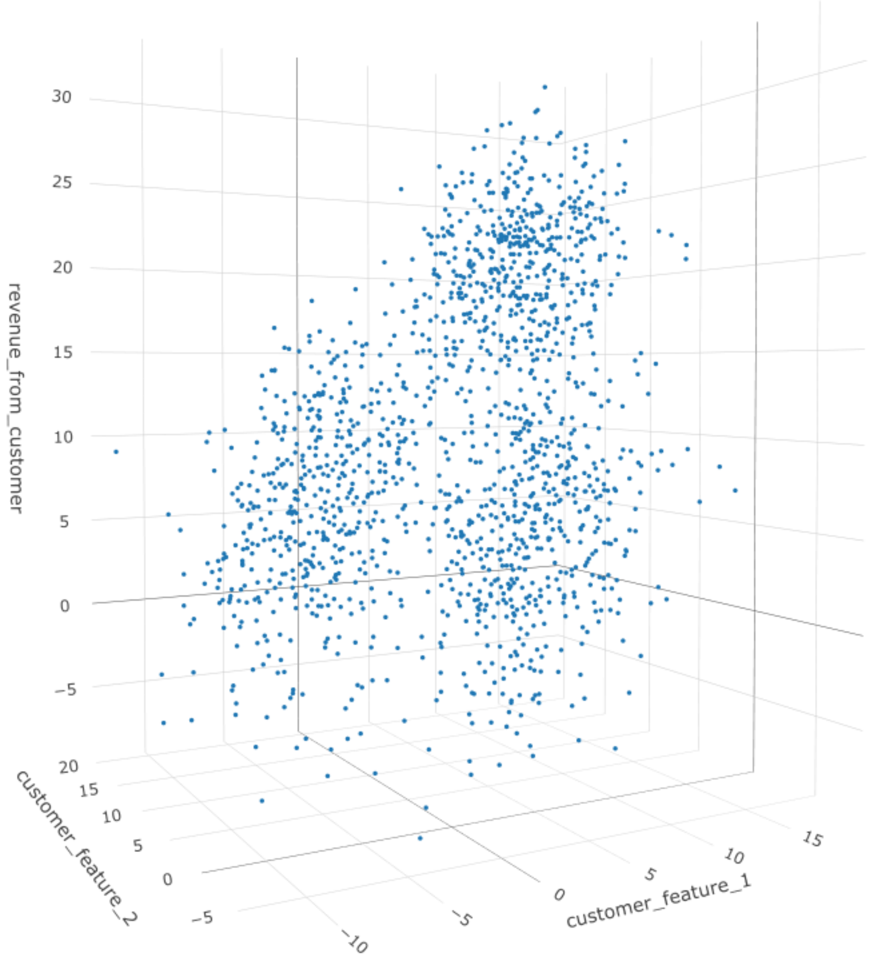
# Machine Learning Live
# Session #6

# Supervised Learning

- Two types of supervised learning: regression and classification




Regression

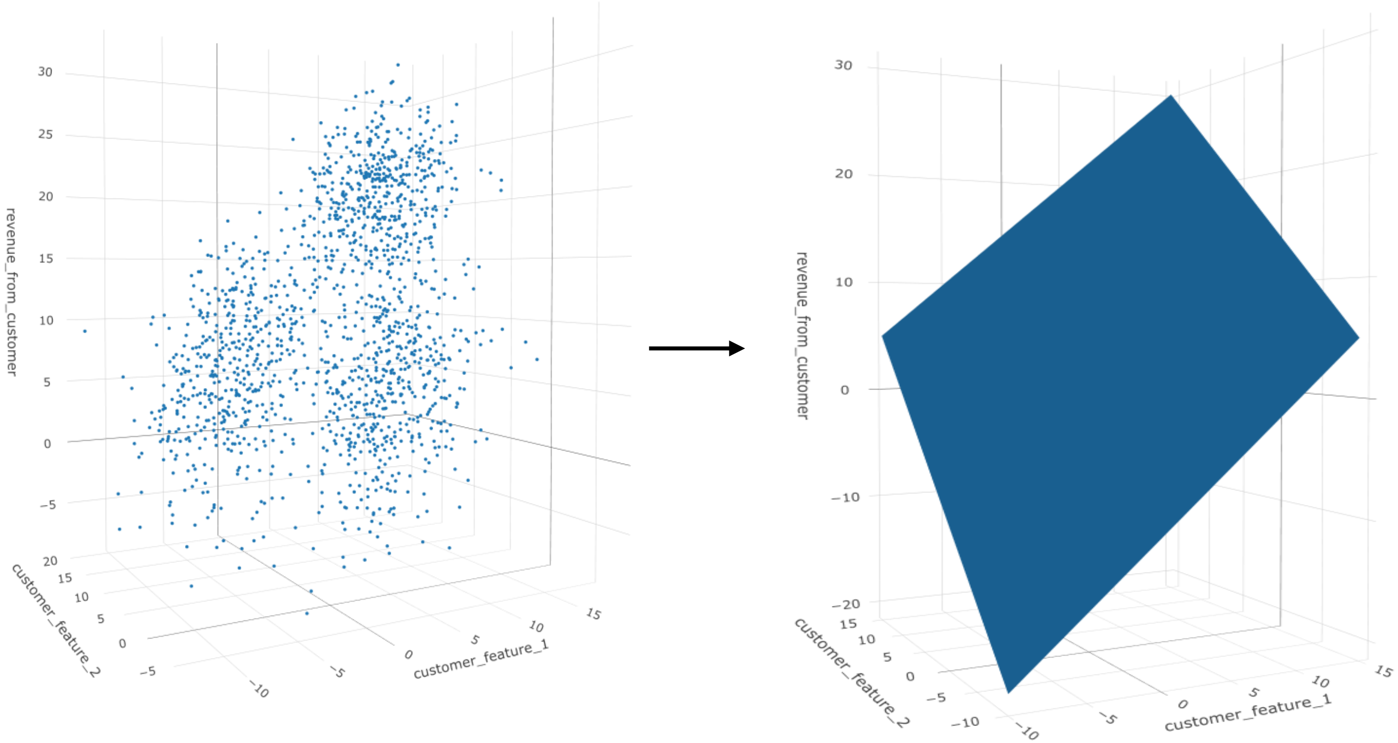

Classification

# Regression Example

- Regression
  - The label (for regression, also called response) is a numerical variable
  - Want to predict the response for a new observation

| Income (Customer Feature 1) | Age (Customer Feature 2) | Revenue from Customer |
|---|---|---|
| 22003 | 45 | 14.03875 |
| 57230 | 54 | 23.31168 |
| 75137 | 28 | 24.05046 |
| 31208 | 54 | 18.5386 |
| 54078 | 23 | 18.50195 |
| 44413 | 44 | 20.63106 |
| 55237 | 46 | 22.32953 |

Training data




$$Predicted\ Revenue = \hat{y}(Income, Age) = w_1^* \times Income + w_2^* \times Age + w_0^*$$

Linear Regression Equation

# Key Concepts of Machine Learning

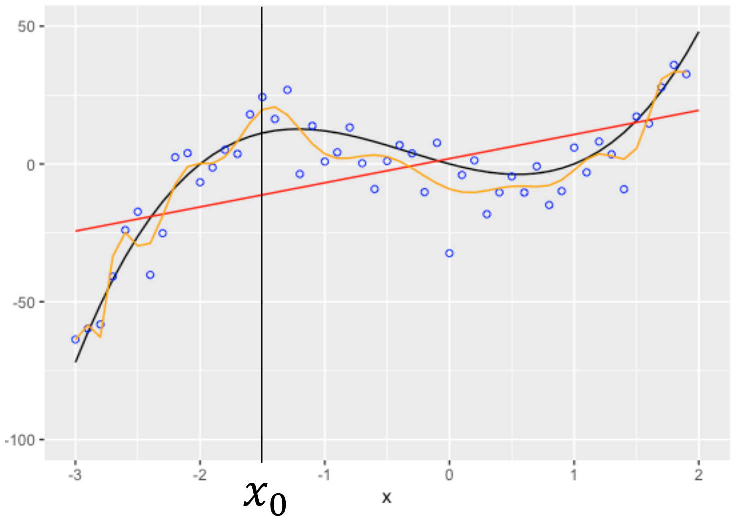
- Goal of regression with one feature:
  - Find a model $\hat{f}$ to predict the response ($y$) given the feature ($x$)
    - $\hat{f}$ is our estimate of the relationship between feature and response
    - E.g., a linear regression model $\hat{f}(x) = \hat{\beta}_1 x + \hat{\beta}_0$
    - Build $\hat{f}$ using the data
- Key concepts of machine learning regarding behavior of $\hat{f}$:
  - Bias-variance trade-off
  - Underfitting and overfitting

# Bias-Variance Trade-Off

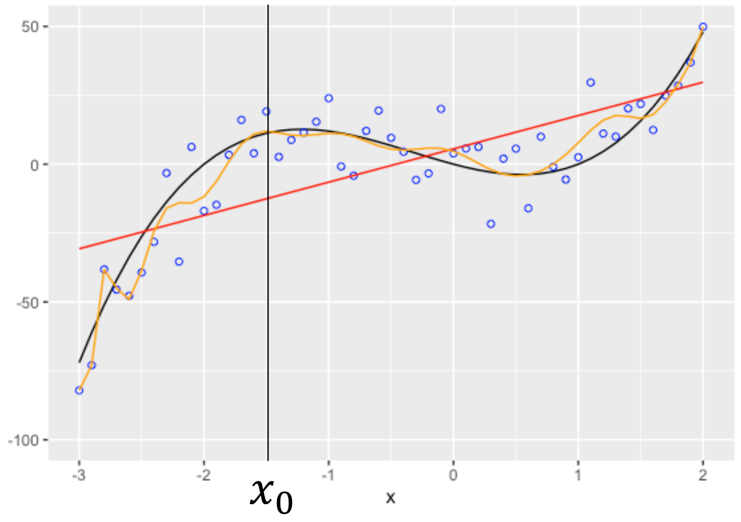
- Bias: difference between average of predictions and true value
- Variance: variability of predictions
- Want a model flexible enough for our problem
  - Too simple can lead to high bias
    - Model pays very little attention to the observations
    - Cannot capture the relationship between features and response
  - Too flexible can lead to high variance
    - Model pays too close attention to the observations → change in observations can lead to very different predictions
    - Model ends up trying to match the observations and does not generalize to new observations
- Typically, as flexibility increases, bias decreases and variance increases → this is the **bias-variance trade-off!**

# Bias-Variance Trade-Off

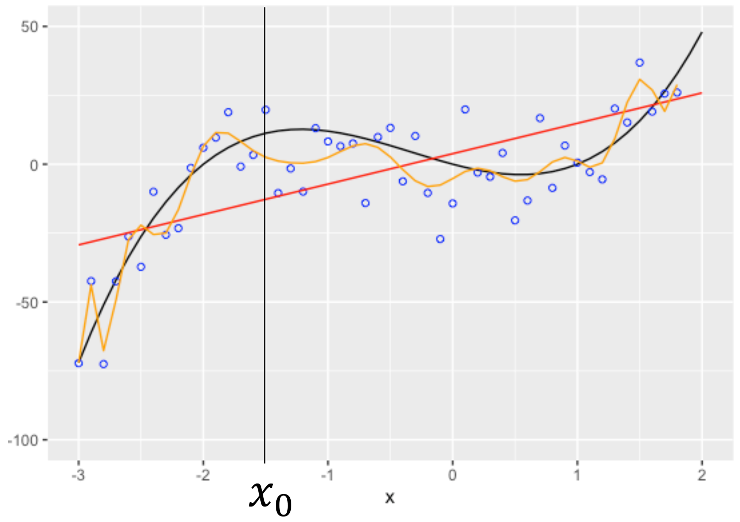
- Bias-variance trade-off
  - Assume the data are noisy observations (blue dots) of a polynomial (black line)
  - Use three independent datasets to build separate linear (red line) and high-order polynomial (orange line) models
  - Use the models to make a prediction at $x_0 = -1.5$




Dataset 1



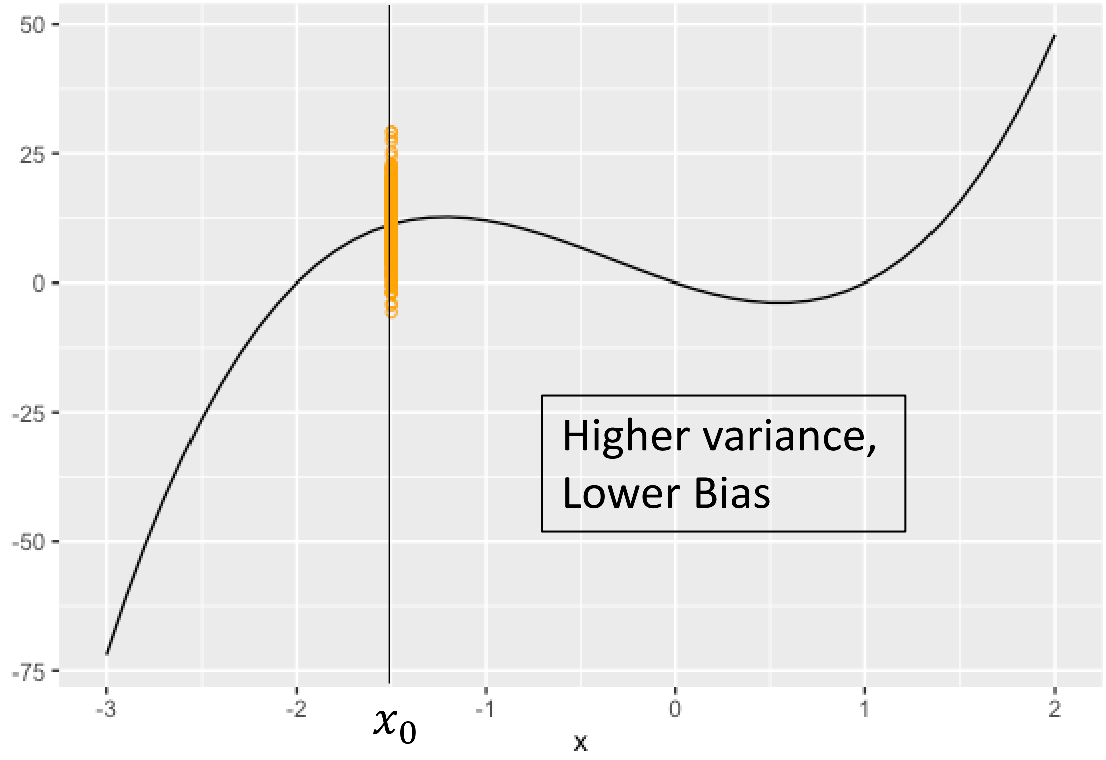

Dataset 2



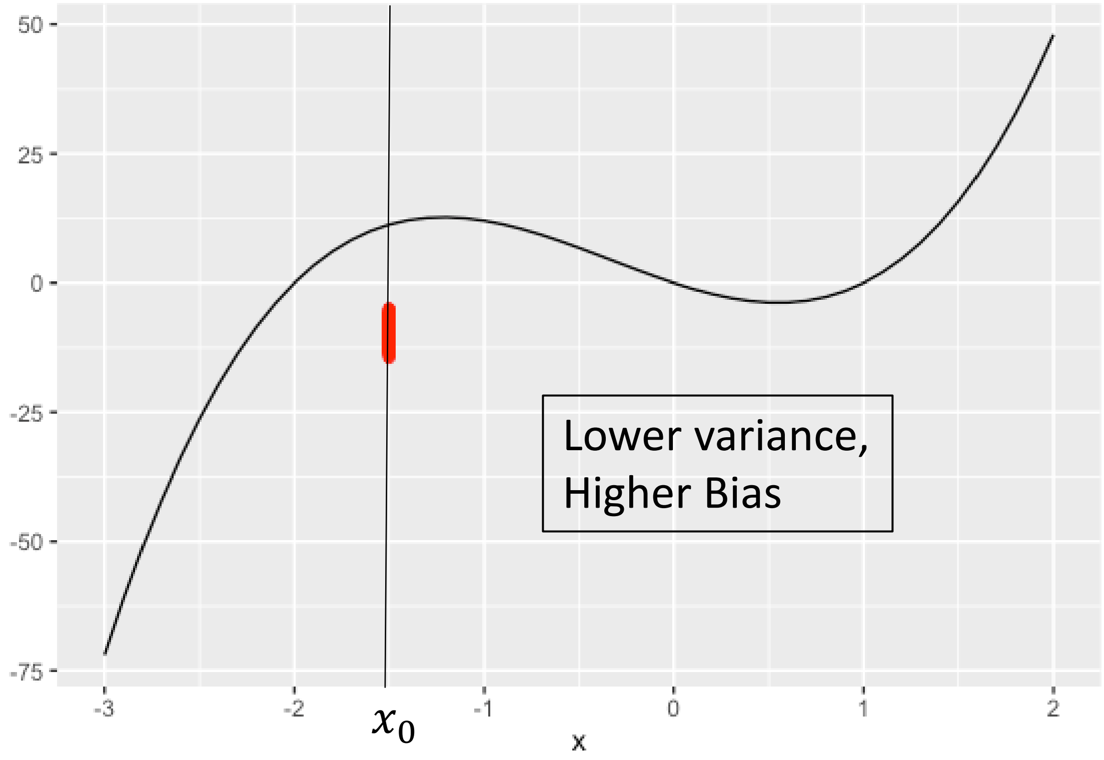

Dataset 3

# Bias-Variance Trade-Off

- Bias-variance trade-off
  - What if we used 500 independent datasets to build separate linear and high-order polynomial models and plotted their predictions at $x_0 = -1.5$



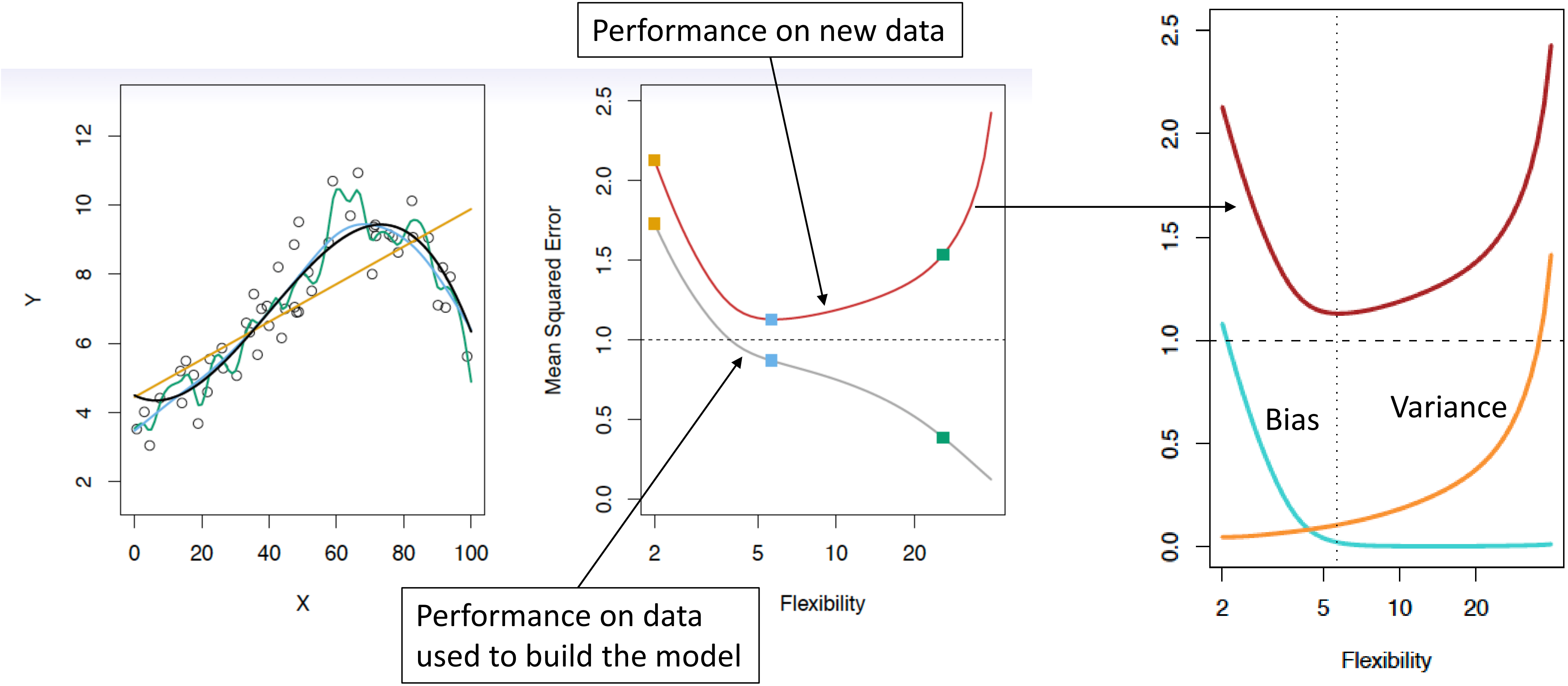

Predictions using High-Order Polynomial Models




Predictions using Linear Models

# Bias-Variance Trade-Off

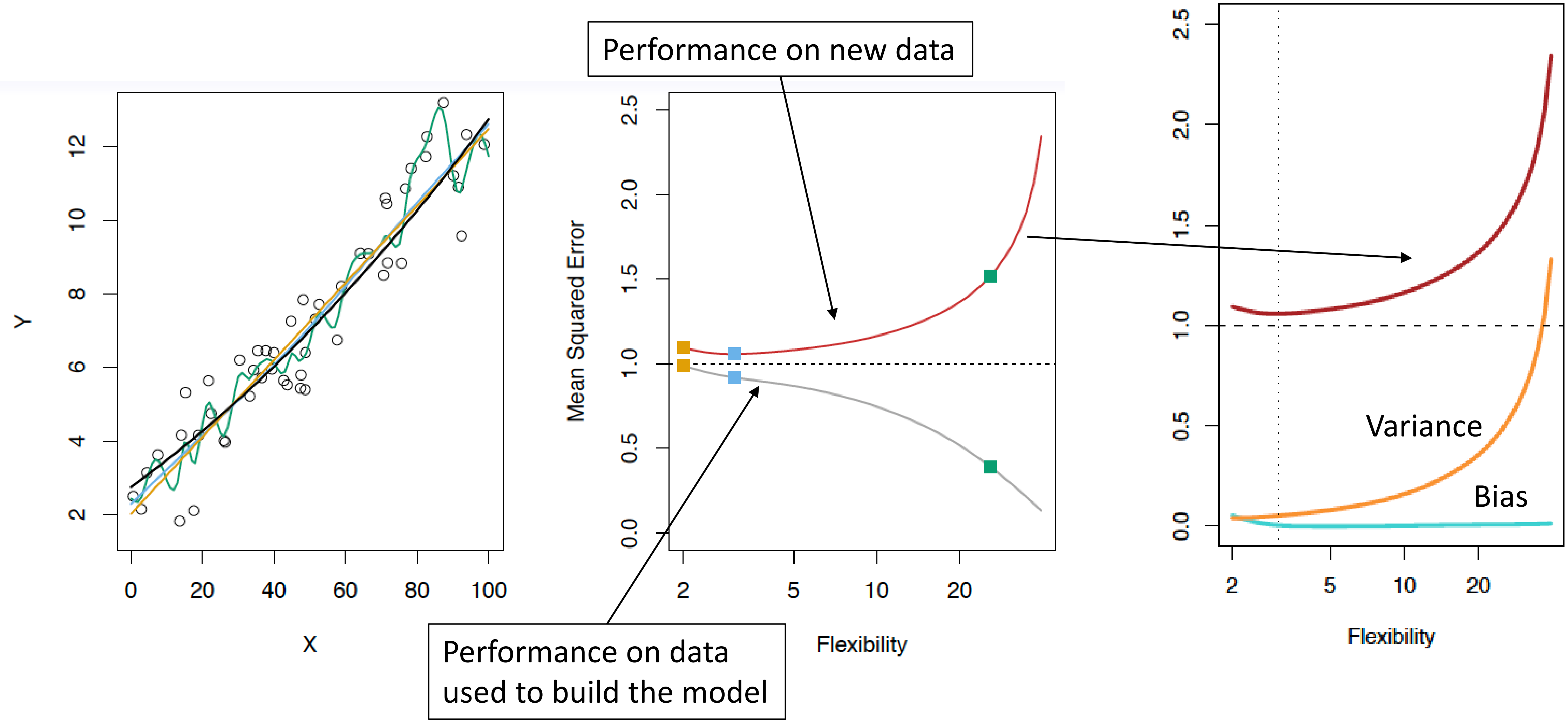
An example of the bias-variance trade-off




James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017) *An Introduction to Statistical Learning with Applications in R*. Springer

# Bias-Variance Trade-Off

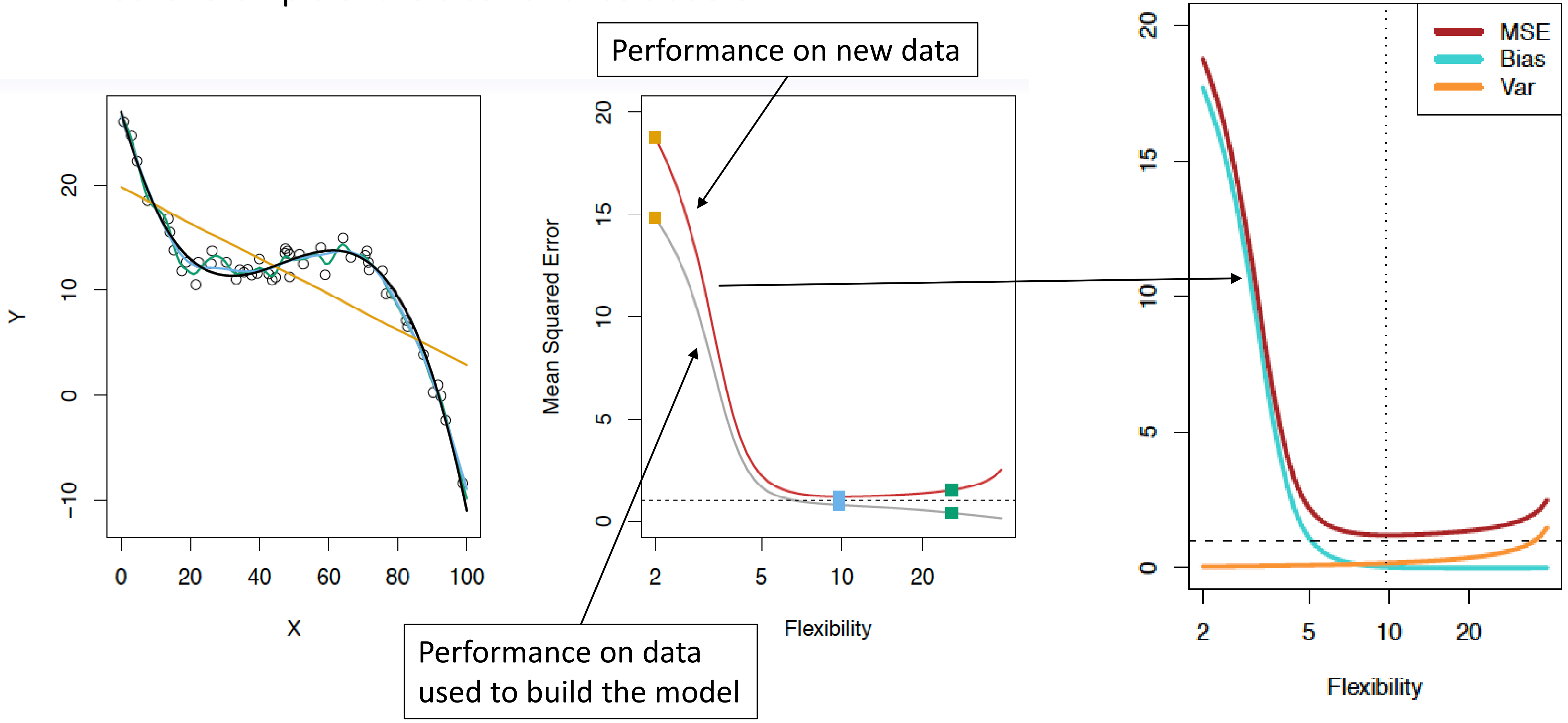
Another example of the bias-variance trade-off




James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017) *An Introduction to Statistical Learning with Applications in R*. Springer

# Bias-Variance Trade-Off

Another example of the bias-variance trade-off




James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017) *An Introduction to Statistical Learning with Applications in R*. Springer

# Bias-Variance Trade-Off

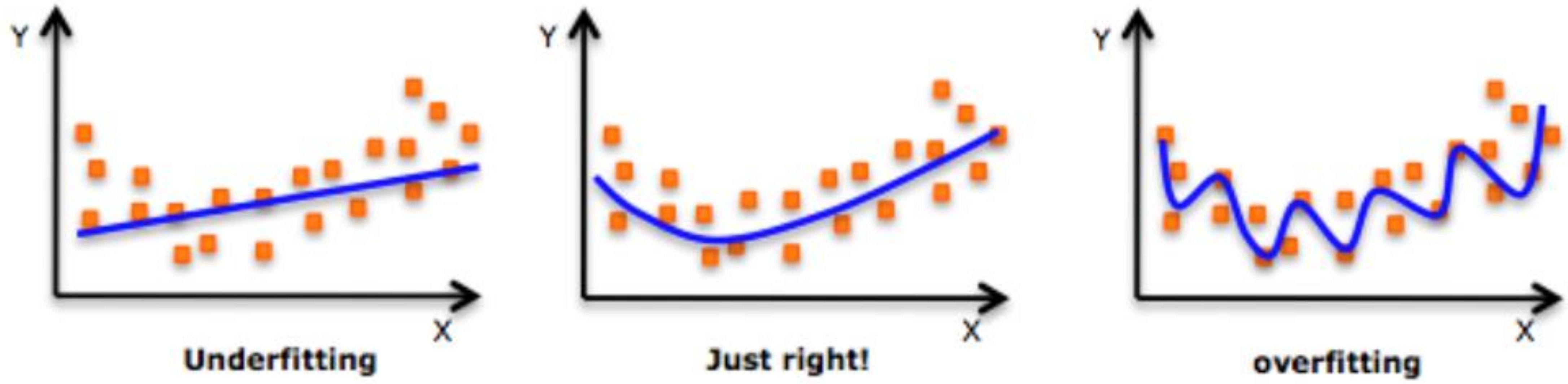
- Typically, as flexibility increases…
  - Bias decreases and variance increases
  - Interpretability decreases
- Knowing the application and purpose of the model is important!
  - If interpretability is not important, then it's not necessary to use an interpretable model

# Underfitting vs. Overfitting

- Underfitting and overfitting
  - Model that is too simple can lead to high bias
    - Model pays very little attention to the observations
    - Cannot capture the relationship between features and response
      - This is called ***underfitting***
  - Model that is too flexible can lead to high variance
    - Model pays too close attention to the observations → slight change in observations can lead to very different predictions
    - Model ends up trying to match the observations and does not generalize to new observations
      - This is called ***overfitting***
  - VERY IMPORTANT!

# Underfitting vs. Overfitting

Visualizing underfitting and overfitting in regression



# Classification

- The label (for classification, also called target) is a categorical variable with some number of levels called classes
- Want to predict the class for a new observation

# Classification

Assume we know the conditional probabilities

$$P(leaves|feature\ 1, feature\ 2)$$

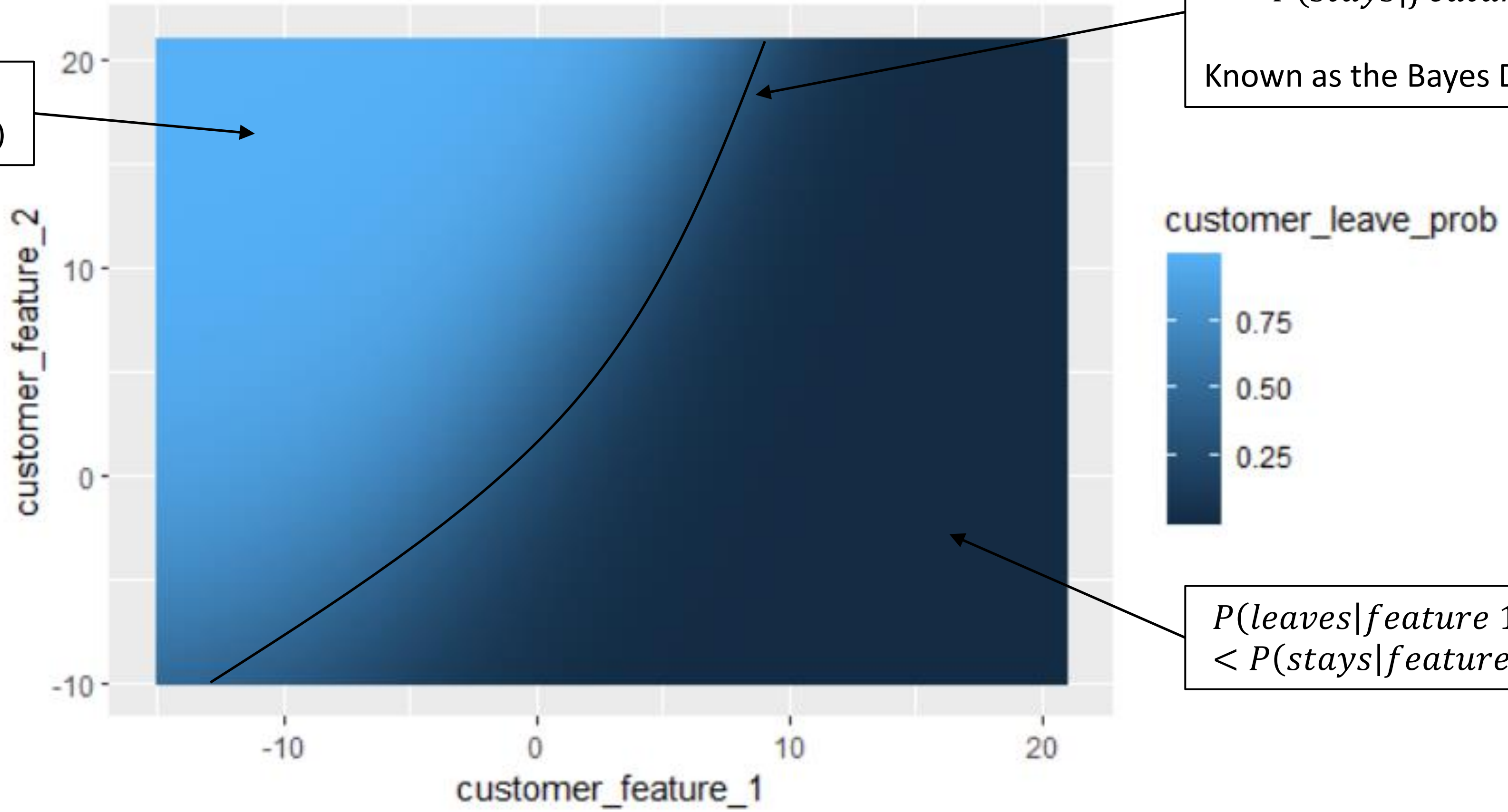
$$P(stays|feature\ 1, feature\ 2)$$

over the entire feature space

$P(leaves|feature\ 1, feature\ 2)$
$> P(stays|feature\ 1, feature\ 2)$

$P(leaves|feature\ 1, feature\ 2)$
$= P(stays|feature\ 1, feature\ 2)$

Known as the Bayes Decision Boundary




$P(leaves|feature\ 1, feature\ 2)$
$< P(stays|feature\ 1, feature\ 2)$

# Classification

## Assume we know the conditional probabilities

$$P(leaves|feature\ 1, feature\ 2)$$

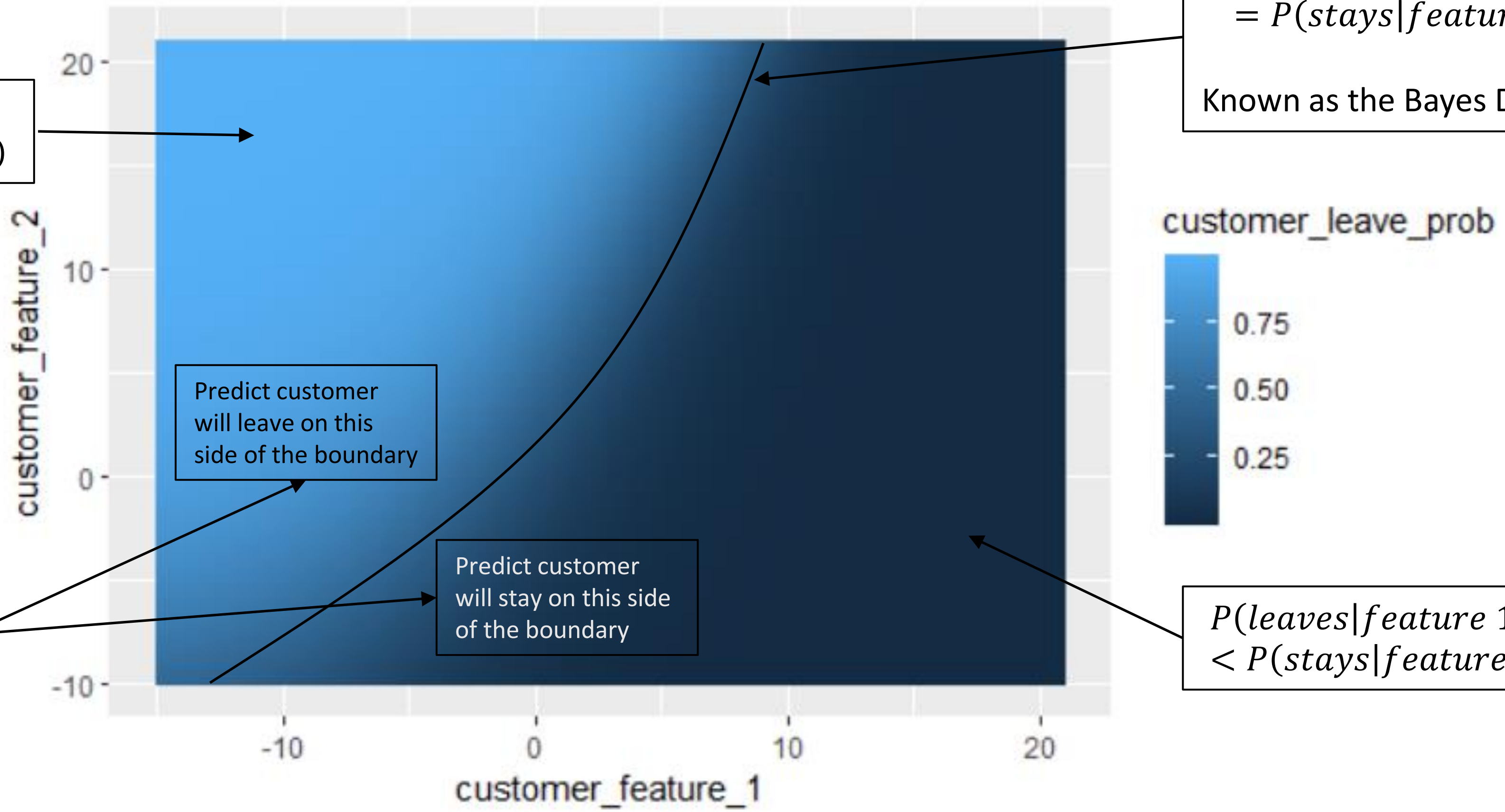
$$P(stays|feature\ 1, feature\ 2)$$

## over the entire feature space

$$P(leaves|feature\ 1, feature\ 2) = P(stays|feature\ 1, feature\ 2)$$

Known as the Bayes Decision Boundary

$$P(leaves|feature\ 1, feature\ 2) > P(stays|feature\ 1, feature\ 2)$$



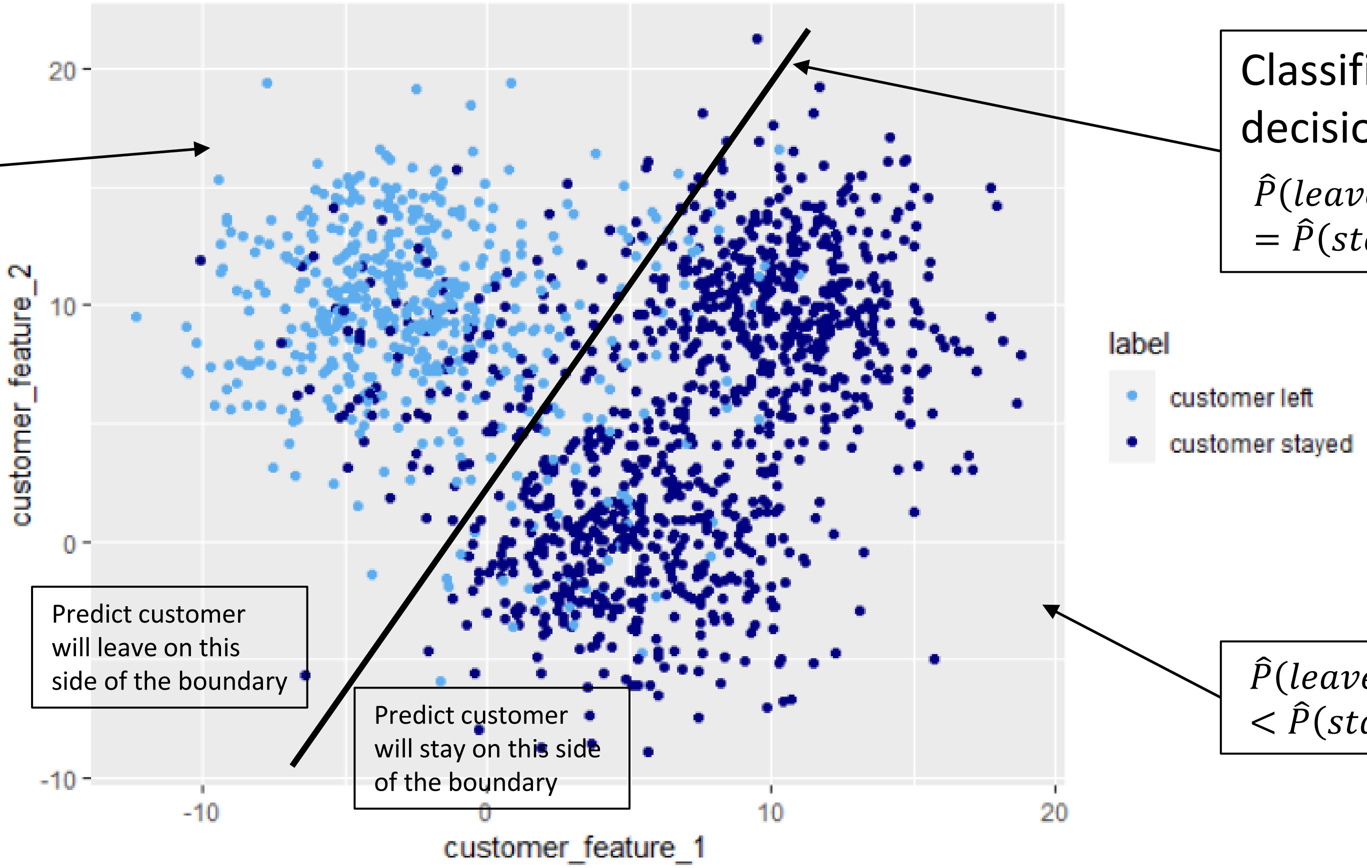

The Bayes Classifier

$$P(leaves|feature\ 1, feature\ 2) < P(stays|feature\ 1, feature\ 2)$$

# Classification

- In practice, we don't have this information, but we can:
  - Assume there is a conditional probability distribution over the feature space
  - Use a classifier to estimate the conditional probabilities
    - Note, now we have the estimated $\hat{P}$ instead of $P$

$\hat{P}(leaves|feature\ 1, feature\ 2)$
$> \hat{P}(stays|feature\ 1, feature\ 2)$




Classifier with a linear decision boundary

$\hat{P}(leaves|feature\ 1, feature\ 2)$
$= \hat{P}(stays|feature\ 1, feature\ 2)$

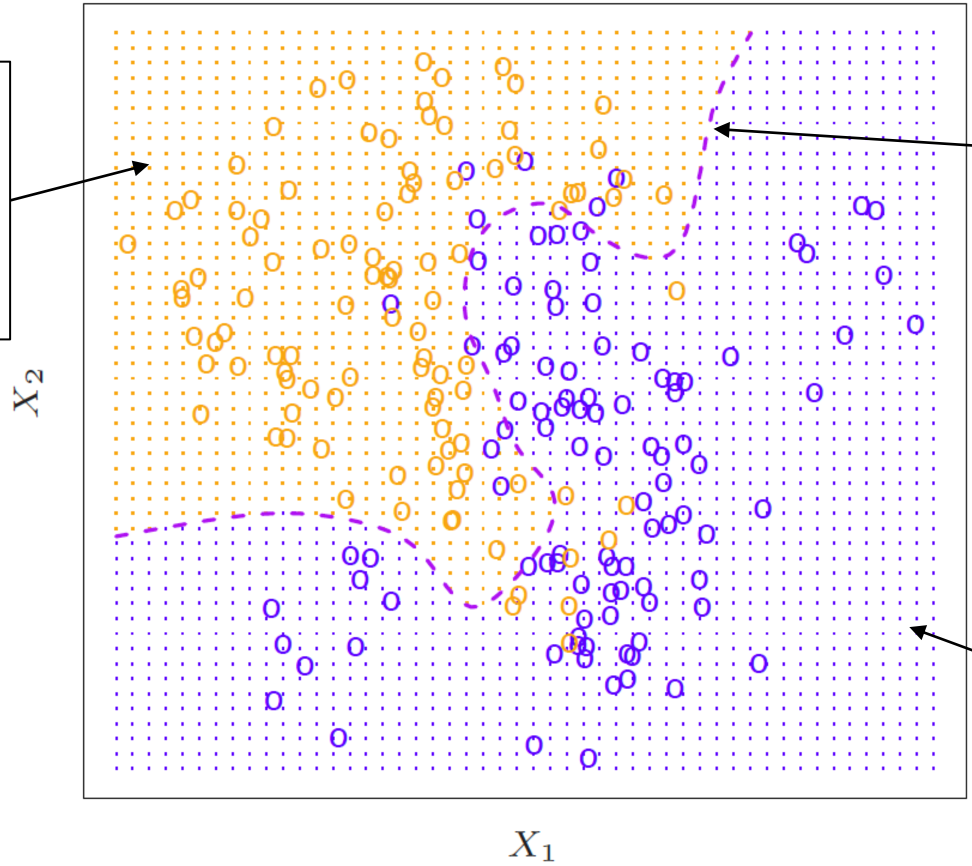
$\hat{P}(leaves|feature\ 1, feature\ 2)$
$< \hat{P}(stays|feature\ 1, feature\ 2)$

# Underfitting vs. Overfitting

## Visualizing underfitting and overfitting in classification: a two-dimensional example

Observations on this side of the boundary are more likely to be of the orange class

$P(purple|x_1, x_2) < P(orange|x_1, x_2)$




Bayes decision boundary between the two classes.

$P(purple|x_1, x_2) = P(orange|x_1, x_2)$

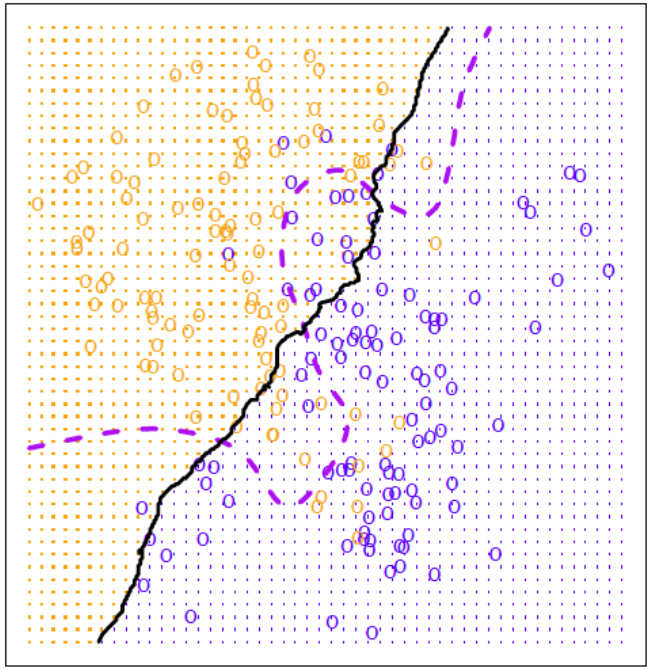
Observations on this side of the boundary are more likely to be of the purple class

$P(purple|x_1, x_2) > P(orange|x_1, x_2)$

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017) *An Introduction to Statistical Learning with Applications in R*. Springer

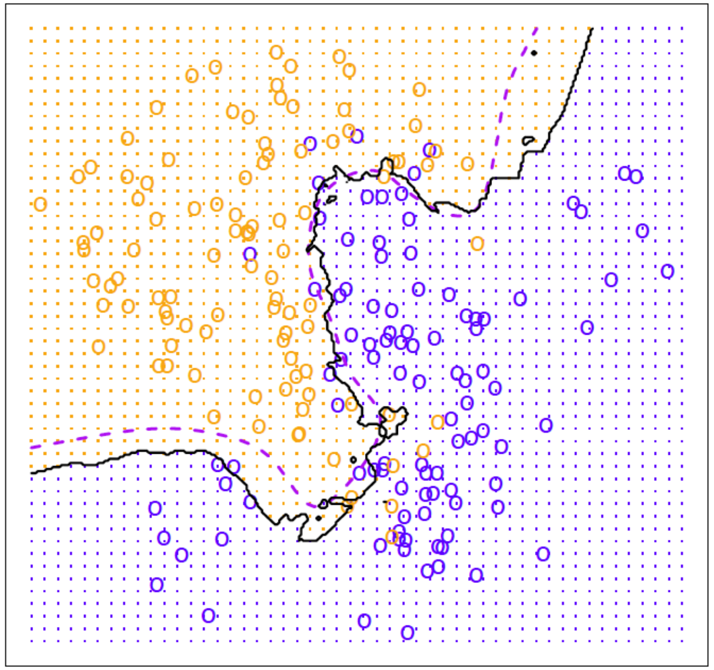
# Underfitting vs. Overfitting

## Visualizing underfitting and overfitting in classification: a two-dimensional example



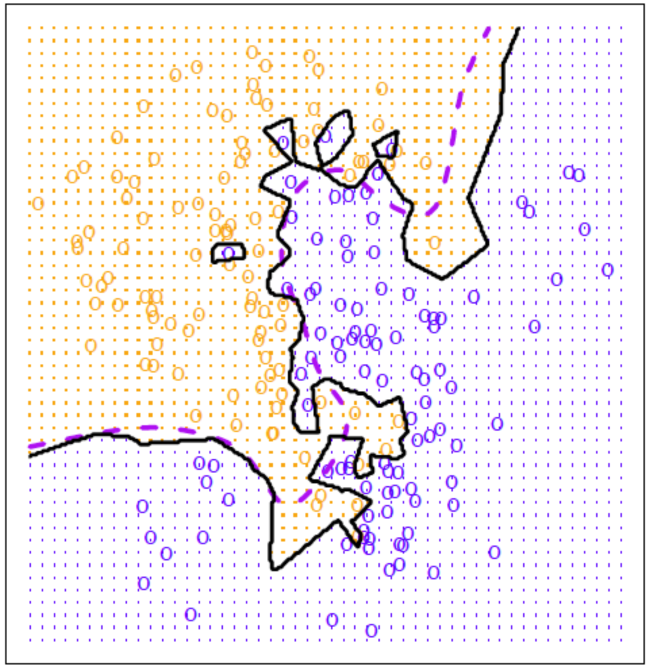
Using 100 nearest neighbors

Underfitting



Using 10 nearest neighbors

Just right!



Using 1 nearest neighbor

Overfitting

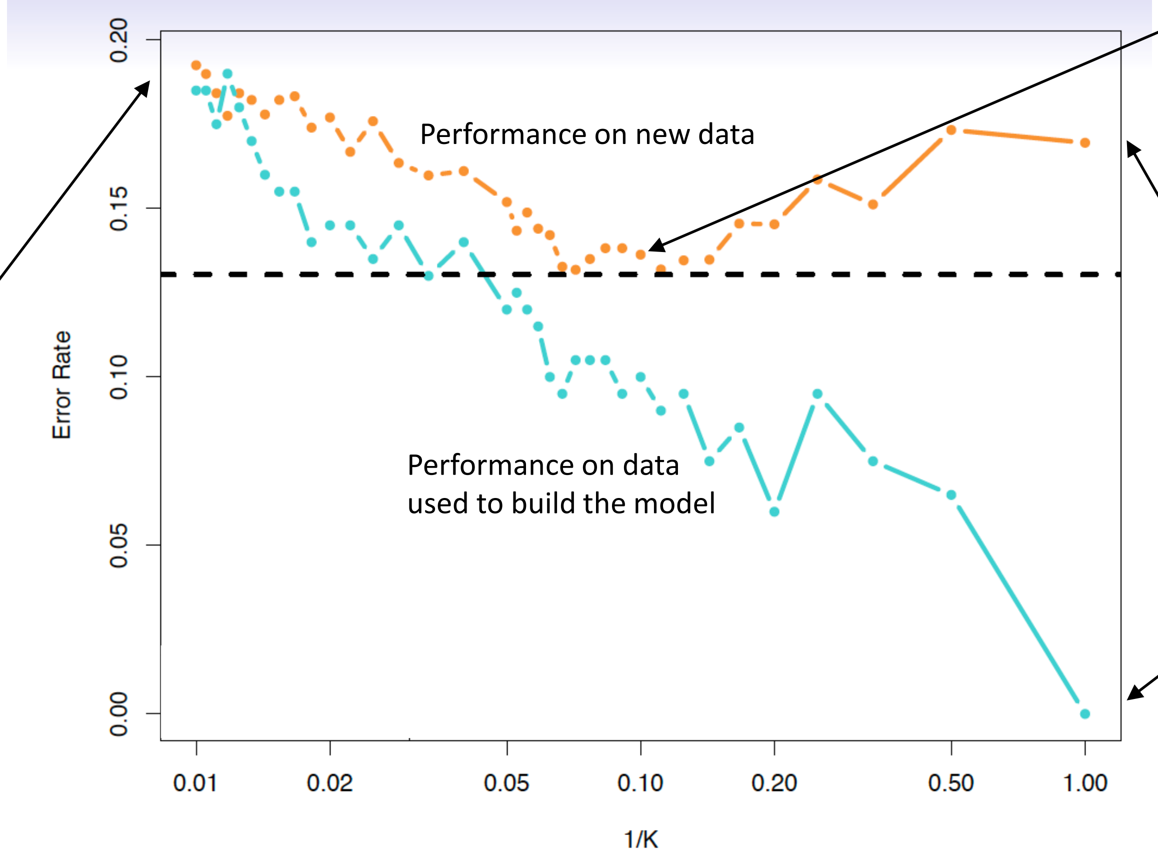
James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017) *An Introduction to Statistical Learning with Applications in R*. Springer

# Underfitting vs. Overfitting

How do we know when it's just right? Look for the characteristic inflection point!

**Just right**!

**Underfitting**: performance on both the data used to build the model and new data is poor




**Overfitting**: performance on data used to build the model gets better, but performance on new data gets worse

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017) *An Introduction to Statistical Learning with Applications in R*. Springer