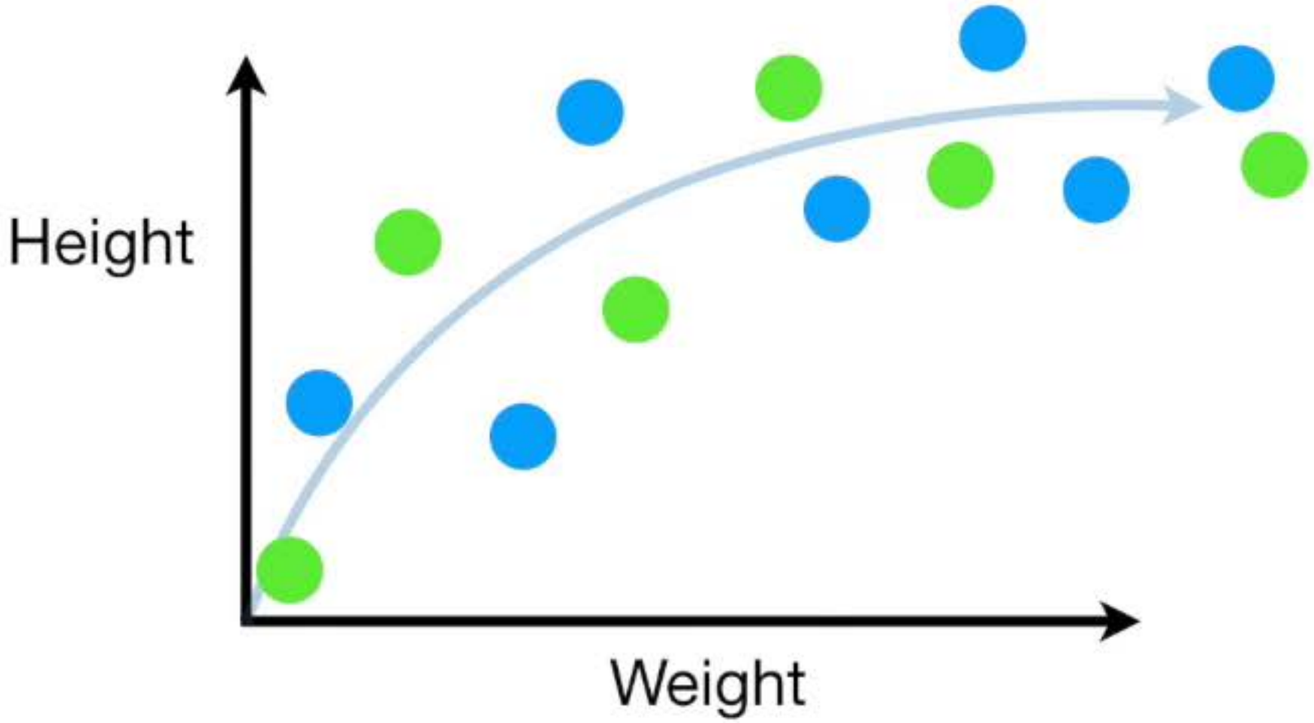
…but, in this case, we don't know the formula, so we're going to use two machine learning methods to approximate this relationship.



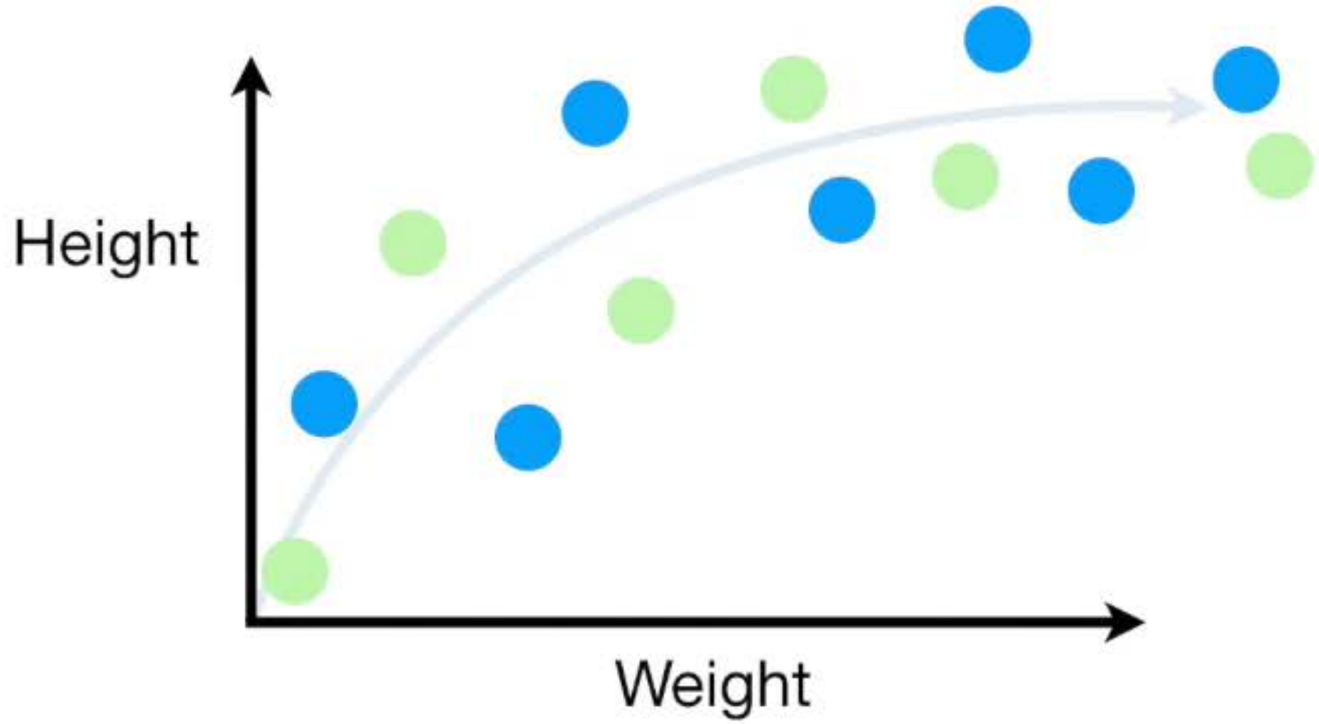
However, I'll leave the "true" relationship curve in the figure for reference.



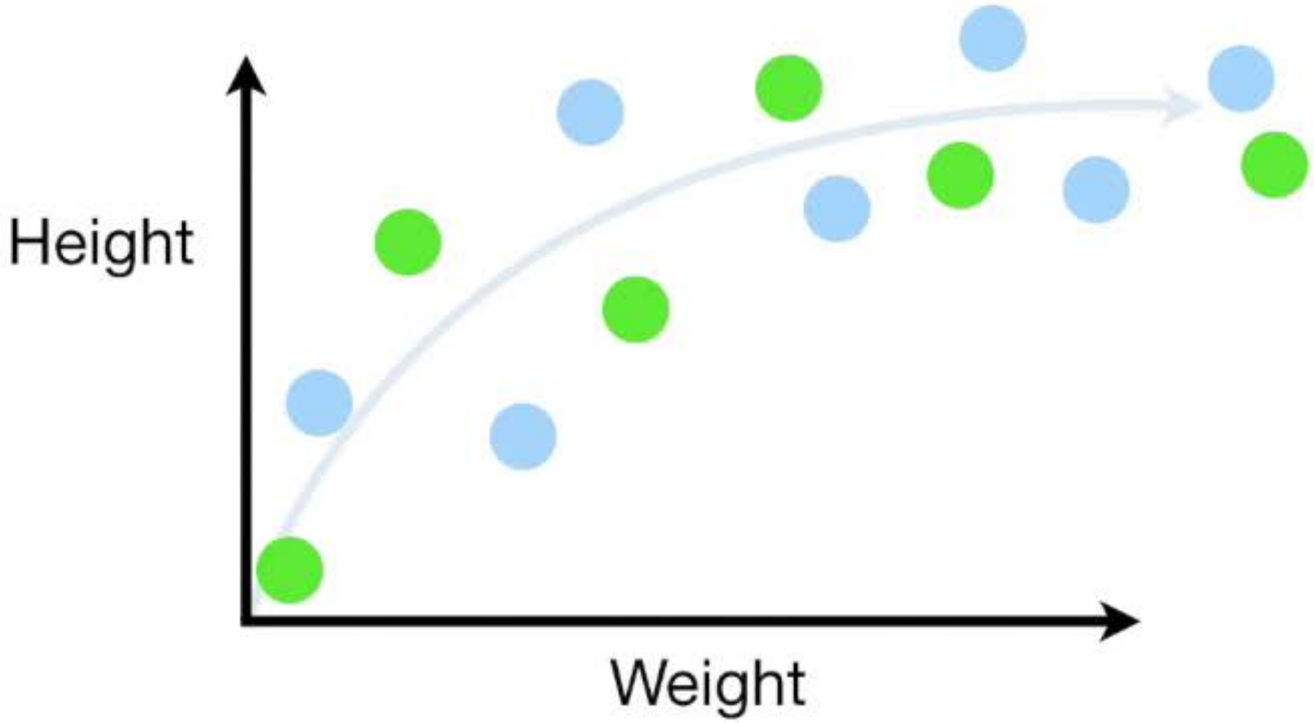
The first thing we do is split the data into two sets, one for training the machine learning algorithms and one for testing them.



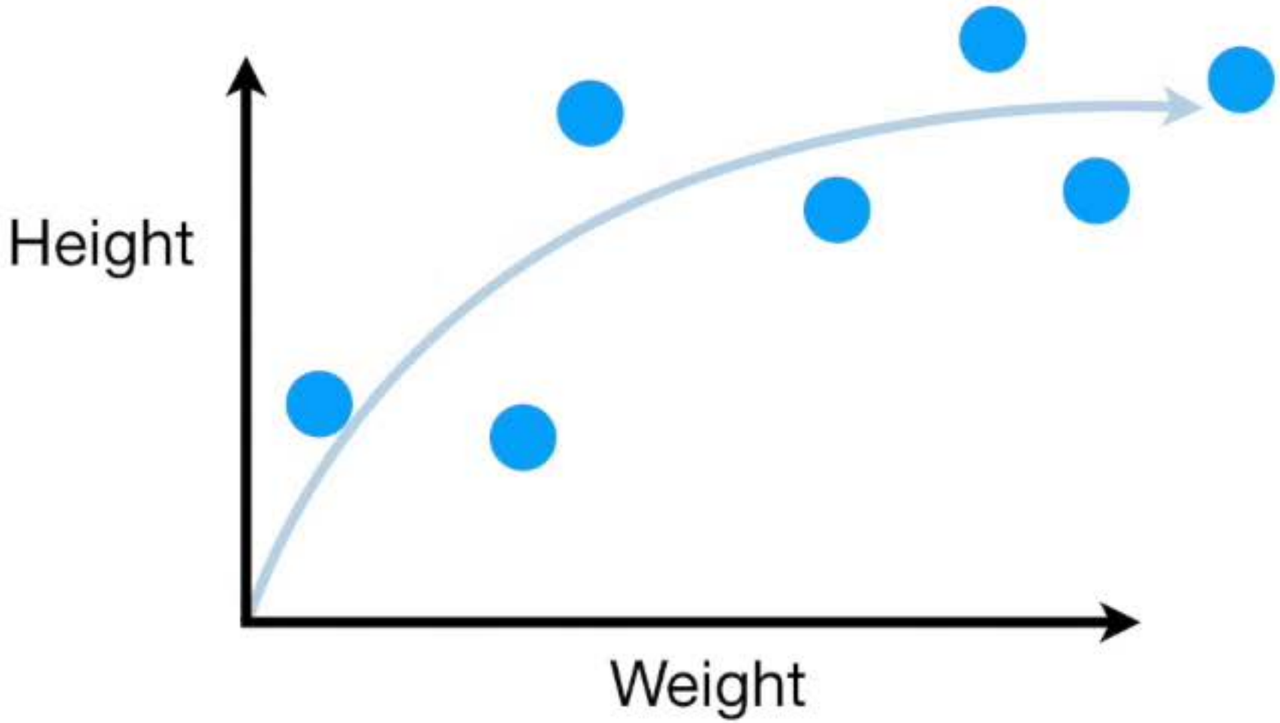
The **Blue Dots** are the **training set**…



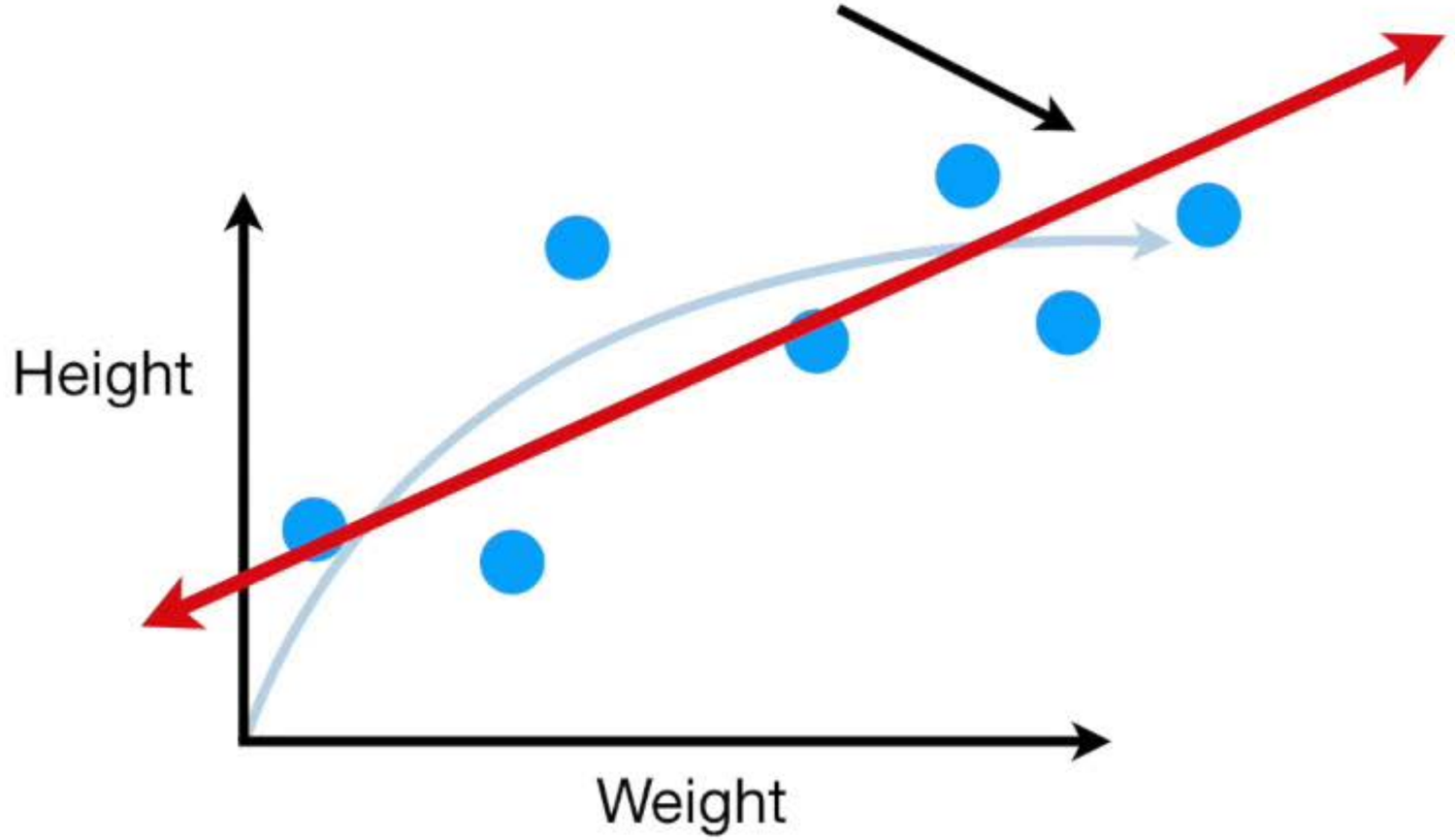
…and the **Green Dots** are the **testing set**.



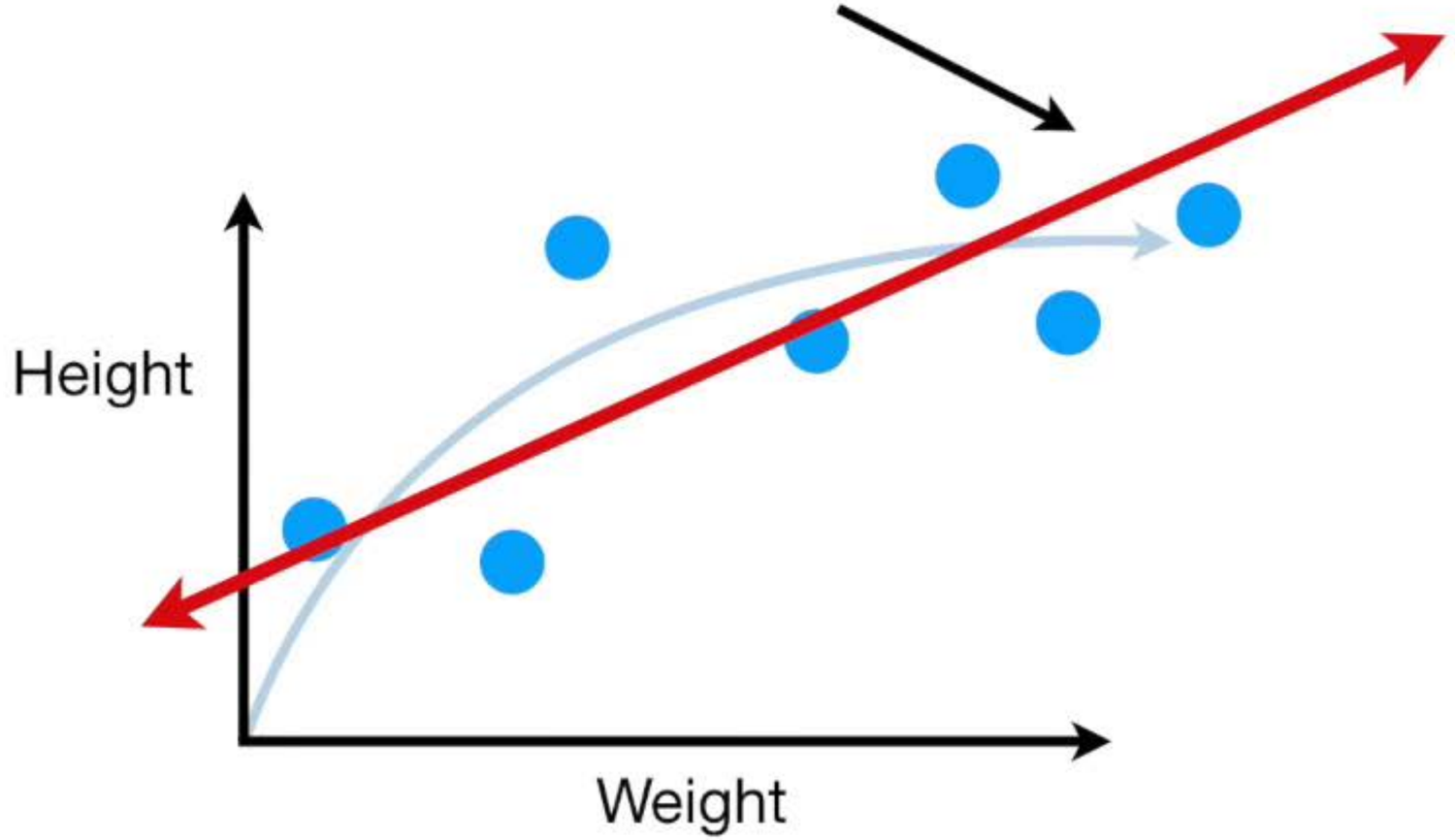
Here is just the **training set**…



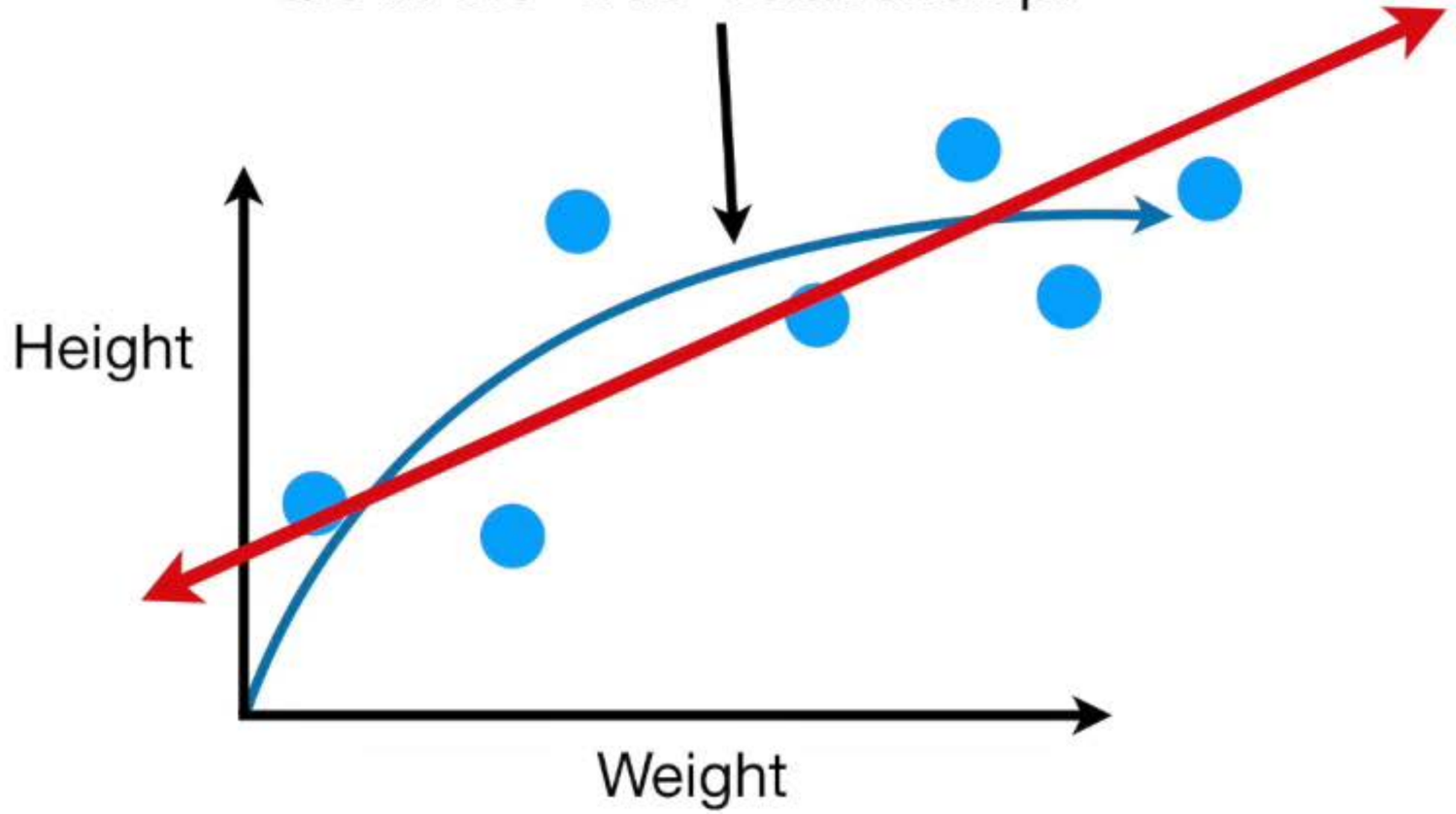
The first machine learning algorithm
that we will use is Linear Regression
(aka “Least Squares”).



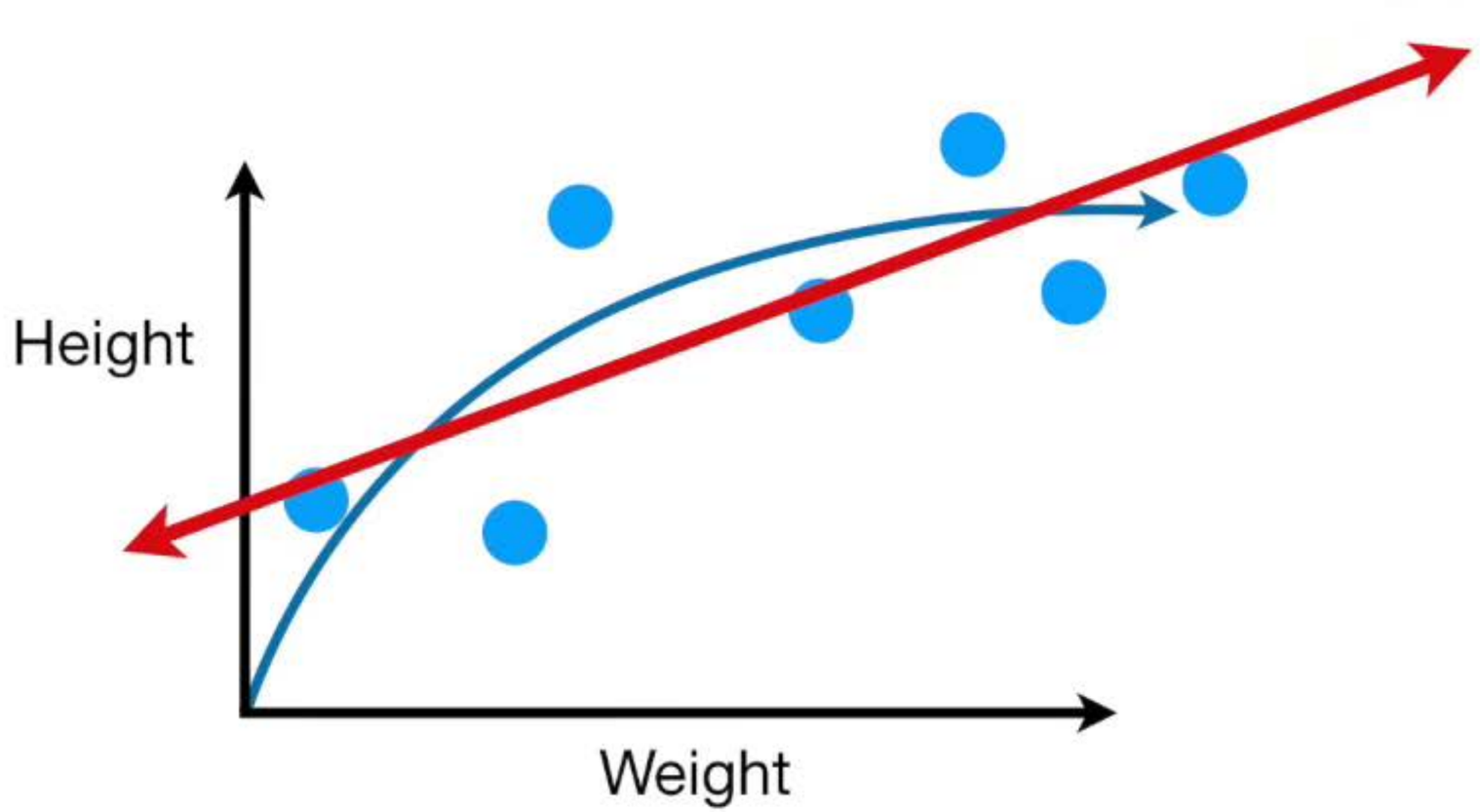
Linear regression fits a **Straight Line** to the **training set**.



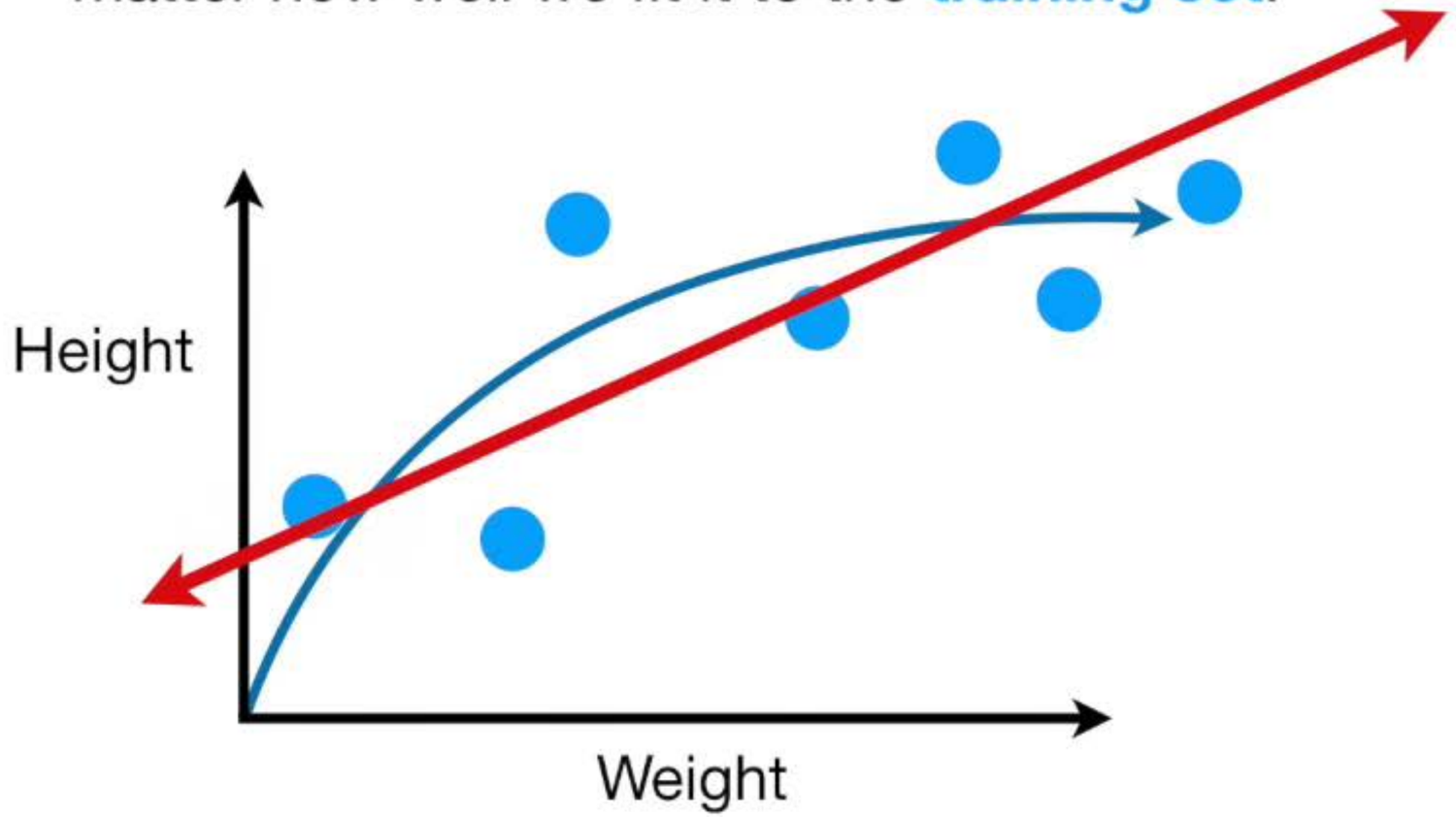
NOTE: The **Straight Line** doesn't have the flexibility to accurately replicate the arc in the "true" relationship.



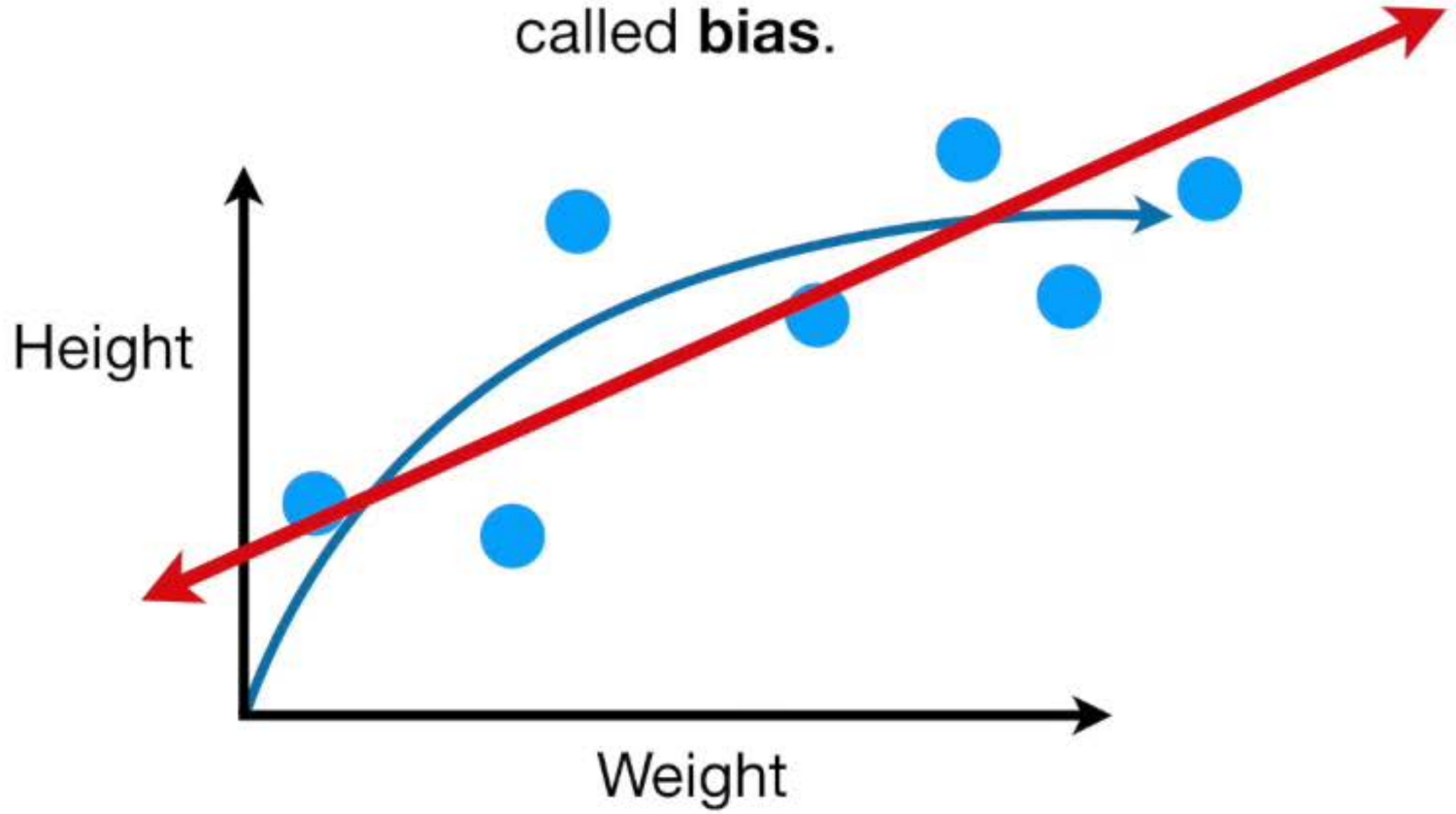
No matter how we try to fit the line, it will never curve…



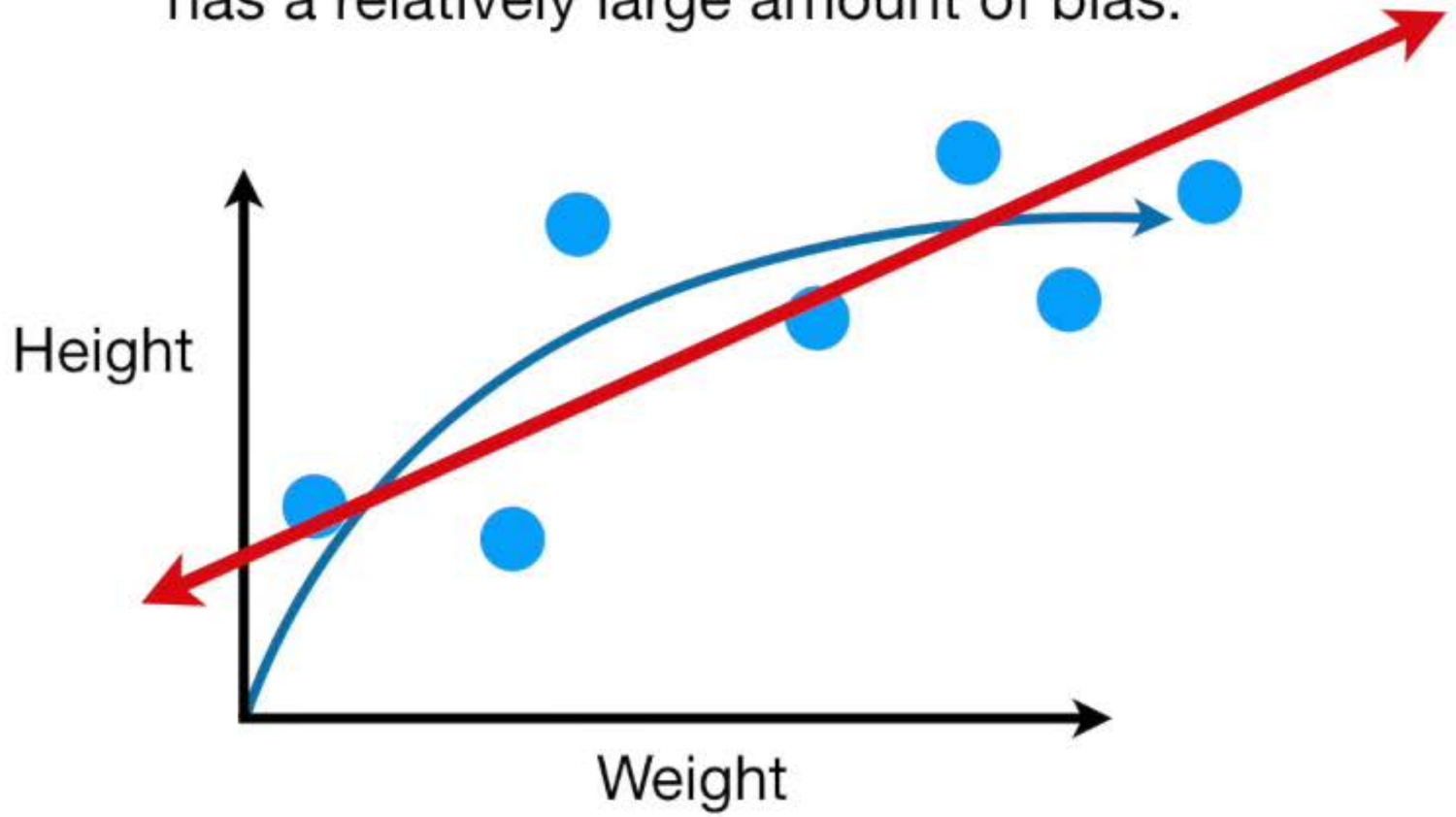
Thus, the **Straight Line** will never capture the true relationship between weight and height, no matter how well we fit it to the **training set**.



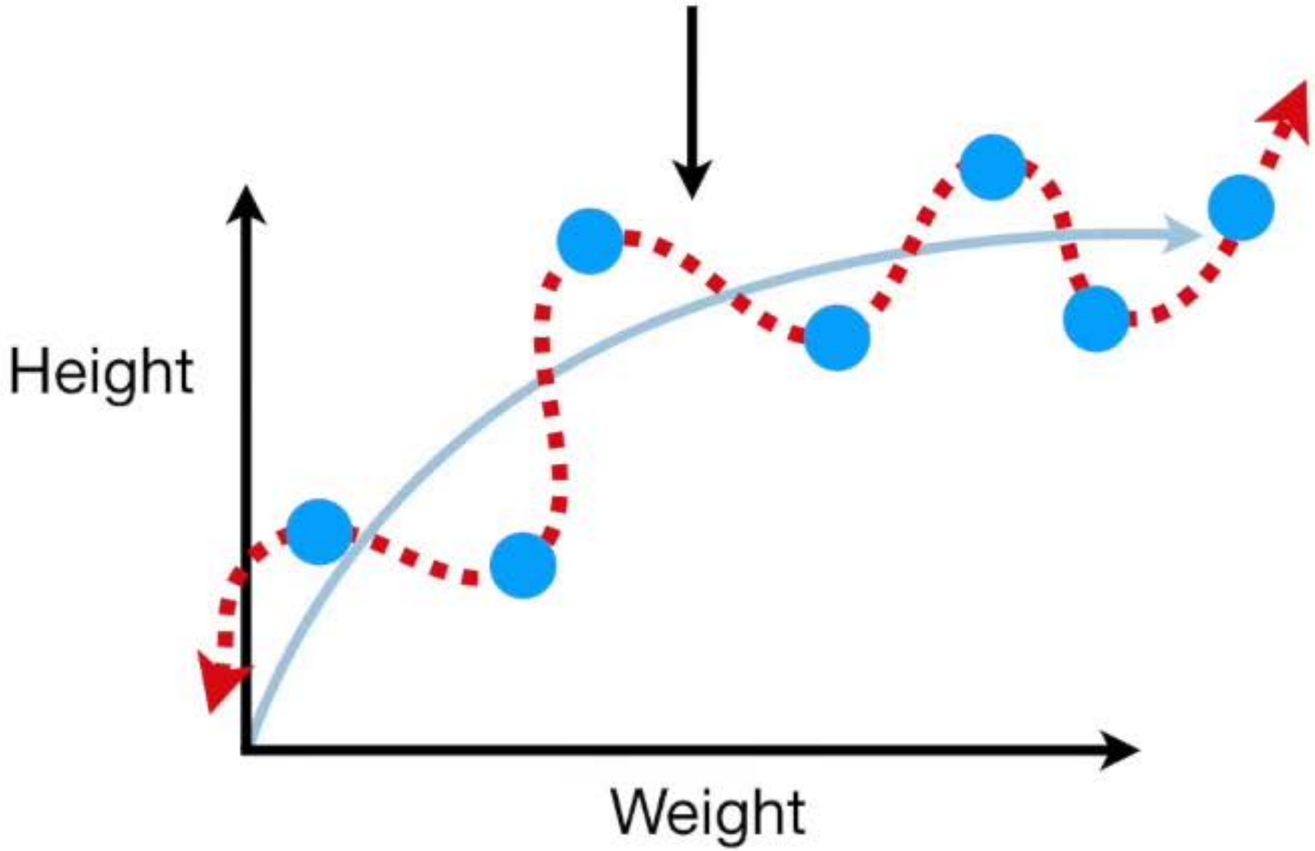
The inability for a machine learning method (like linear regression) to capture the true relationship is called **bias**.



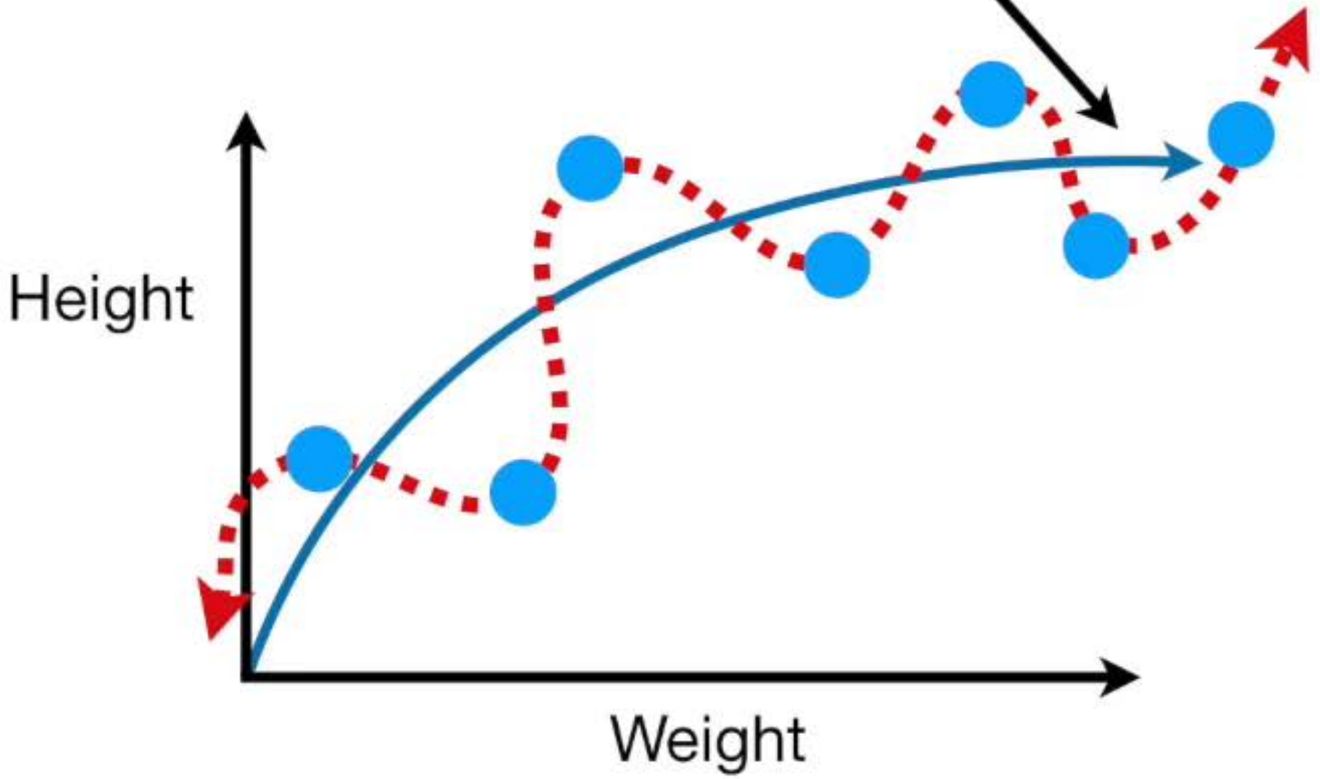
Because the **Straight Line** can't be curved like the "true" relationship, it has a relatively large amount of bias.



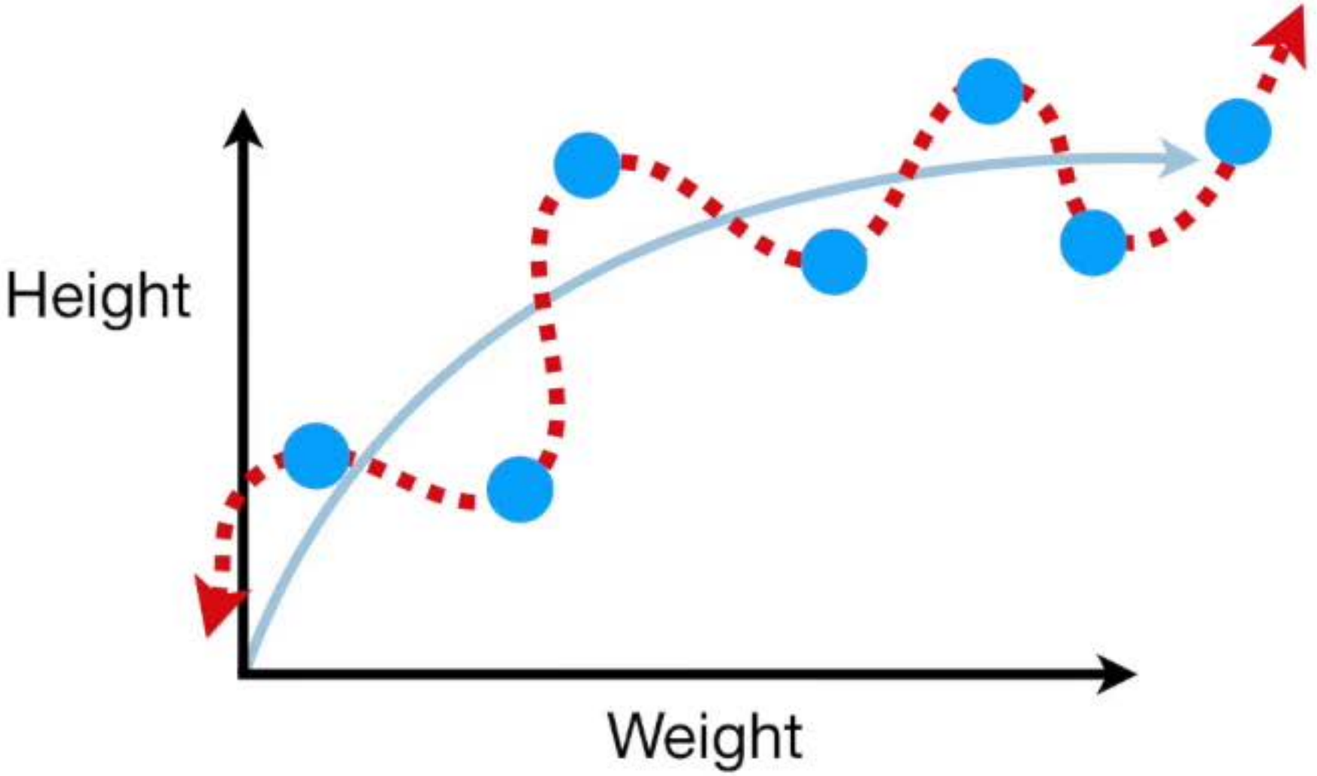
Another machine learning method might fit a **Squiggly Line** to the **training set**…



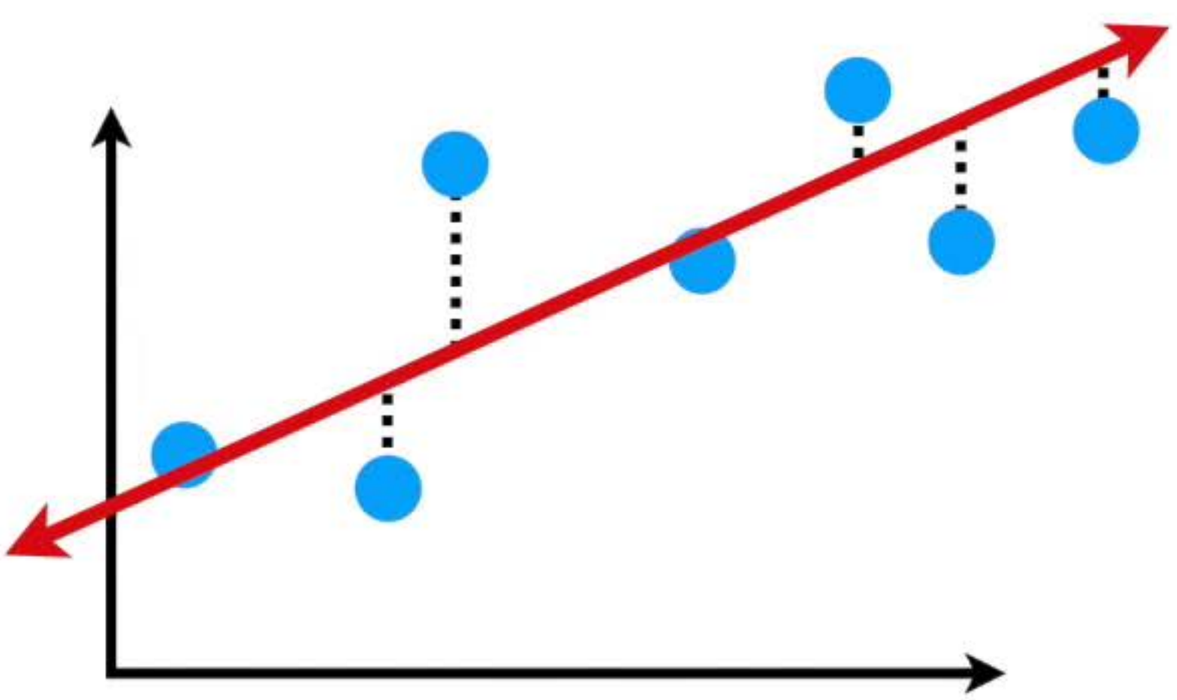
The **Squiggly Line** is super flexible and hugs the **training set** along the arc of the true relationship.

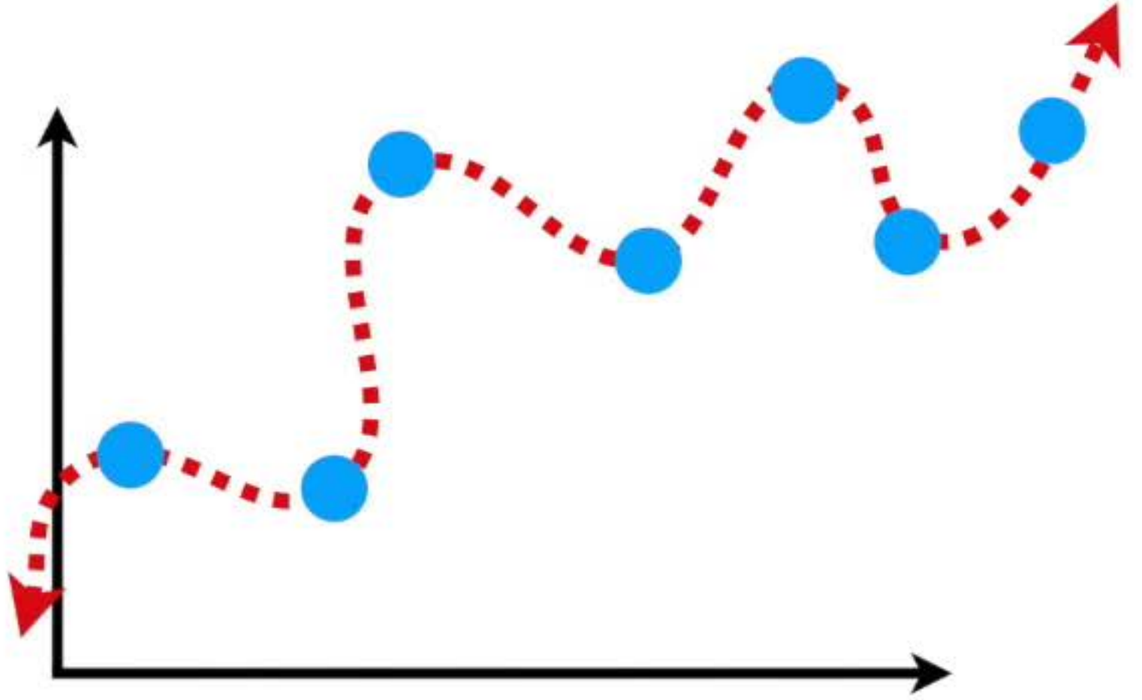


Because the **Squiggly Line** can handle the arc in the true relationship between weight and height, it has very little **bias**.

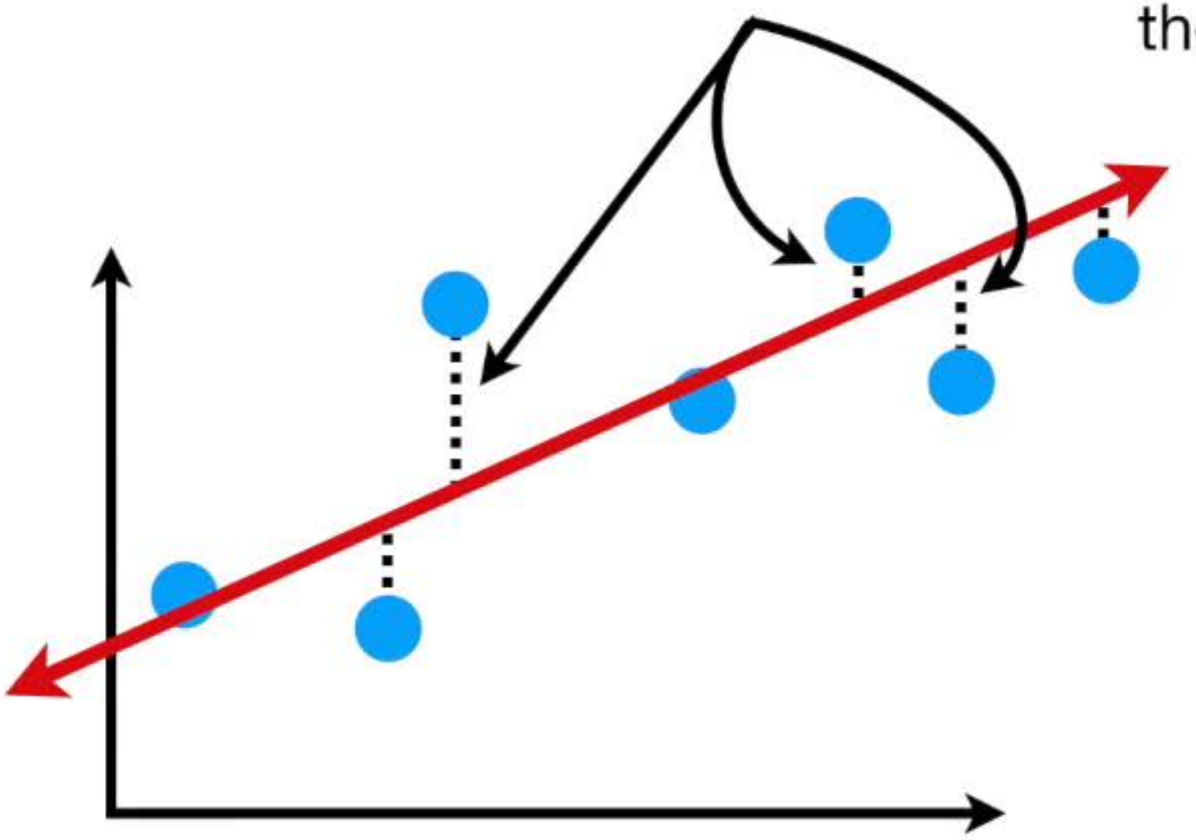


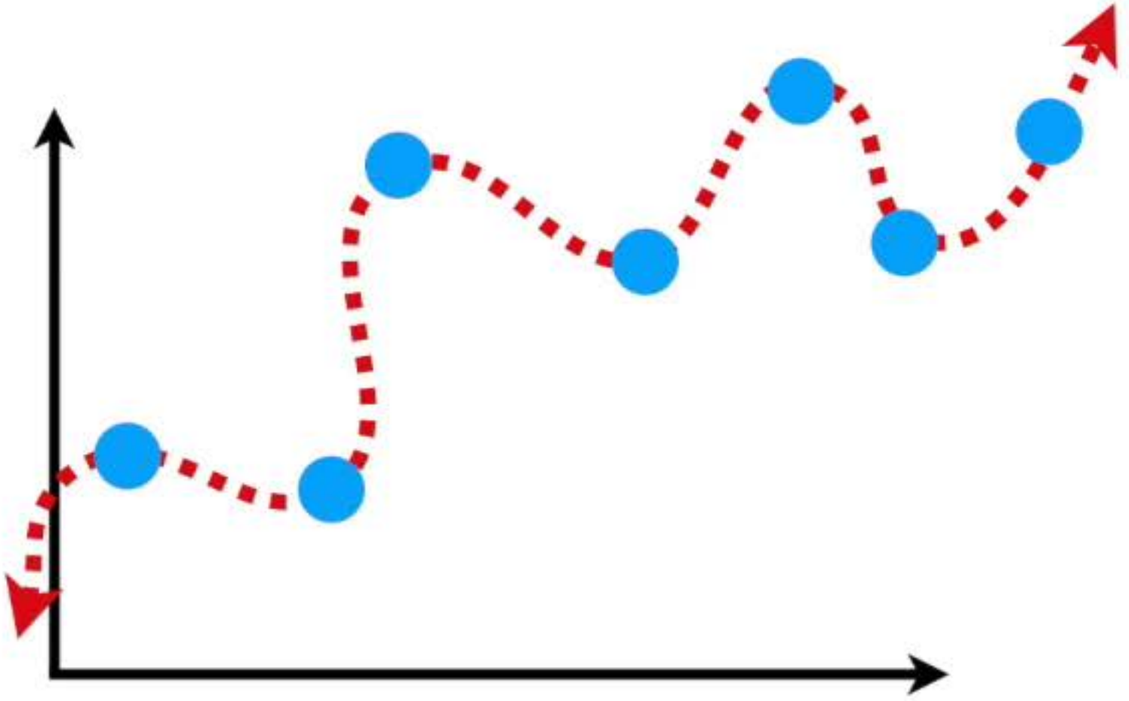
We can compare how well the **Straight Line** and the **Squiggly Line** fit the **training set** by calculating their sums of squares.



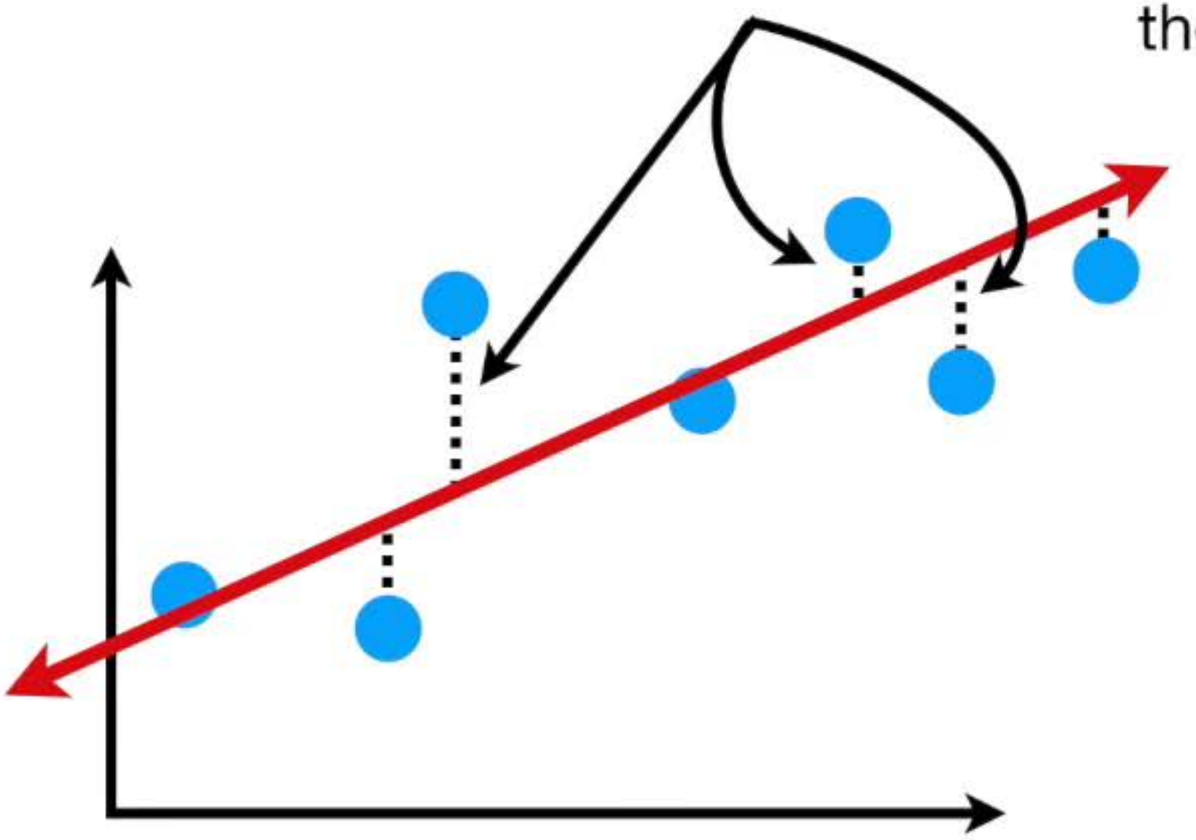


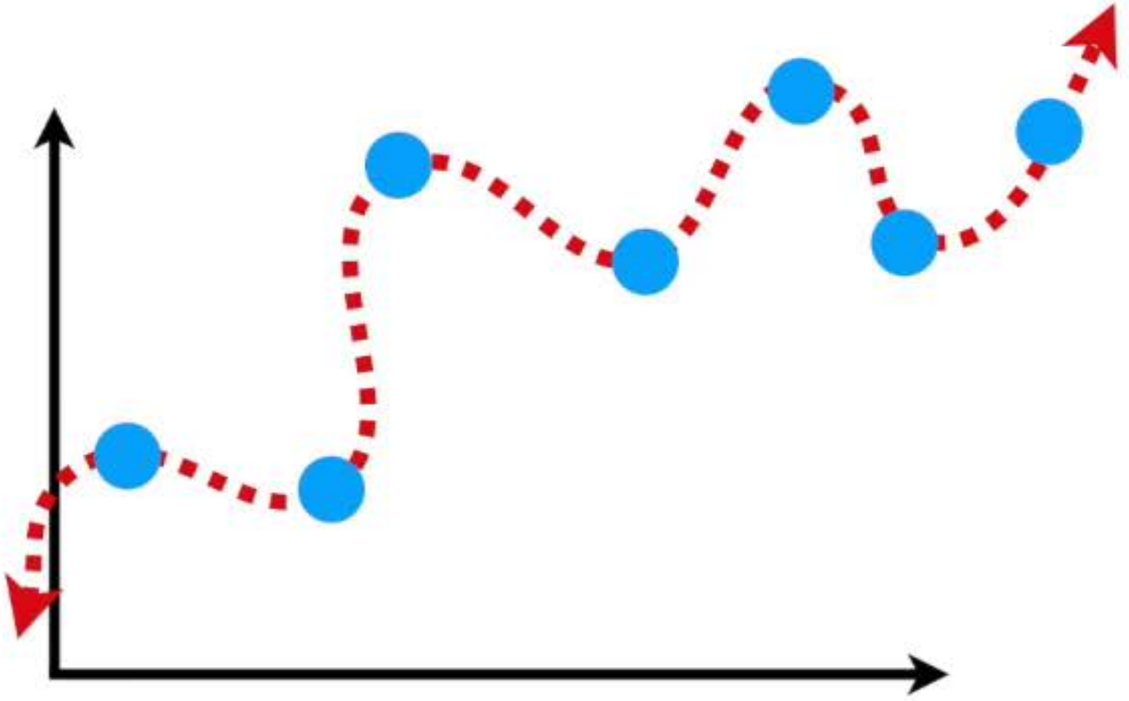
In other words, we measure the distances from the fit lines to the data, square them and add them up.



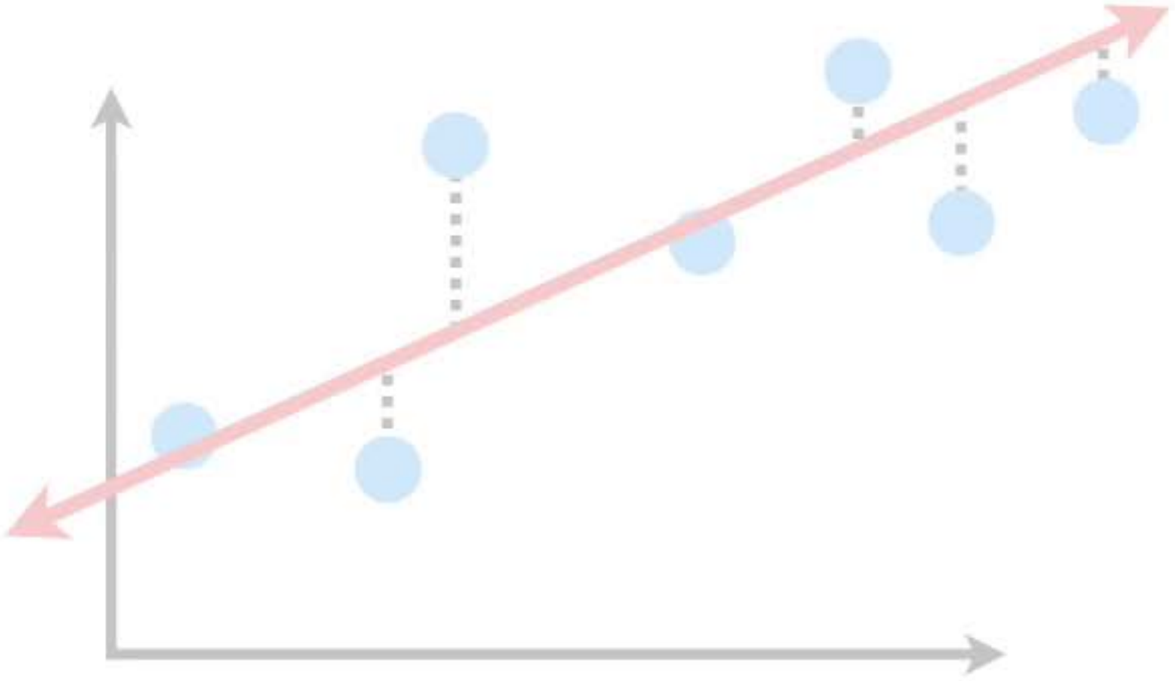


In other words, we measure the distances from the fit lines to the data, square them and add them up.

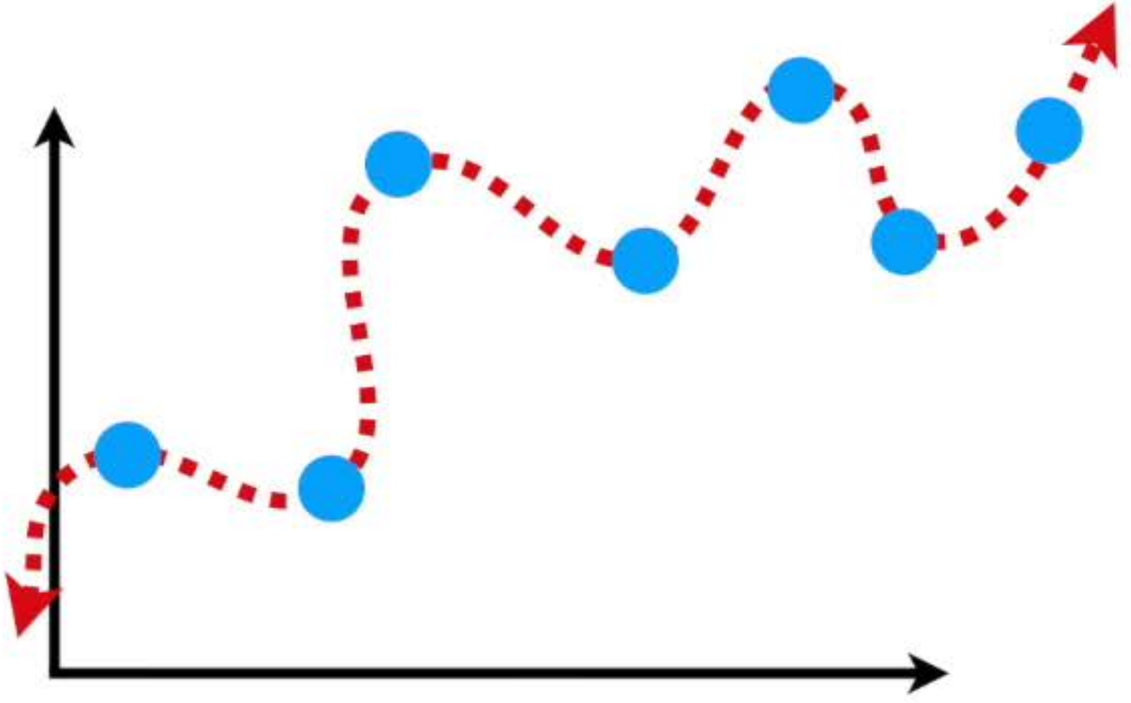




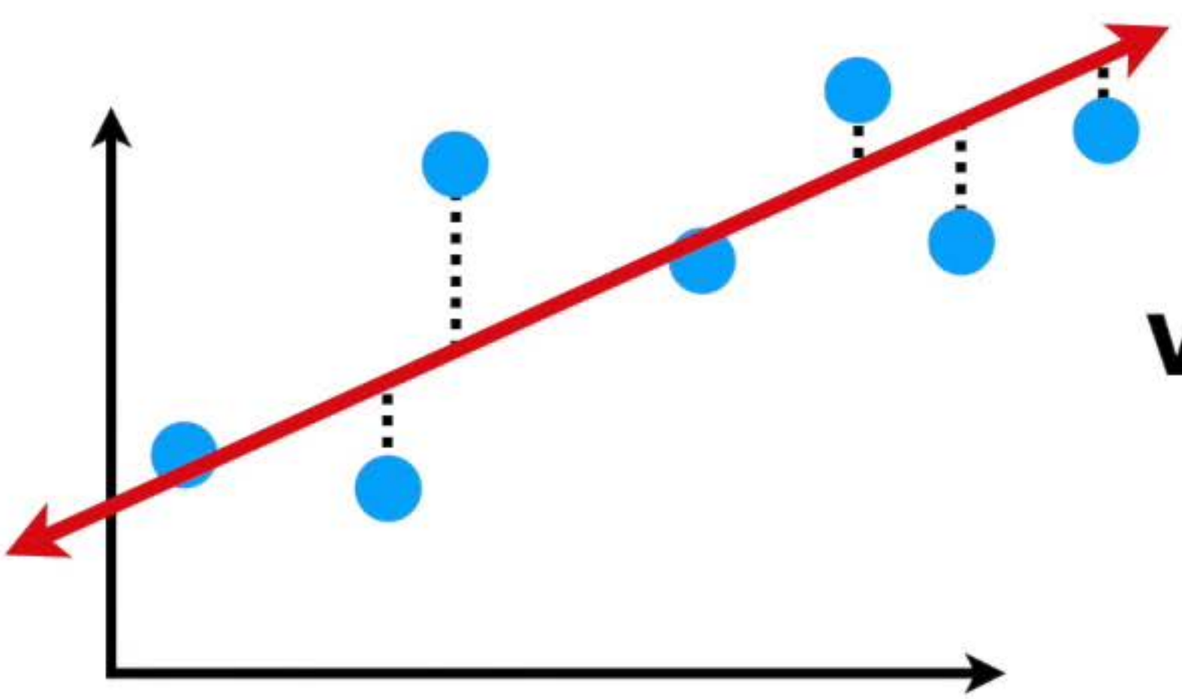
(psst. they are squared so that negative distances do not cancel out the positive distances)…



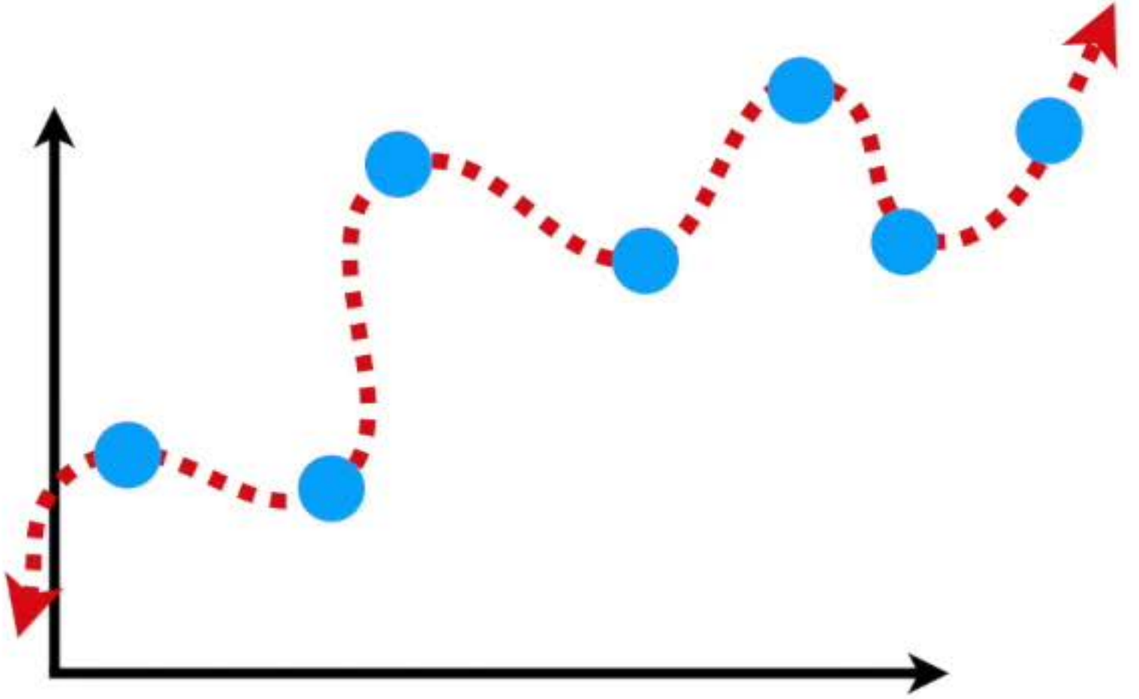
Notice how the **Squiggly Line** fits the data so well that the distances between the line and the data are all 0.



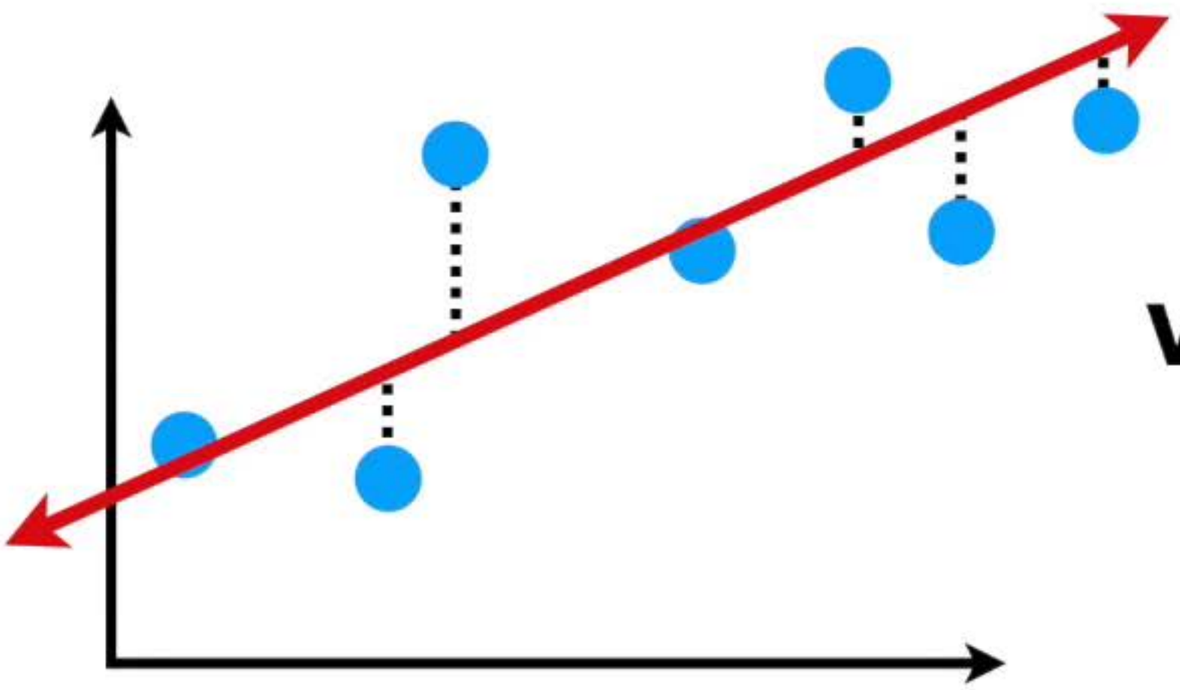
In the contest to see whether the **Straight Line** fits the **training set** better than the **Squiggly Line**…



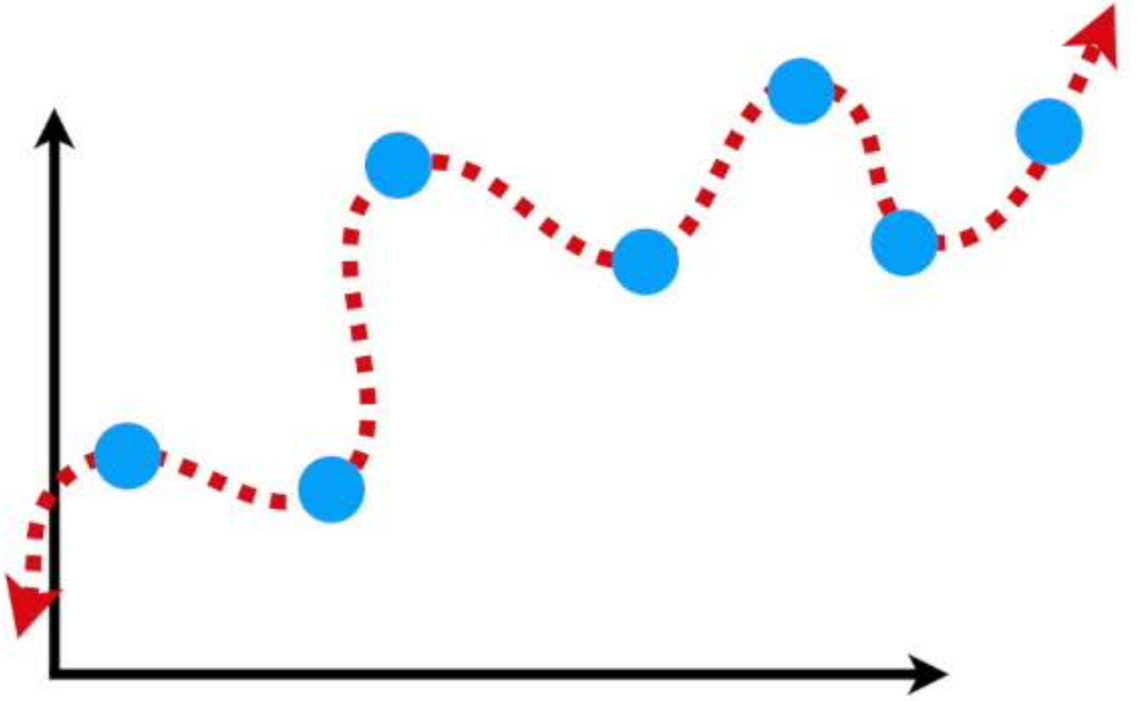
VS.



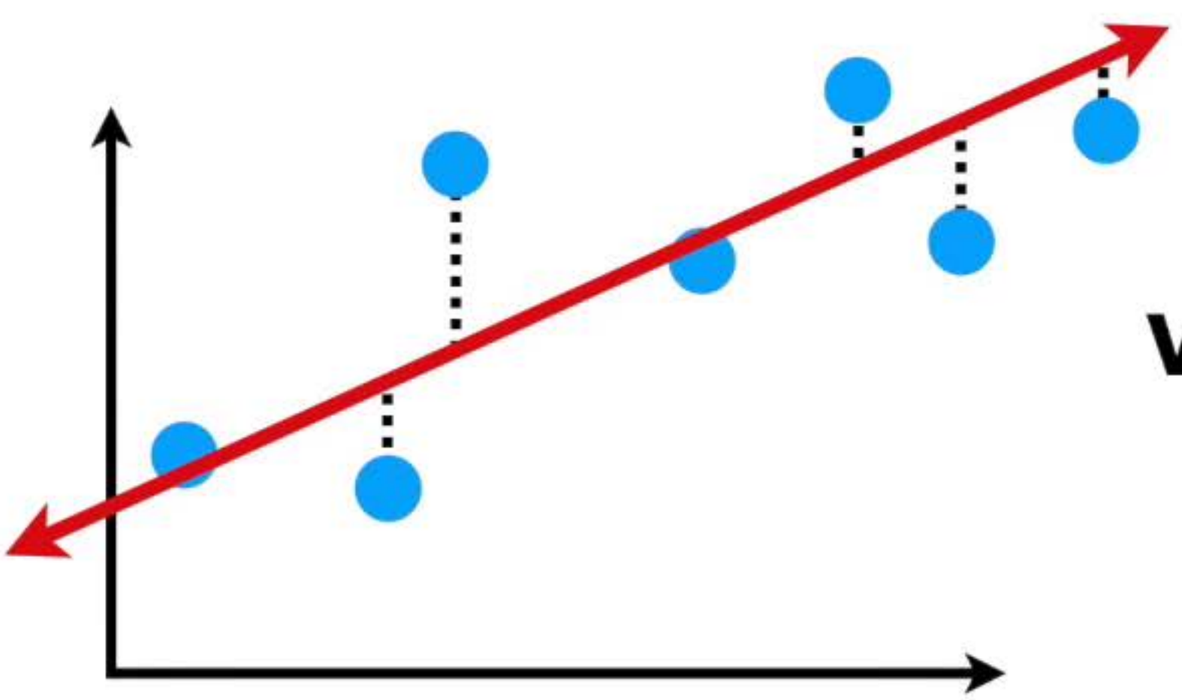
The **Squiggly Line** wins.



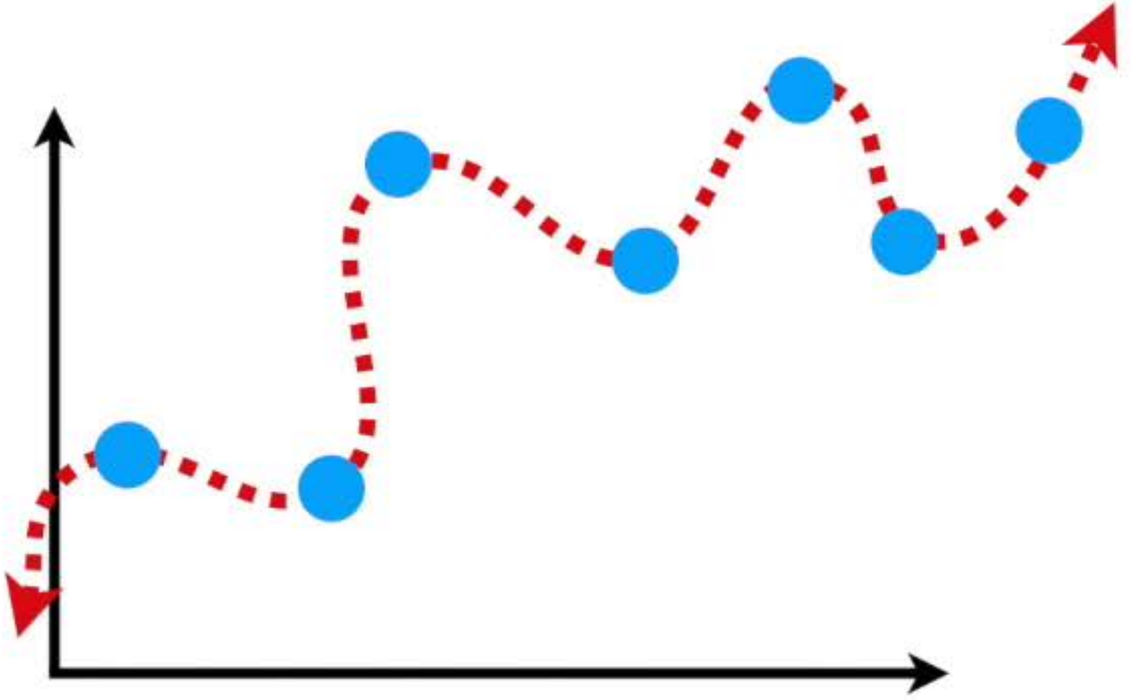
VS.



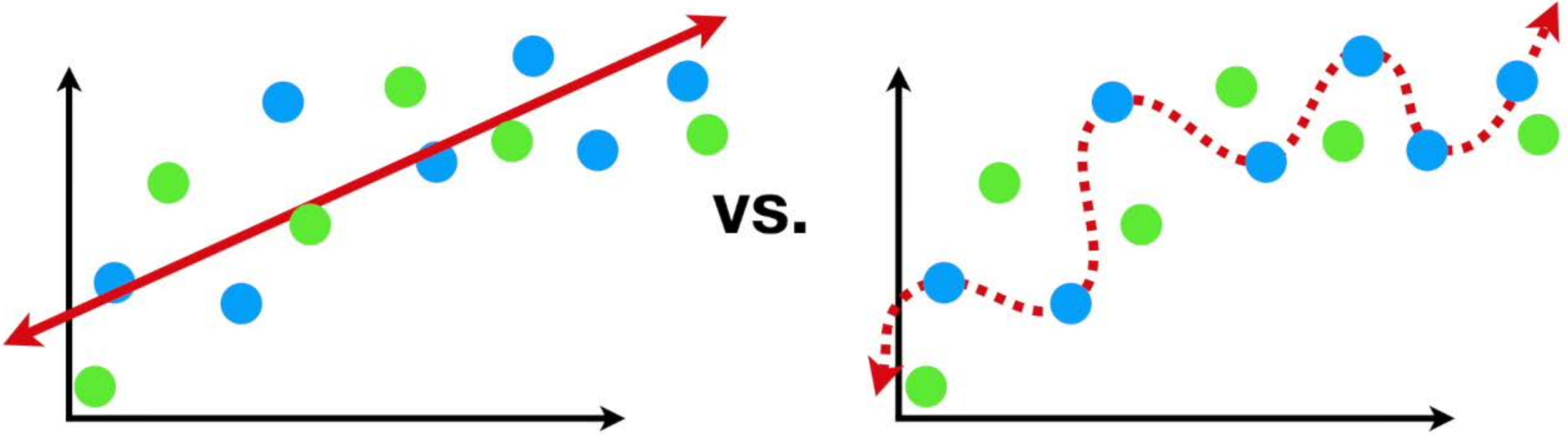
But remember, so far we've only calculated the Sums of Squares for the **training set**.



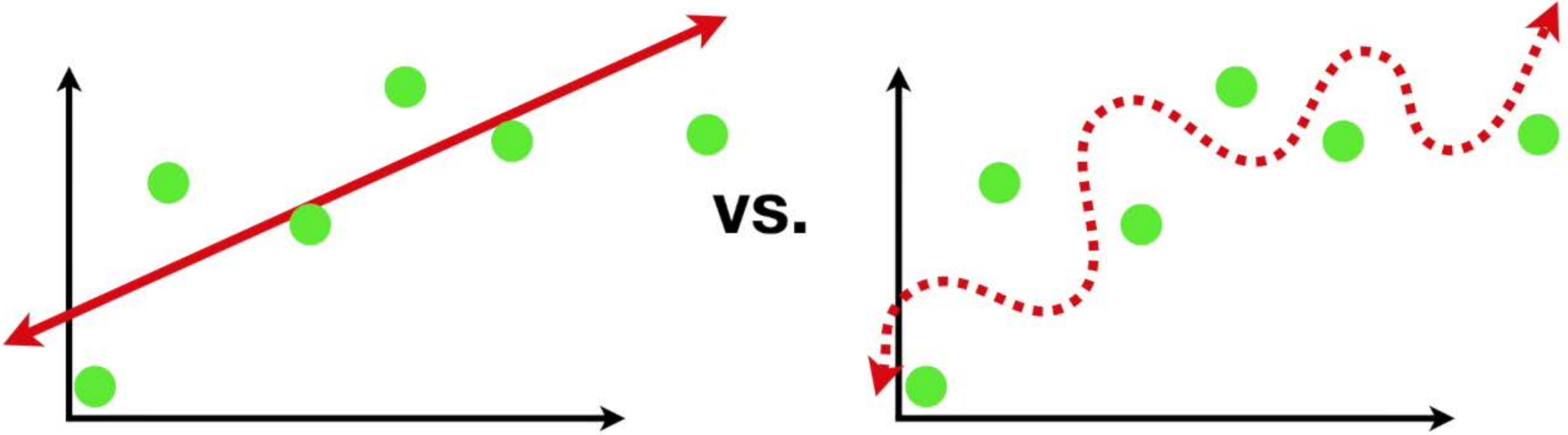
vs.



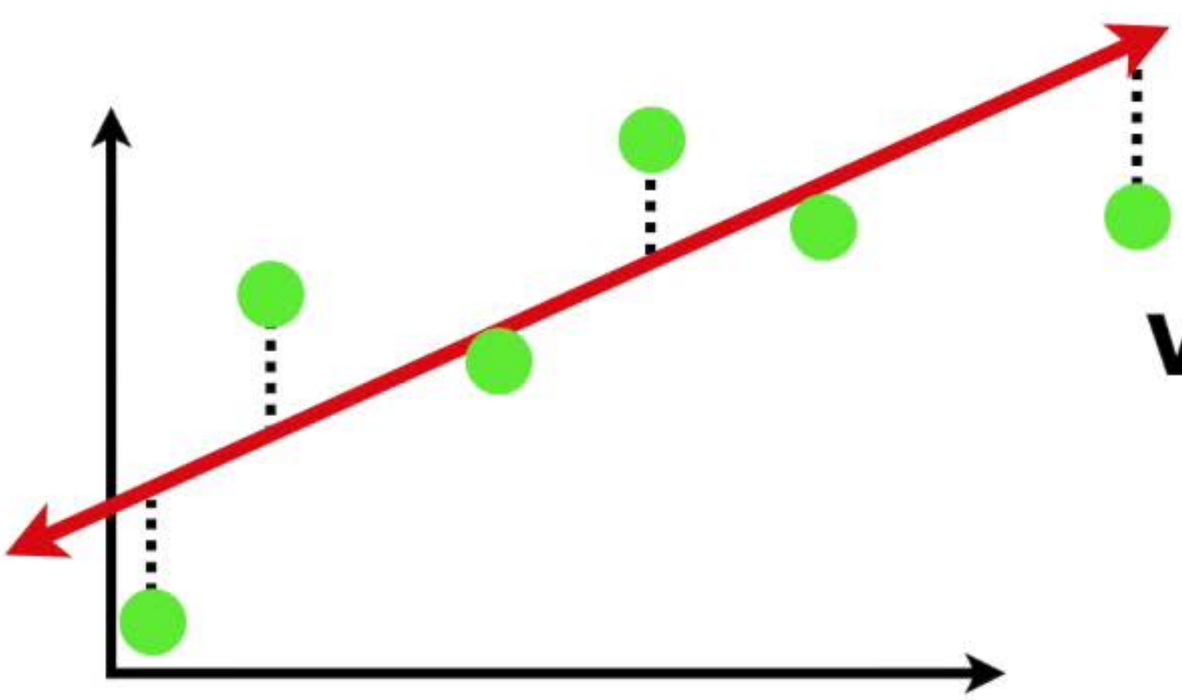
We also have a **testing set**…



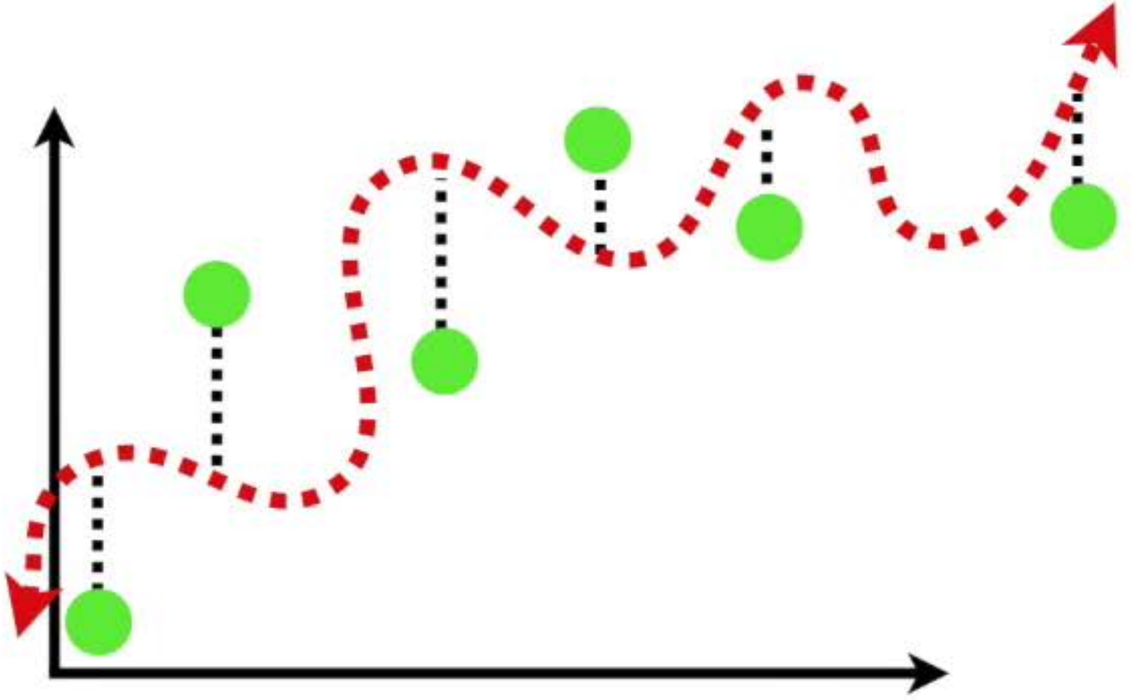
Now let's calculate the Sums of Squares for the **testing set**.



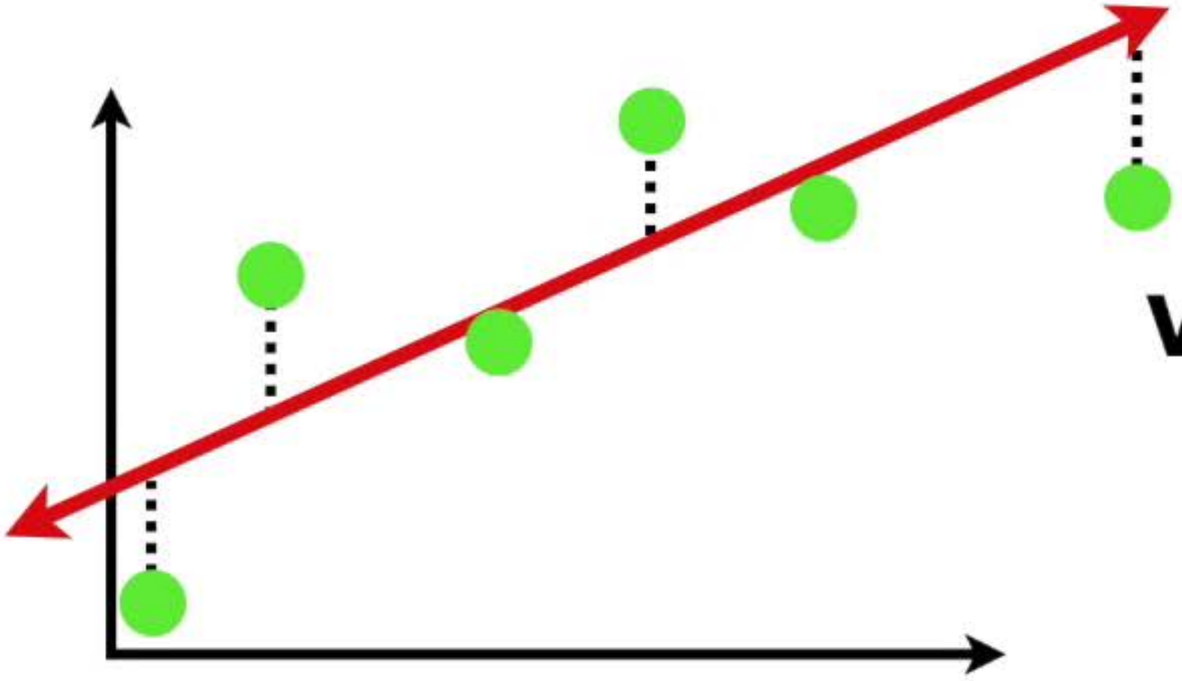
In the contest to see whether the **Straight Line** fits the **testing set** better than the **Squiggly Line**…



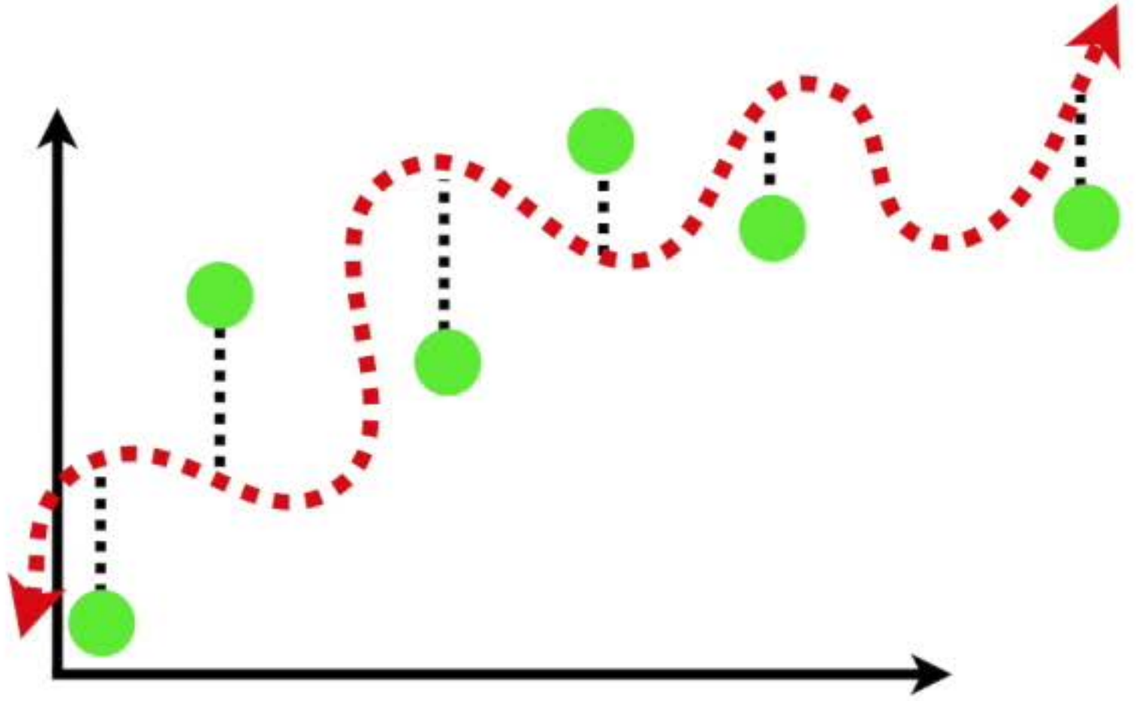
VS.



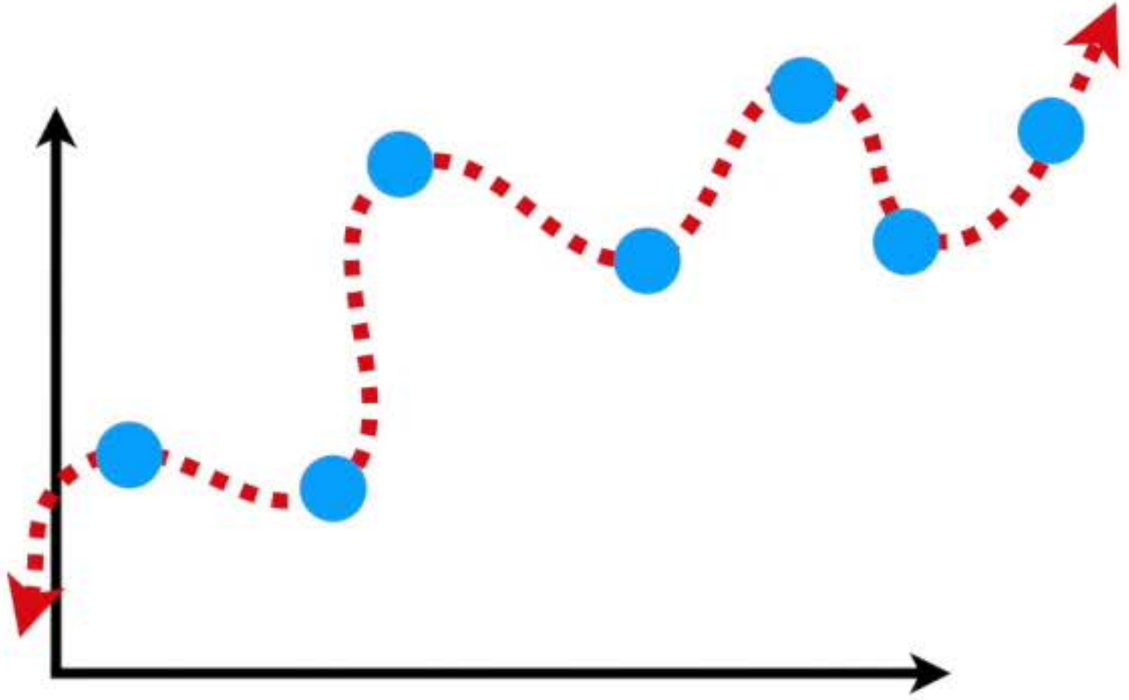
The **Straight Line** wins.

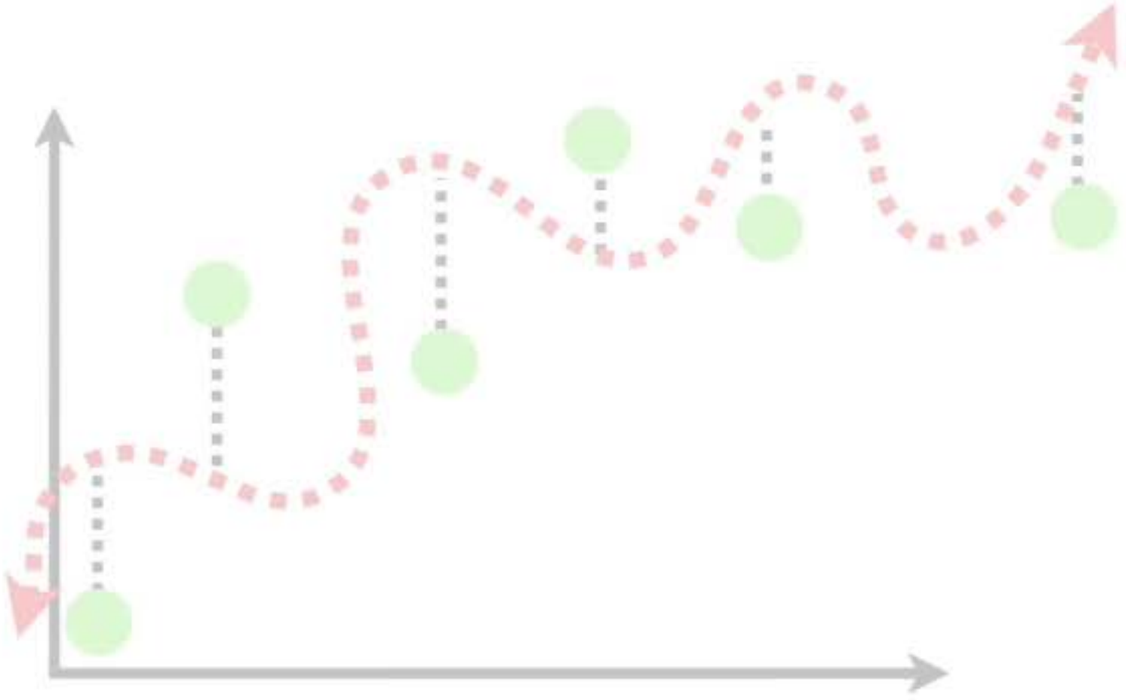


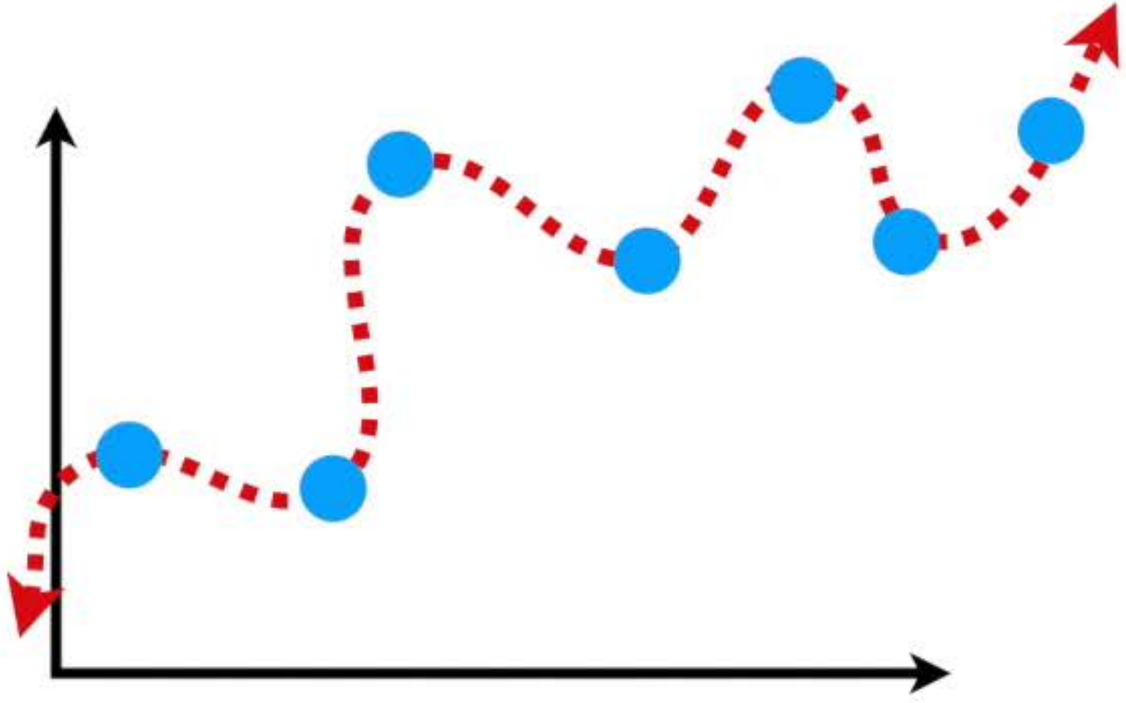
vs.



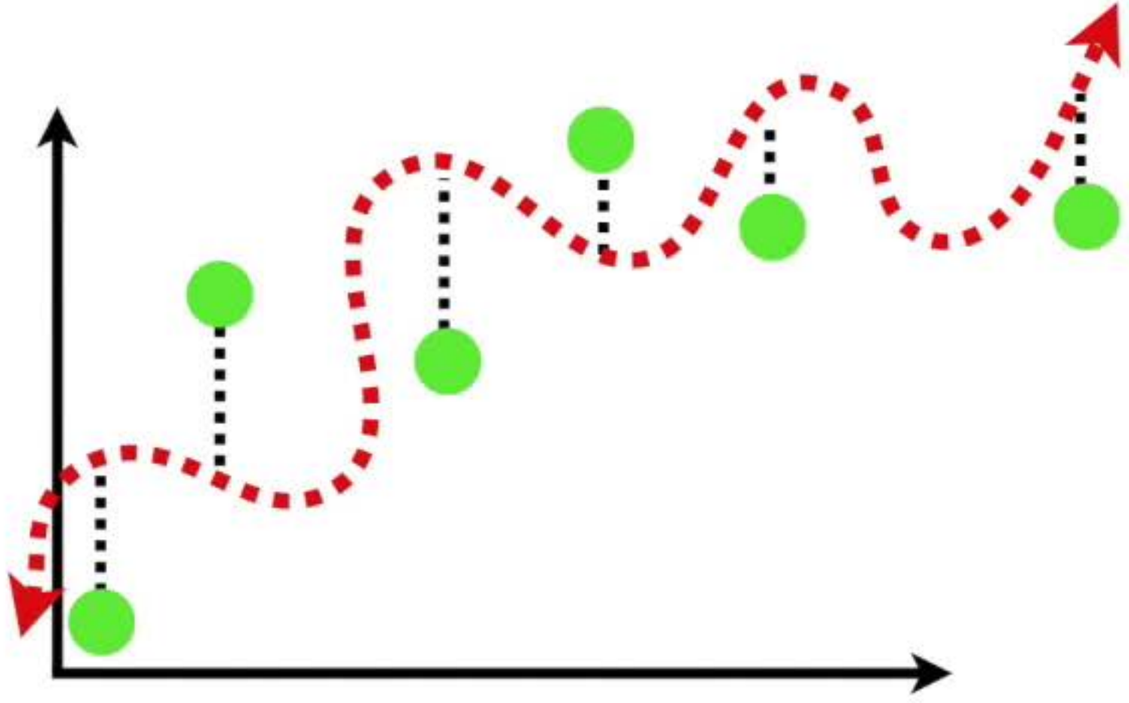
Even though **Squiggly Line** did a great job fitting the **training set**…



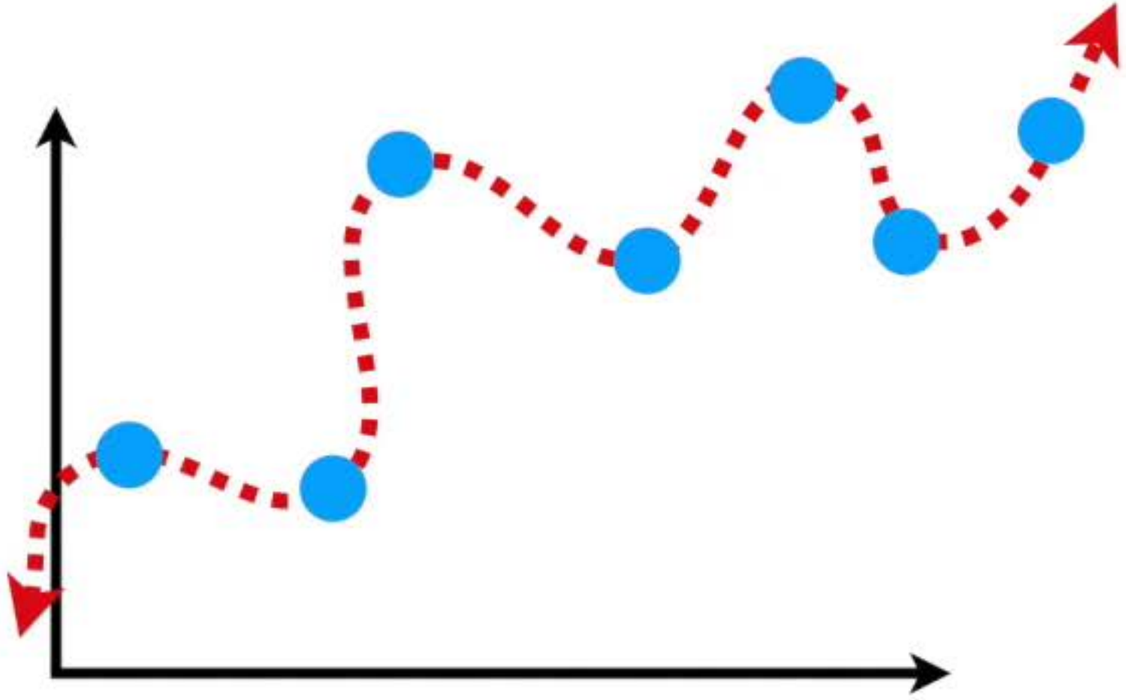


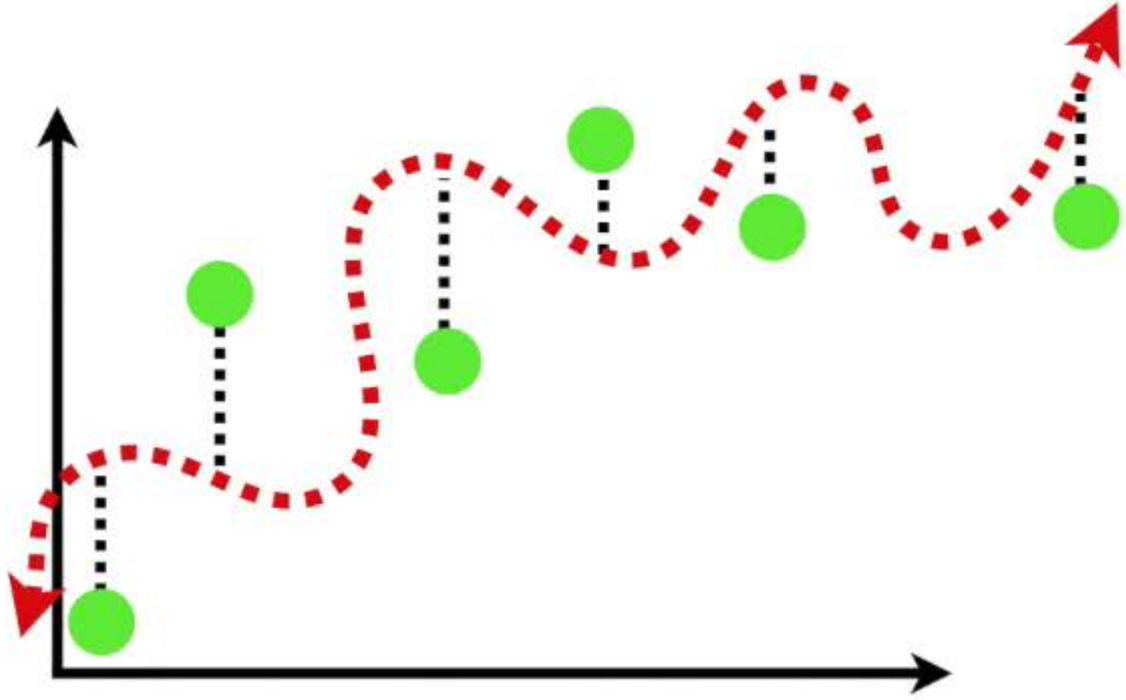


…it did a terrible job fitting the **testing set**…

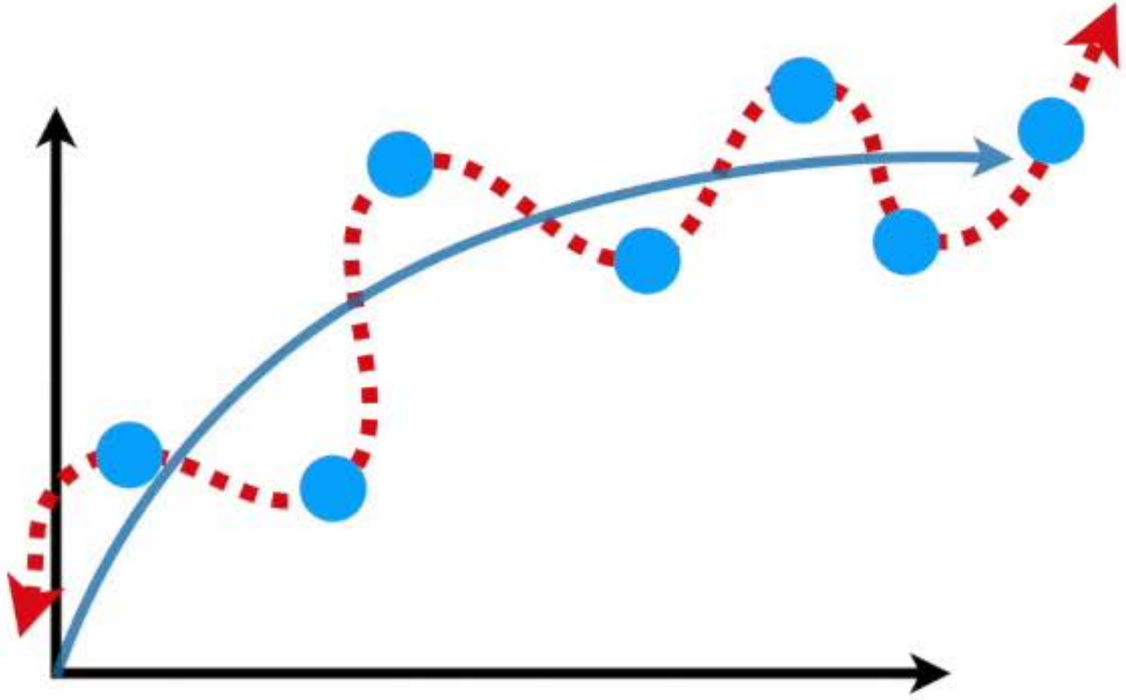


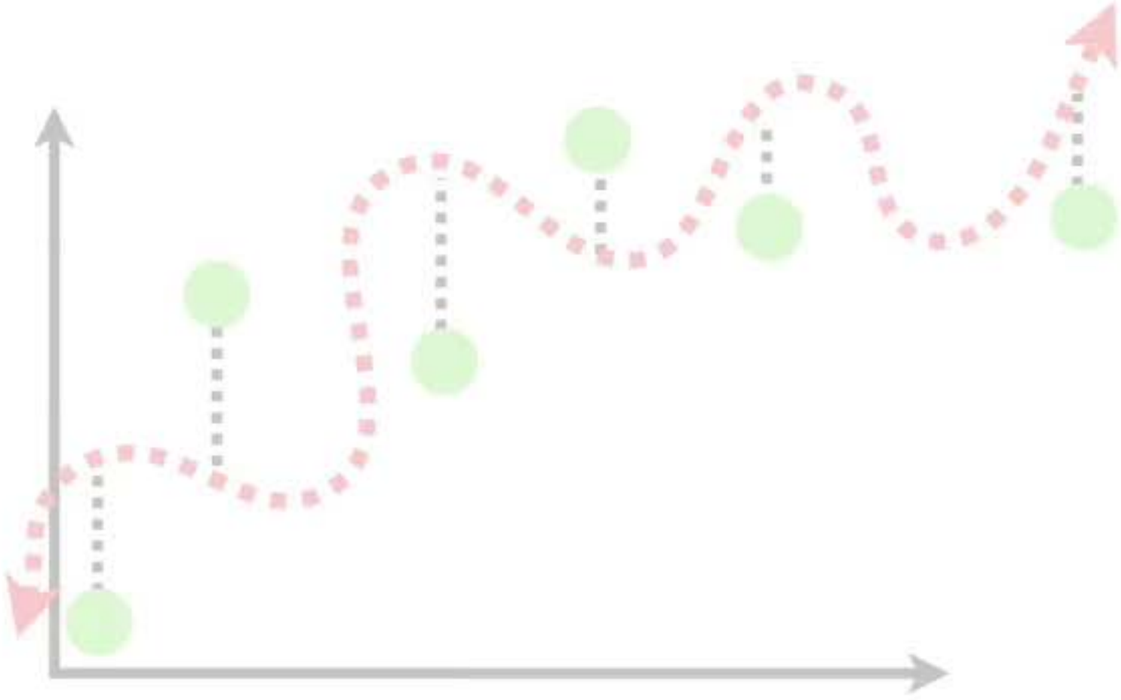
In Machine Learning lingo, the difference in fits between data sets is called **Variance**.



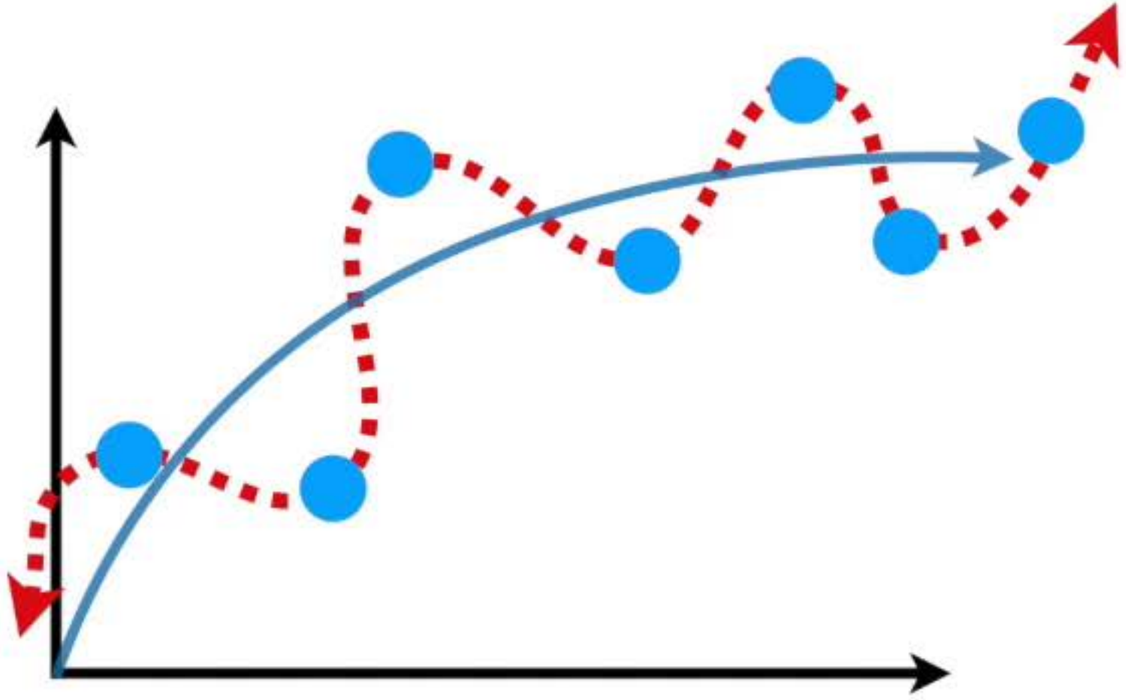


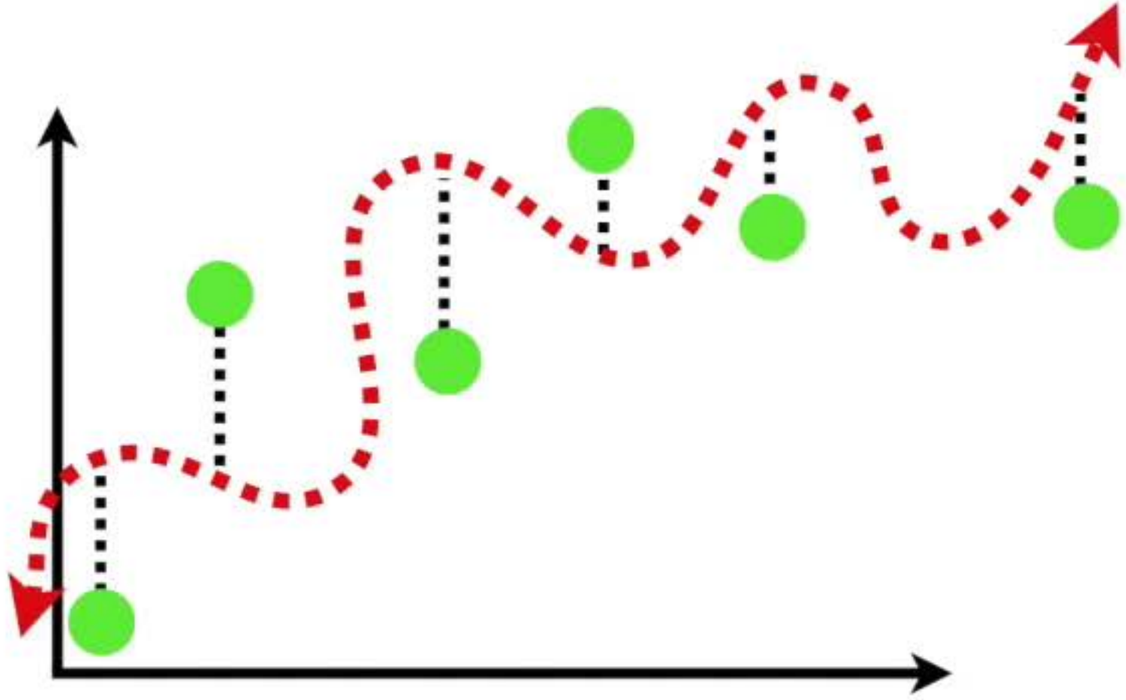
The **Squiggly Line** has **low bias**, since it is flexible and can adapt to the curve in the relationship between weight and hight…



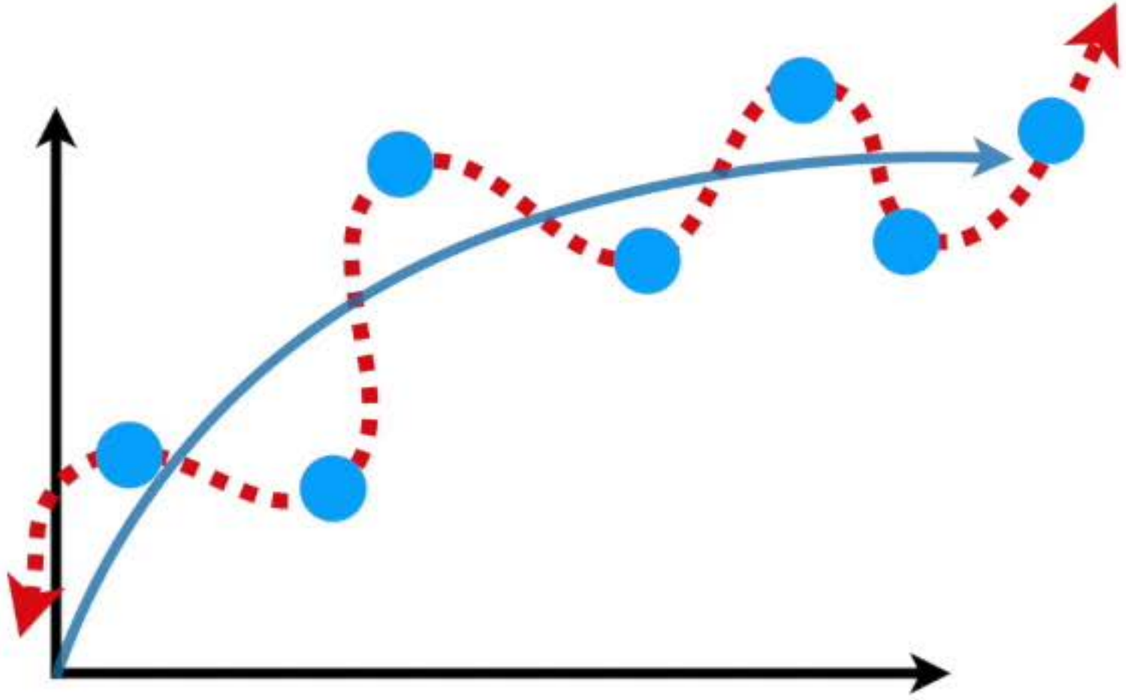


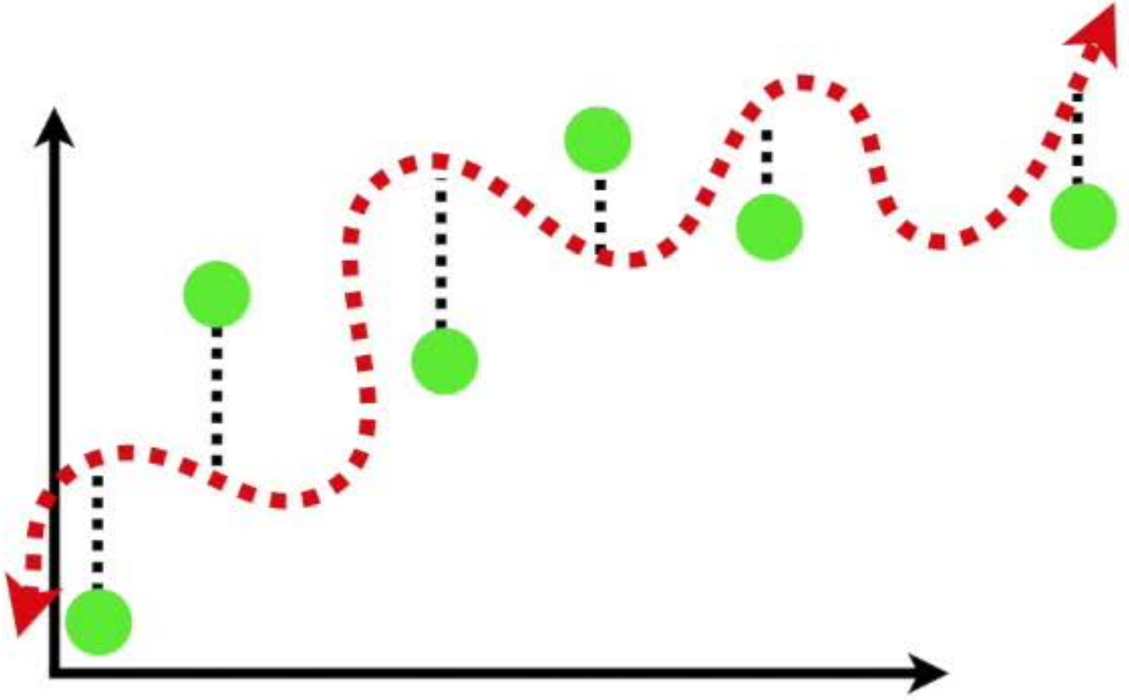
…but the **Squiggly Line** has **high variability**, because it results in vastly different Sums of Squares for different datasets.



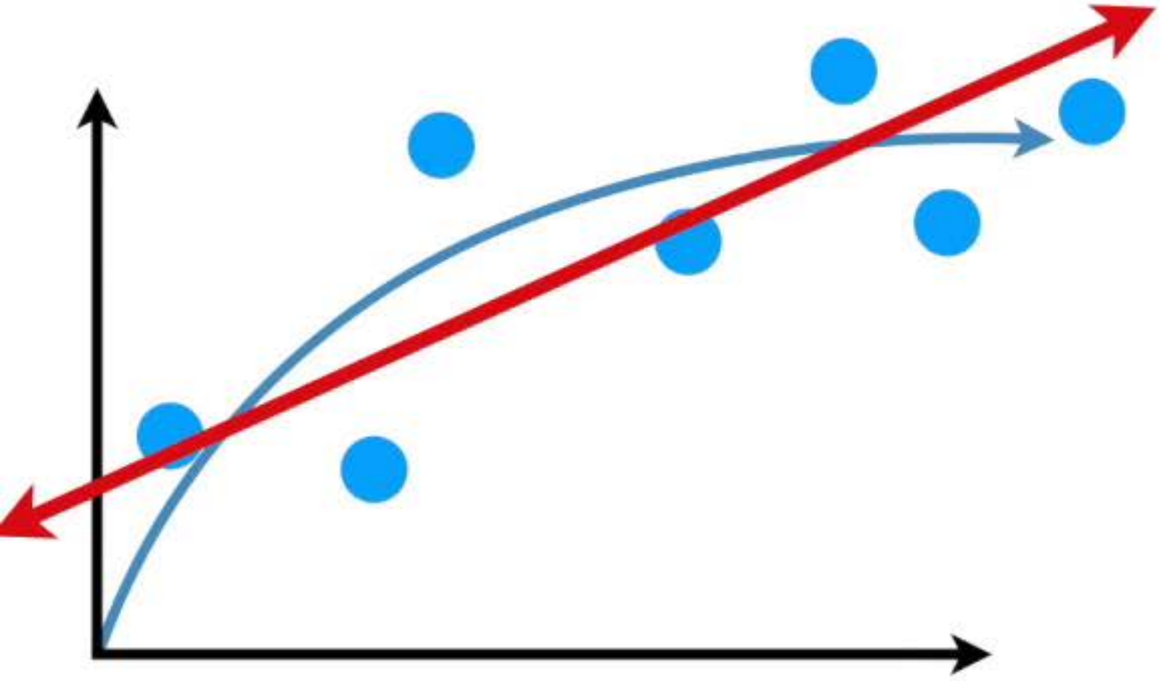


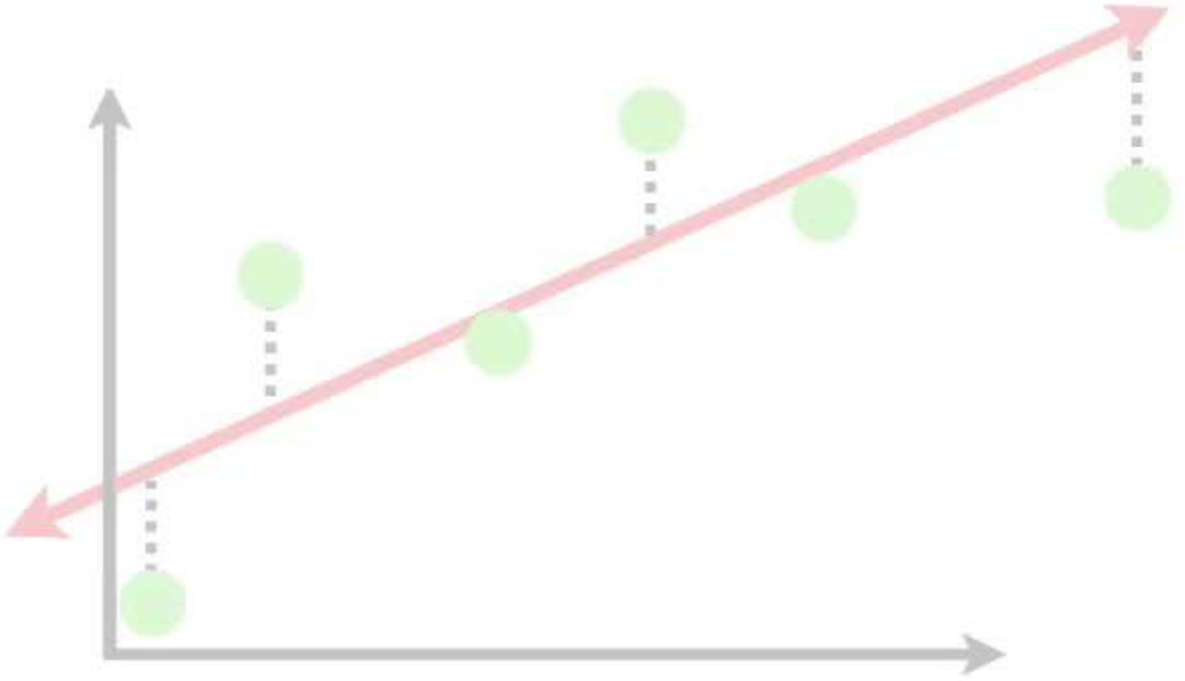
In other words, it's hard to predict how well the **Squiggly Line** will perform with future data sets. It might do well sometimes, and other times it might do terribly.



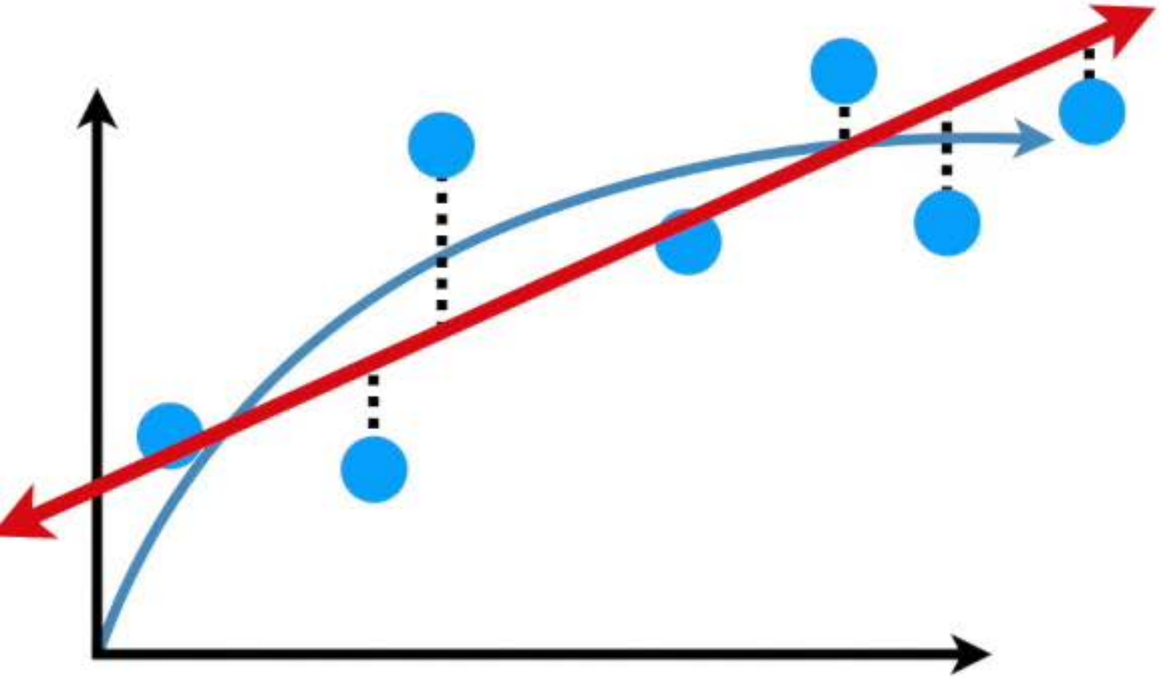


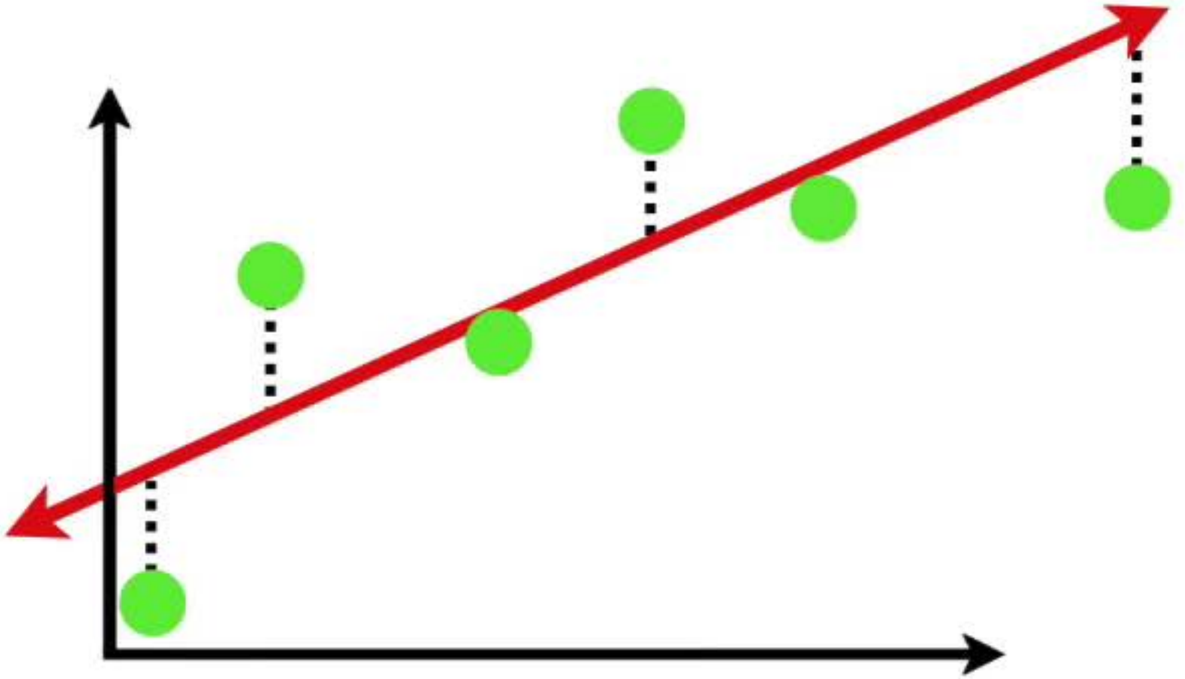
In contrast, the **Straight Line** has relatively **high bias**, since it can not capture the curve in the relationship between weight and height…



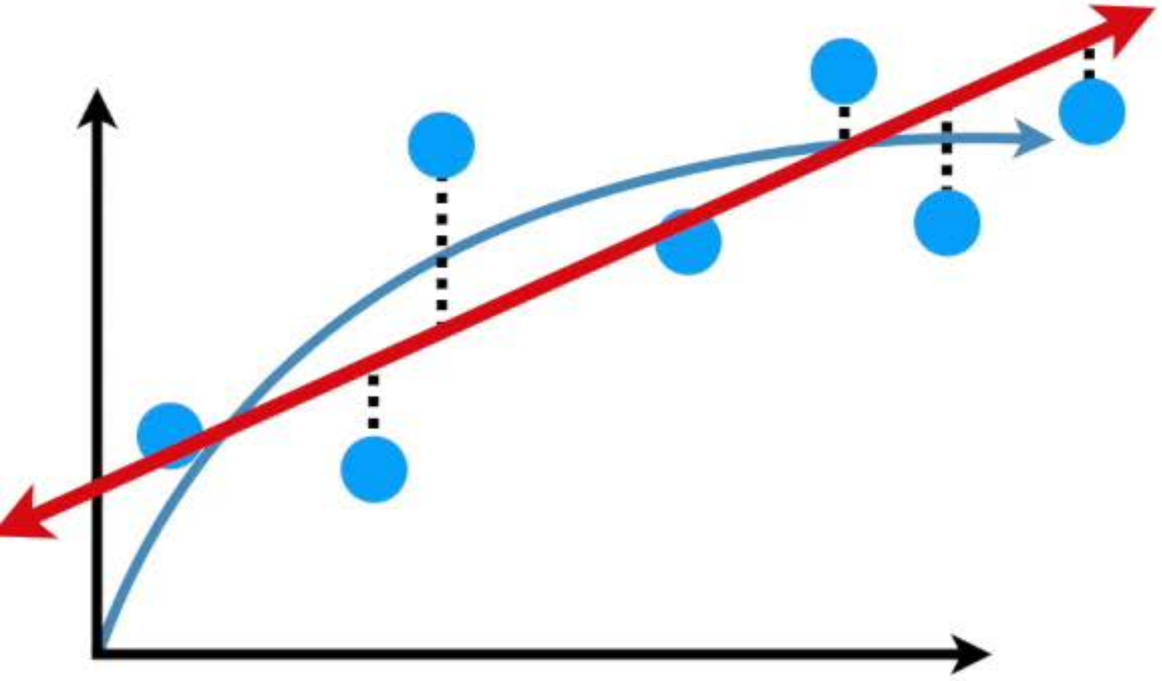


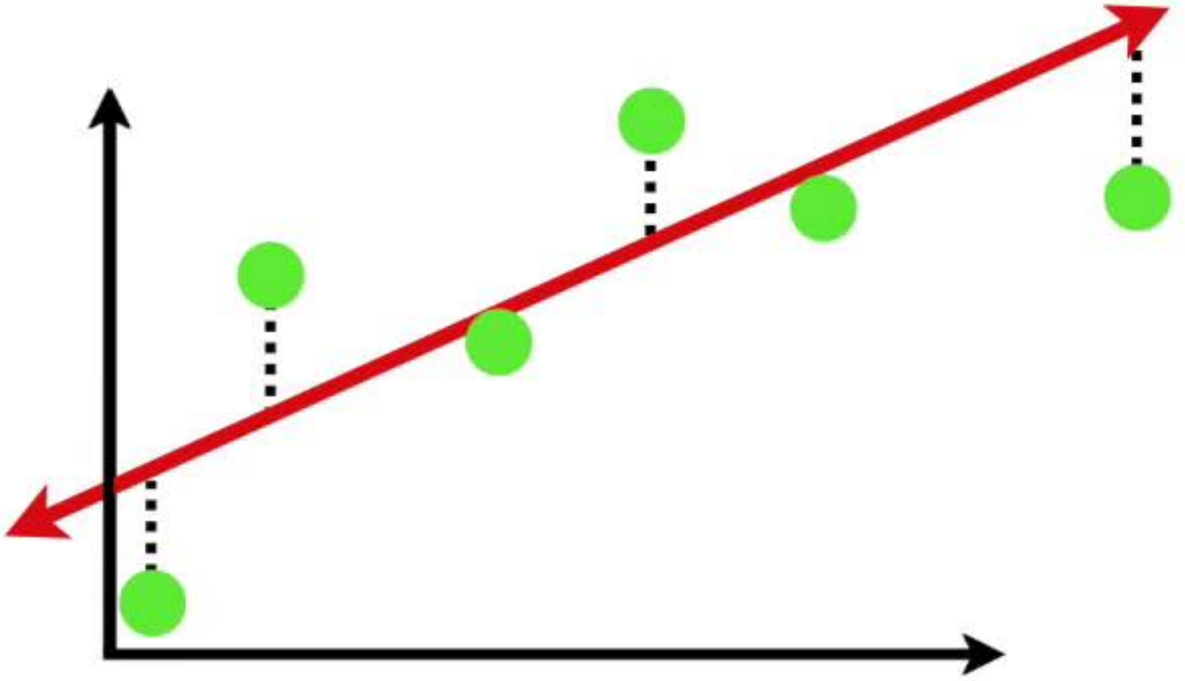
…but the **Straight Line** has relatively **low variance**, because the Sums of Squares are very similar for different datasets.





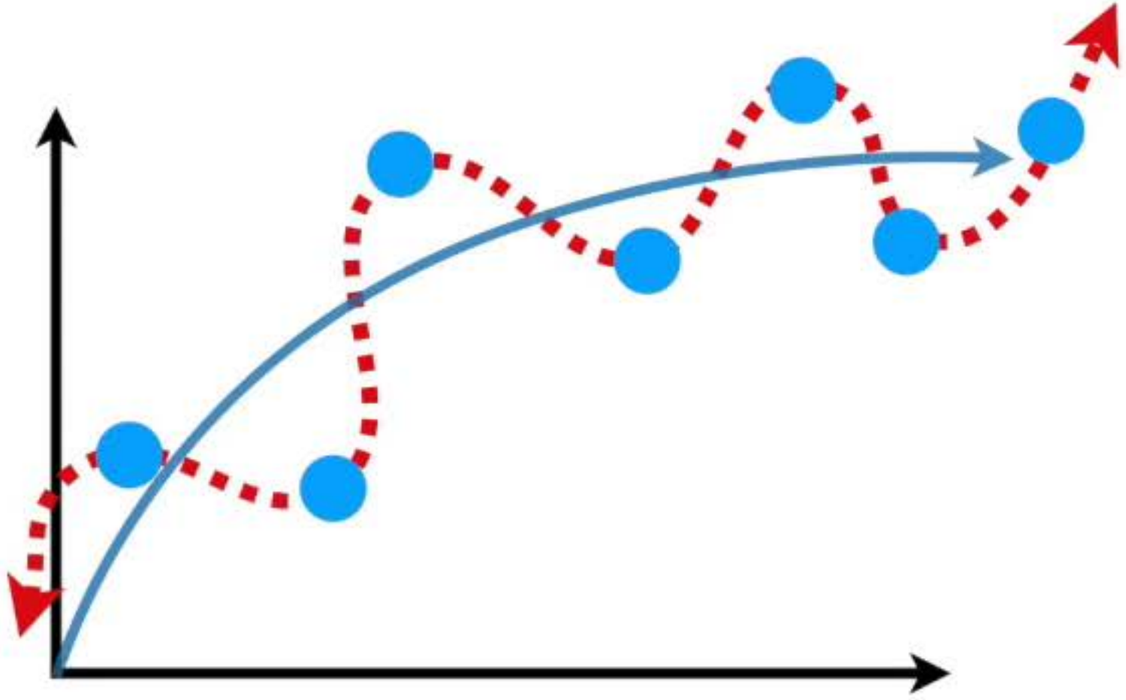
In other words, the **Straight Line** might only give good predictions, and not great predictions. But they will be consistently good predictions.

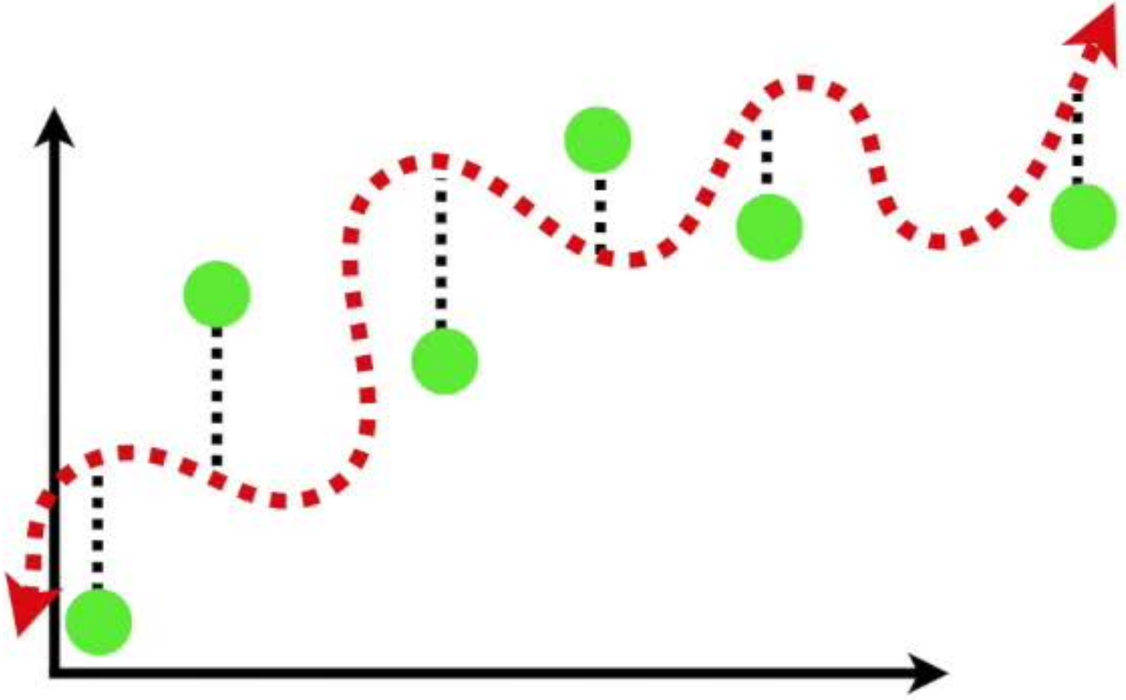




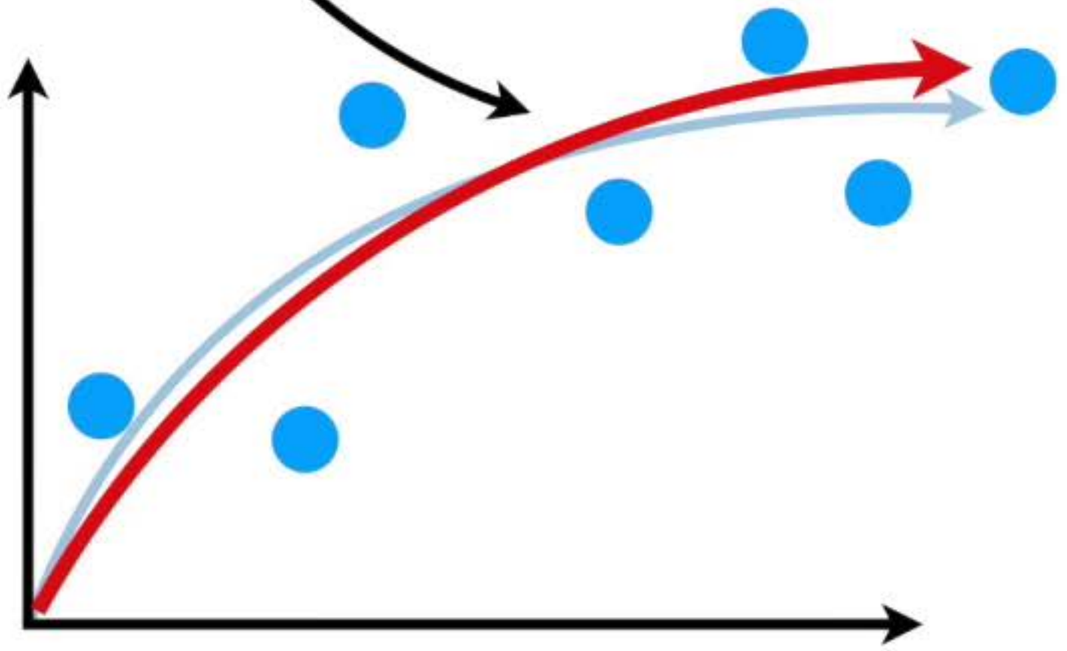
Terminology Alert!!!

Because the **Squiggly Line** fits the **training set** really well, but not the **testing set**, we say that the **Squiggly Line** is **overfit**.

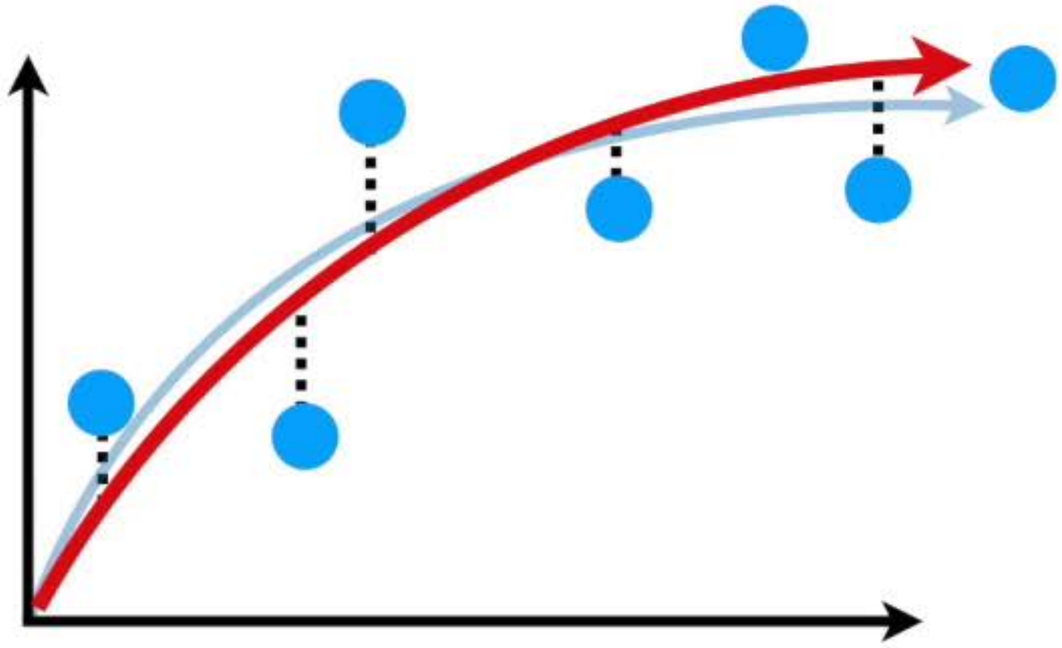


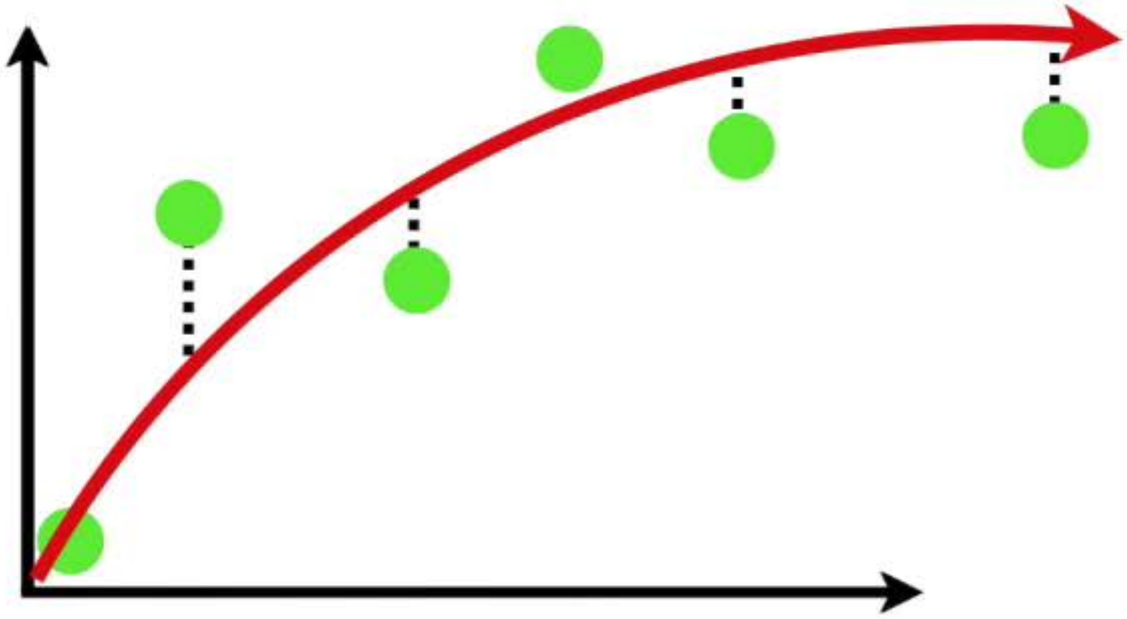


In machine learning, the ideal algorithm has **low bias** and can accurately model the true relationship…

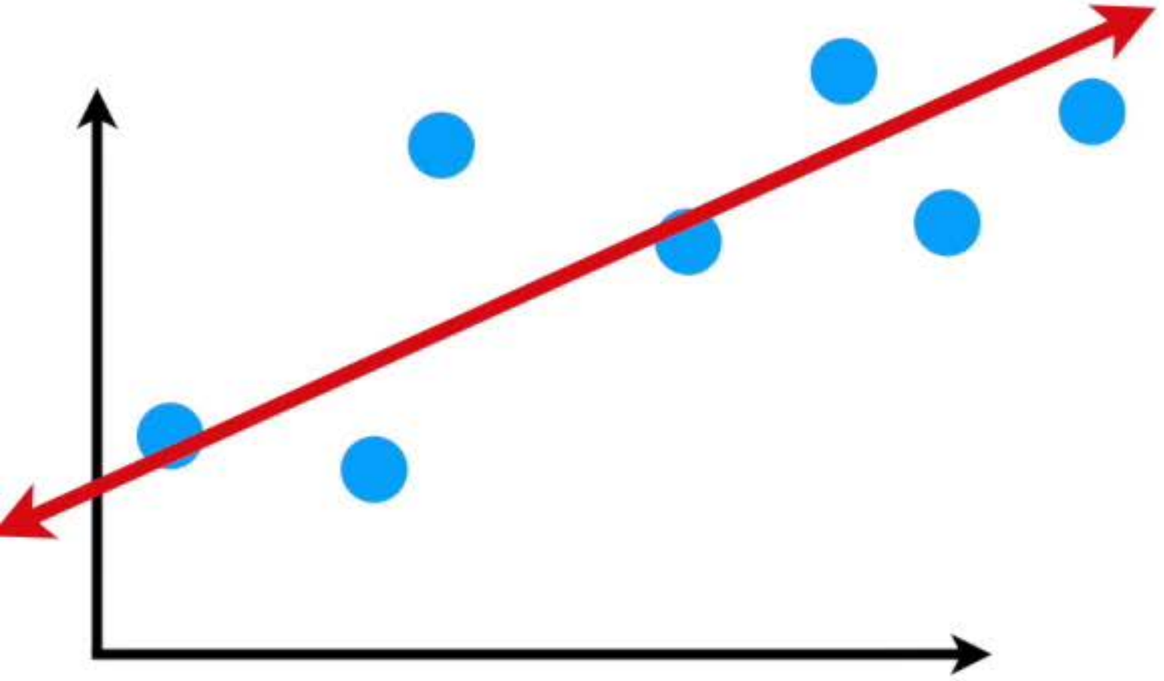


…and it has **low variability**, by producing consistent predictions across different datasets.

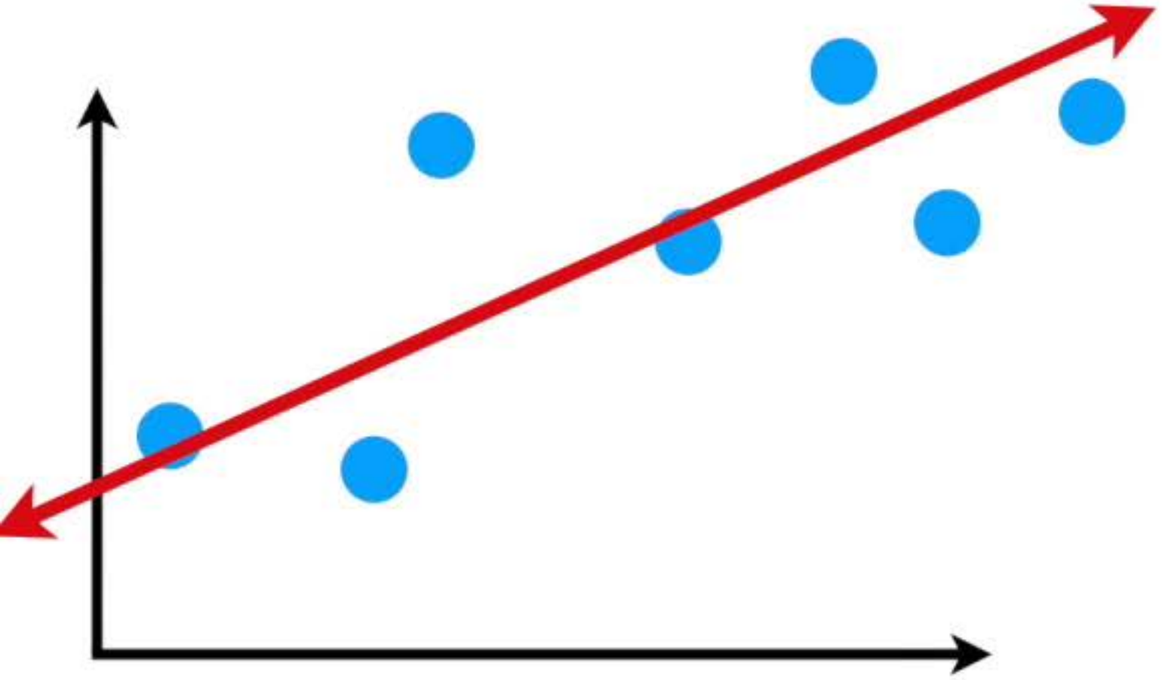


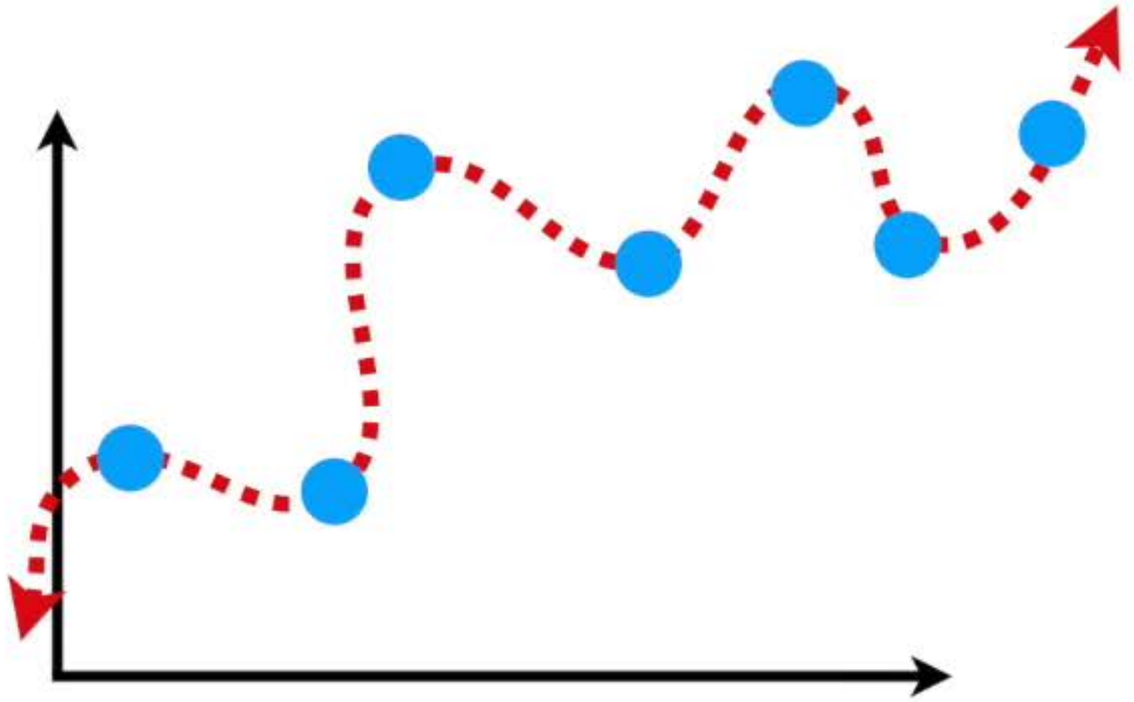


This is done by finding the sweet spot between a simple model…



…and a complex model.





Terminology Alert!!!

Three commonly used methods for finding the sweet spot between simple and complicated models are: **regularization**, **boosting** and **bagging**.