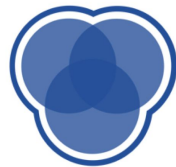# INFO 1998: Introduction to Machine Learning

**CDS Education**

We explore, learn, and educate big minds.

# Mid-Semester Feedback

- **How are we doing? Improvements/suggestions are welcome!**

https://forms.gle/D9hZayTLs3ZguQ4q9

# Lecture 6: Intro to Classifiers

**INFO 1998: Introduction to Machine Learning**

**CDS Education**

We explore, learn, and educate big minds.

# Agenda

1. **What is a Classifier?**
2. **K-Nearest Neighbors Classifier**
3. **Fit/Overfitting**
4. **Confusion Matrices**

# What are Classifiers?

# What are Classifiers?

**Classifiers are able to help answer questions like…**

- "What species is this?"

- "How would consumers rate this restaurant?"

- "Which Hogwarts House do I belong to?"

- "Am I going to pass this class?"

# What are Classifiers?

- Classifiers predict the class/category of a set of data points
- This class/category is based off of the target variable we are looking at
- Difference between linear regression and classifiers
  - Linear regression is used to predict the value of a **continuous variable**
  - Classifiers are used to predict **categorical or binary variables**

# What are Classifiers?

Two categories of classifiers: lazy learners and eager learners

- **Lazy Learners**
  - Store the training data and wait until a testing data appear
  - Classification is conducted based on the most related data in the stored training data
  - Less training time, more time in predicting
- **Eager Learners**
  - Construct a classification model based on the given training data before receiving data for classification
  - More training time, less time in predicting

# K-Nearest Neighbors Classifier

# What is the KNN Classifier?

- Lazy learner classifier
- Easy to interpret
- Fast to calculate
- Good for coarse analysis

# How Does It Work?

KNN is based off of a simple assumption that the values **nearest** to a data point are **similar** to it

Uses the k (a user specified value) nearest data points to predict the unknown one

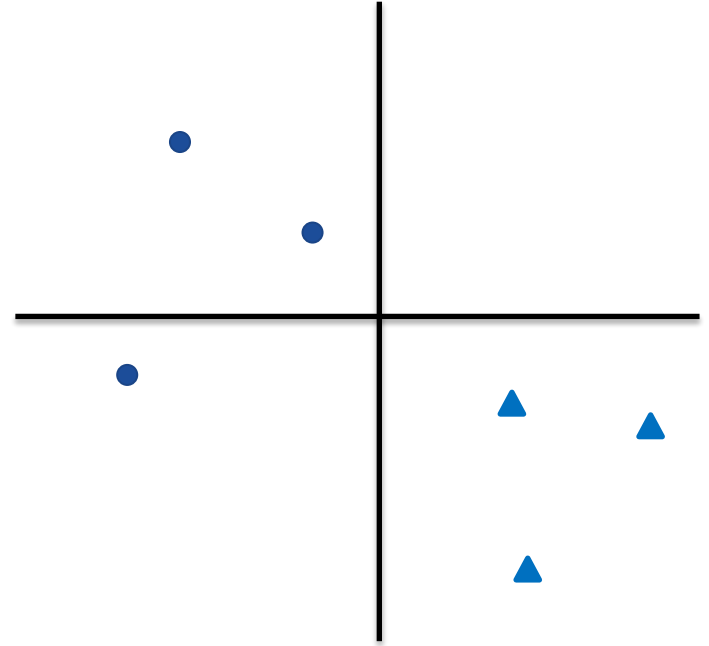k is a **hyperparameter** of the KNN model (a parameter that affects the learning process)!

# How Does It Work?

# How Does It Work? (Step-By-Step Example)
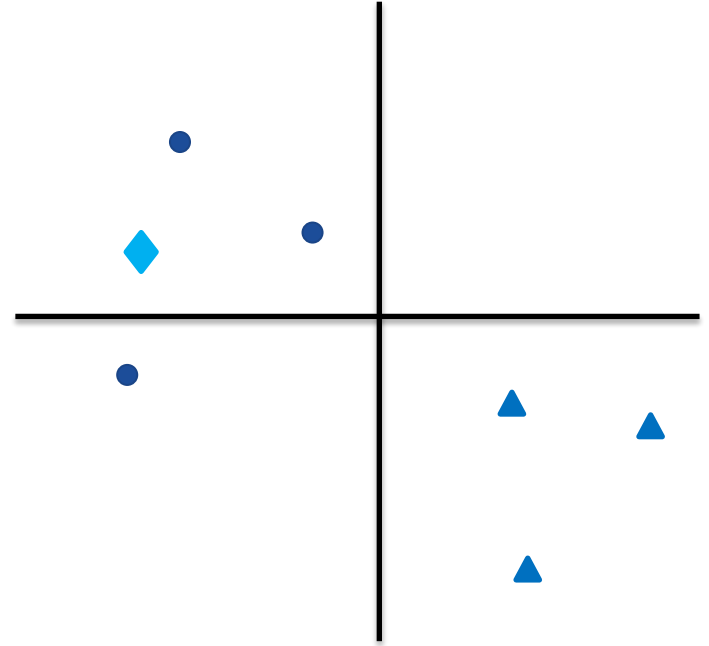
**Define** a *k* value (in this case k = 3)

# How Does It Work? (Step-By-Step Example)

**Define** a *k* value (in this case k = 3)

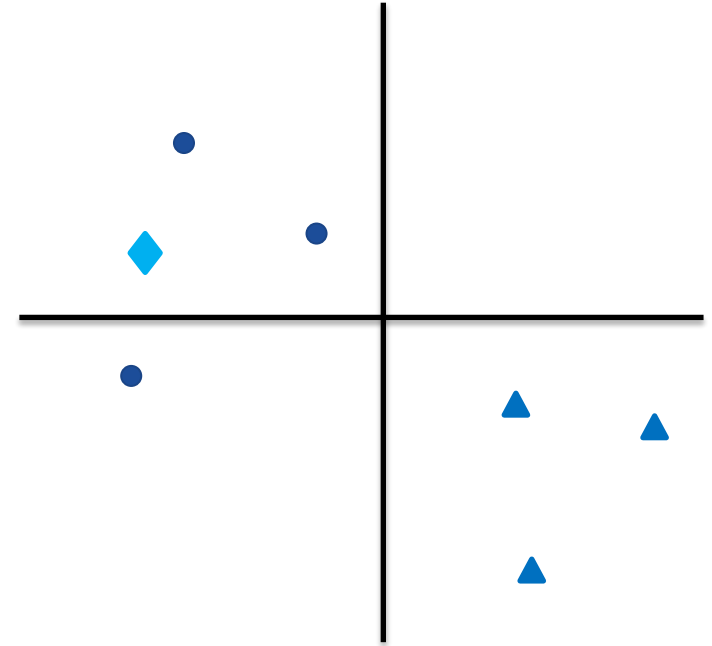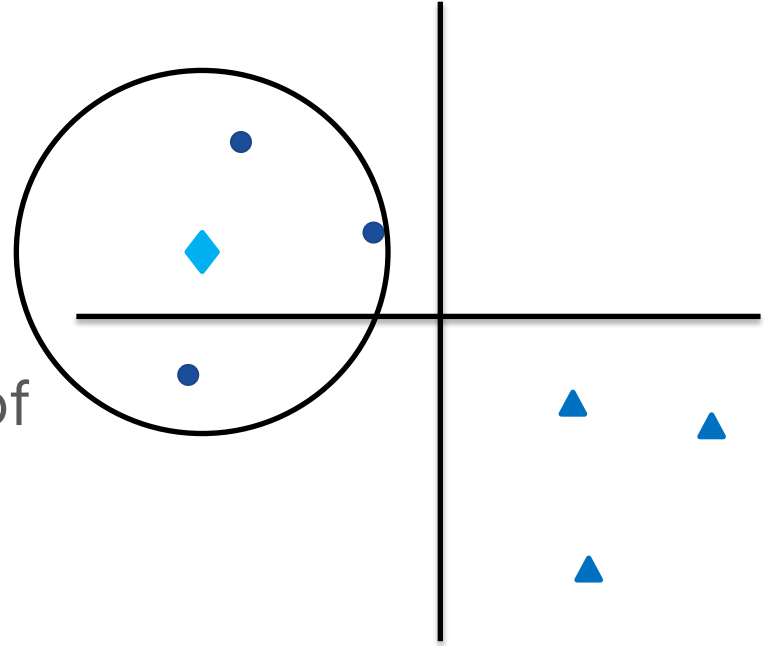**Pick** a point to predict (blue diamond)

# How Does It Work? (Step-By-Step Example)

**Define** a *k* value (in this case k = 3)

**Pick** a point to predict (blue diamond)

**Count** the number of closest points

# How Does It Work? (Step-By-Step Example)

**Define** a *k* value (in this case k = 3)

**Pick** a point to predict (blue diamond)

**Count** the number of closest points

**Increase** the radius until the number of points within the radius adds up to 3

● 3/3
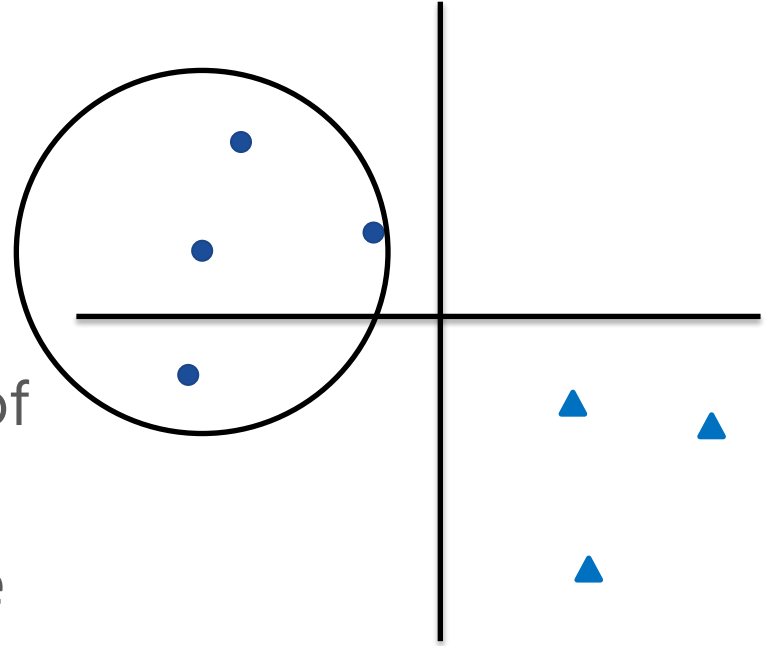
▲ 0/3

# How Does It Work? (Step-By-Step Example)

**Define** a *k* value (in this case k = 3)

**Pick** a point to predict (blue diamond)

**Count** the number of closest points

**Increase** the radius until the number of points within the radius adds up to 3

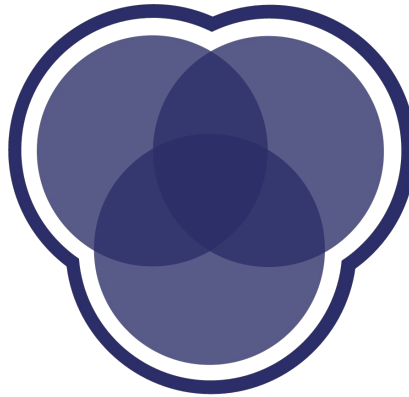**Predict** the blue diamond to be a blue circle!

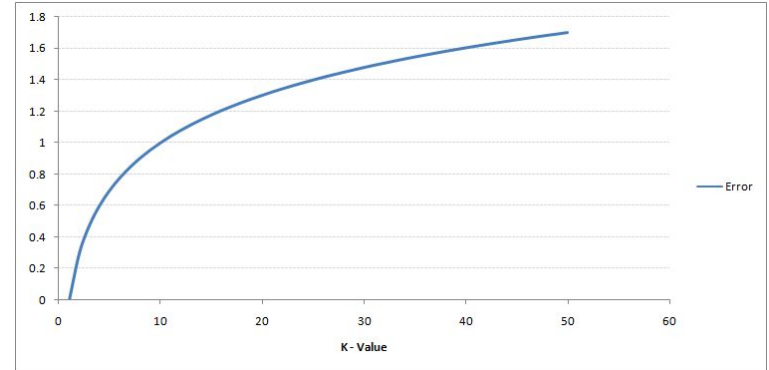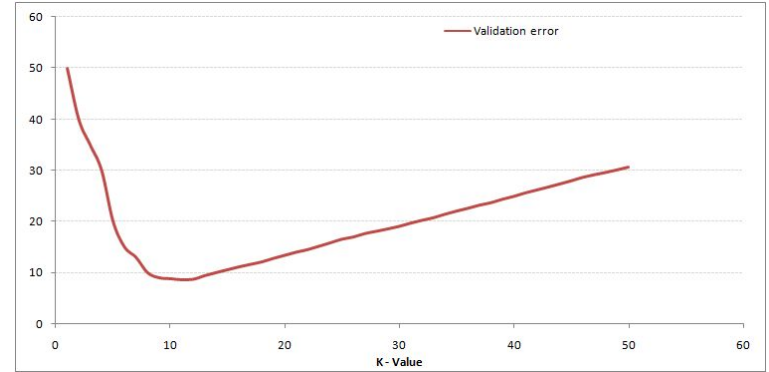● 3/3

▲ 0/3

# Demo

# Fit/Overfitting

# Overfitting

When the model corresponds too closely to training data and then isn't transferable to other data.
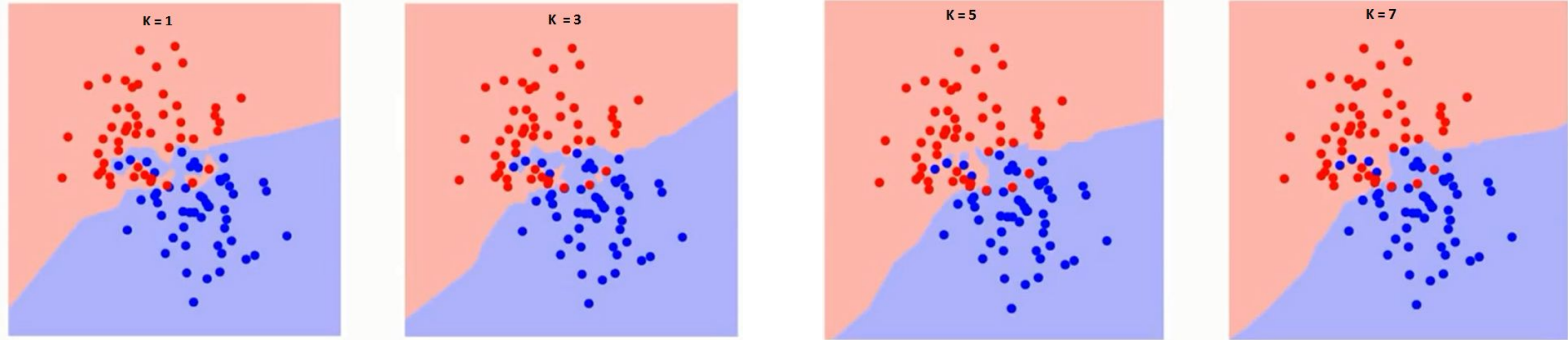
Can fix by:
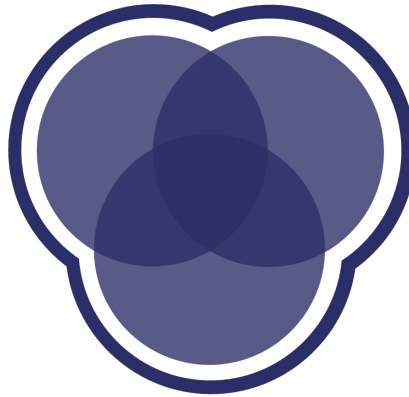- Splitting data into training and validation sets
- Increasing k

# Relationship Between k and Fit

The **k** value you use has a relationship to the fit of the model

A higher k gives a smoother line, but too large of a k and it is the average of all the data (or the label that is most common/likely)

# Demo

# Confusion Matrix

# What is a Confusion Matrix?

Table used to describe the performance of a classifier on a set of binary test data for which the true values are known

|  | p' (Predicted) | n' (Predicted) |
|---|---|---|
| P (Actual) | True Positive | False Negative |
| n (Actual) | False Positive | True Negative |

# Sensitivity

Called the **true positive rate**

Tells us how many positives are correctly identified as positives

**Optimize for:** airport security & initial diagnosis of fatal disease

**Sensitivity** = True Positive/
(True Positive + False Negative)

# Specificity

Called the **true negative rate**

Tells us how many negatives are correctly identified as negatives

**Optimize for:** testing for a disease with a risky treatment

**Specificity** = True Negative/
(True Negative + False Positive)

## Question

Which is an example of when you would want **higher specificity**?

- DNA tests for a death penalty case
- Deciding which iPhone to buy
- Airport security

# Overall Accuracy

Proportion of correct predictions

**Accuracy** =
(True Positive + True Negative) / Total

# Overall Error Rate

Proportion of incorrect predictions

**Error** =
(False Positive + False Negative) / Total

# Precision

Proportion of correct positive predictions among all positive predictions

**Precision** = True Positive /
(True Positive + False Positive)

# Coming Up

- **Assignment 6**: Due at 5:30pm on March 30th, 2022

- **Mid-Semester Check-In:** Starts next week! More details on ED Discussion!

- **Feedback Survey:** Please fill it out!

- **Next Lecture**: Applications of Supervised Learning

**CDS Education**
We explore, learn, and educate big minds.