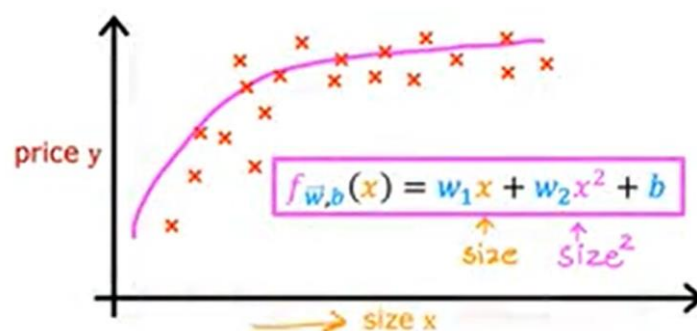


Régression polynomiale

So far we've just been fitting straight lines to our data. Let's take the ideas of multiple linear regression and feature engineering to come up with a new algorithm called polynomial regression, which will let you fit curves, non-linear functions, to your data. Let's say you have a housing data-set that looks like this, where feature x is the size in square feet. It doesn't look like a straight line fits this data-set very well. Maybe you want to fit a curve, maybe a quadratic function to the data like this which includes a size x and also x squared, which is the size raised to the power of two.

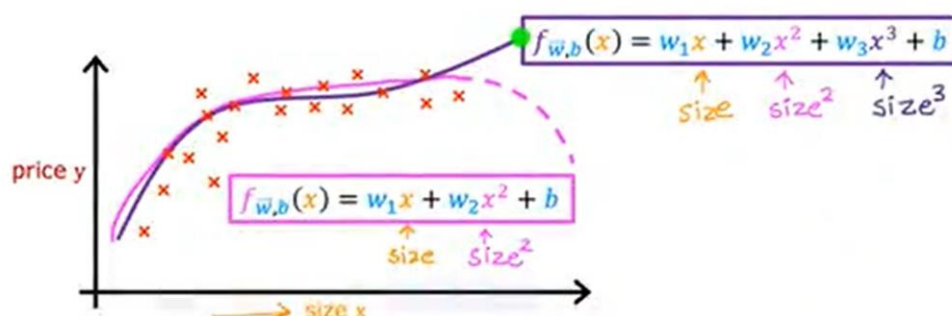
Polynomial regression



Maybe that will give you a better fit to the data. But then you may decide that your quadratic model doesn't really make sense because a quadratic function eventually comes back down. Well, we wouldn't really expect housing prices to go down when the size increases. Big houses seem like they should usually cost more. Then you may choose a cubic function where we now have not only x squared, but x cubed. Maybe this model produces this curve here, which is a somewhat better fit to the data because the size does eventually come back up as the size increases.

These are both examples of polynomial regression, because you took your optional feature x , and raised it to the power of two or three or any other power. In the case of the cubic function, the first feature is the size, the second feature is the size squared, and the third feature is the size cubed.

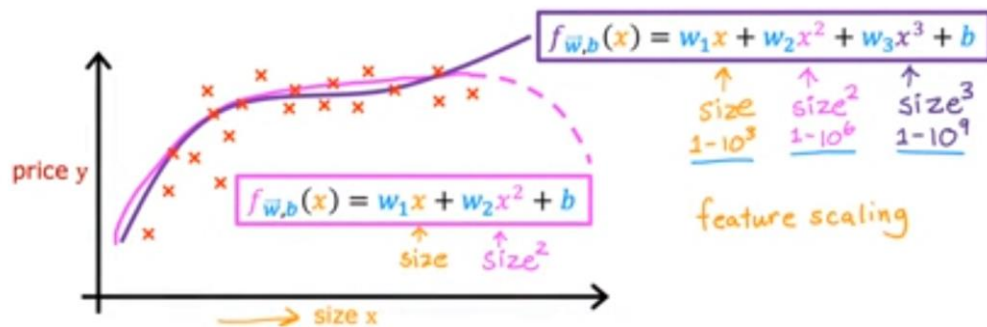
Polynomial regression



I just want to point out one more thing, which is that if you create features that are these powers like the square of the original features like this, then feature scaling becomes increasingly important. If the size of the house ranges from say, 1-1,000 square feet, then the second feature, which is a size squared,

will range from one to a million, and the third feature, which is size cubed, ranges from one to a billion. These two features, x squared and x cubed, take on very different ranges of values compared to the original feature x . If you're using gradient descent, it's important to apply feature scaling to get your features into comparable ranges of values.

Polynomial regression



Finally, just one last example of how you really have a wide range of choices of features to use. Another reasonable alternative to taking the size squared and size cubed is to say use the square root of x . Your model may look like w_1 times x plus w_2 times the square root of x plus b . The square root function looks like this, and it becomes a bit less steep as x increases, but it doesn't ever completely flatten out, and it certainly never ever comes back down.

This would be another choice of features that might work well for this data-set as well. You may ask yourself, how do I decide what features to use? Later in the second course in this specialization, you see how you can choose different features and different models that include or don't include these features, and you have a process for measuring how well these different models perform to help you decide which features to include or not include.

For now, I just want you to be aware that you have a choice in what features you use. By using feature engineering and polynomial functions, you can potentially get a much better model for your data.

Choice of features

