K-Nearest Neighbors

Sauman Das
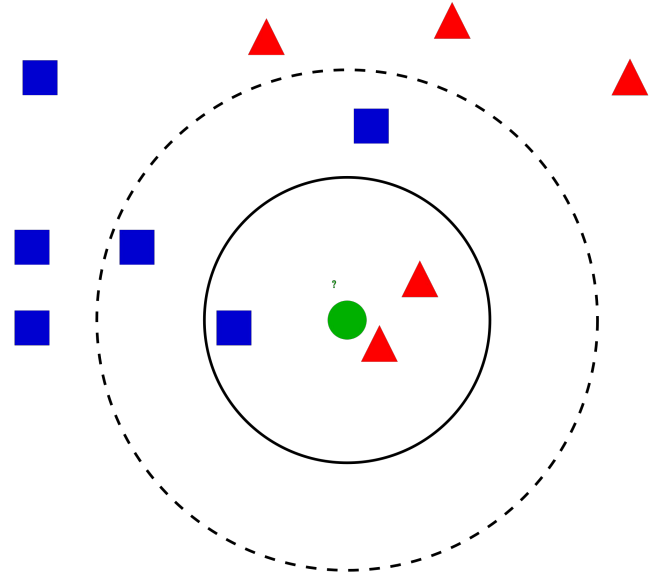
Overview

- KNN Algorithm

- Pros and Cons

- Application

Classification Algorithm

- Given test instance *x*

  - Identify *k* closest points to *x* from the training set

  - Identify classes of the *k* closest points

  - Predicted class is the most common of the *k* classes

Did we create a model?

Lazy Learner

- No actual "model" is created

- Use the entire training dataset for every test instance

  - How does this compare to decision tree (which is actually a model)?

Pros

- Very simple model
  - Try implementing from scratch!
- No training time
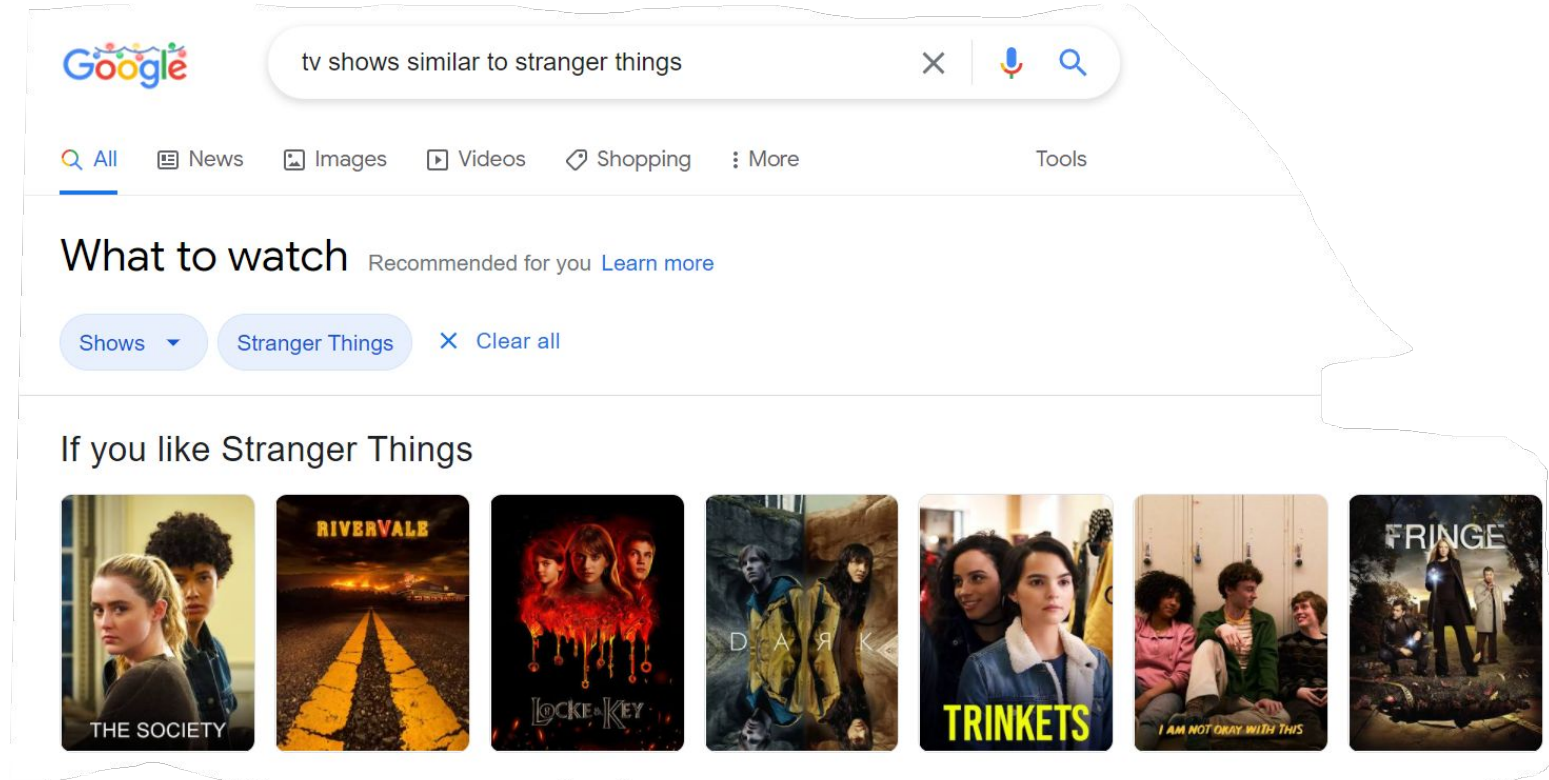 - Enables us to adaptively increase training data

Cons

- Large amount of memory

    - Stores entire training dataset!

- Runtime for each test case is large

- Simple algorithm could lead to poor results

Recommender Systems

Recommender System

- Each movie has associated features

- If you want 3 of the most similar movies, run KNN with *k*=3

- Output is the closest neighbors
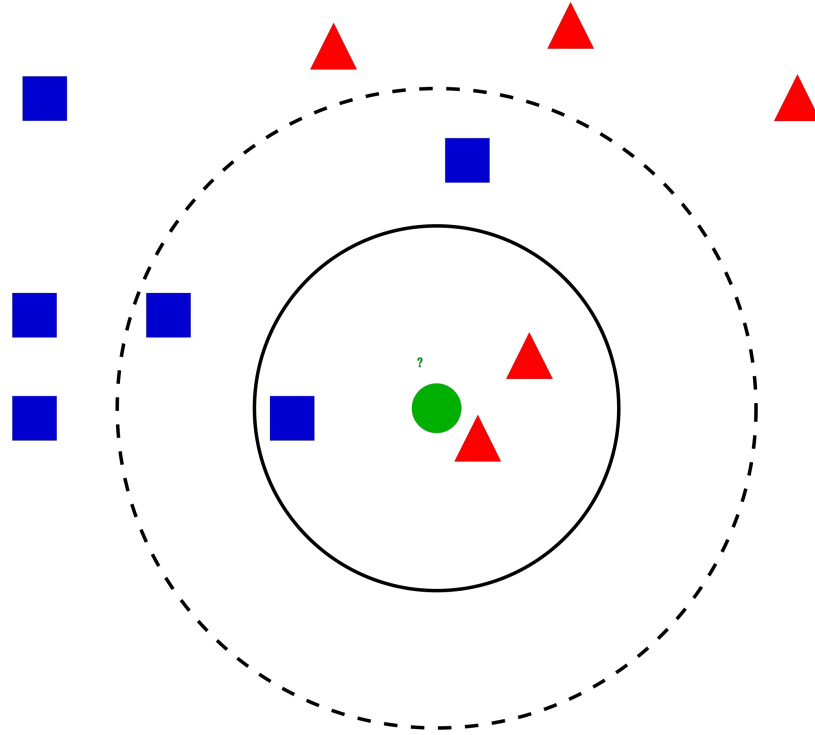 - Not an actual class value in this case

Recommender System

- KNN is useful in this scenario

- As new movies are added frequently to the
  database, we don't retrain the model

- Works as long as there is a database of
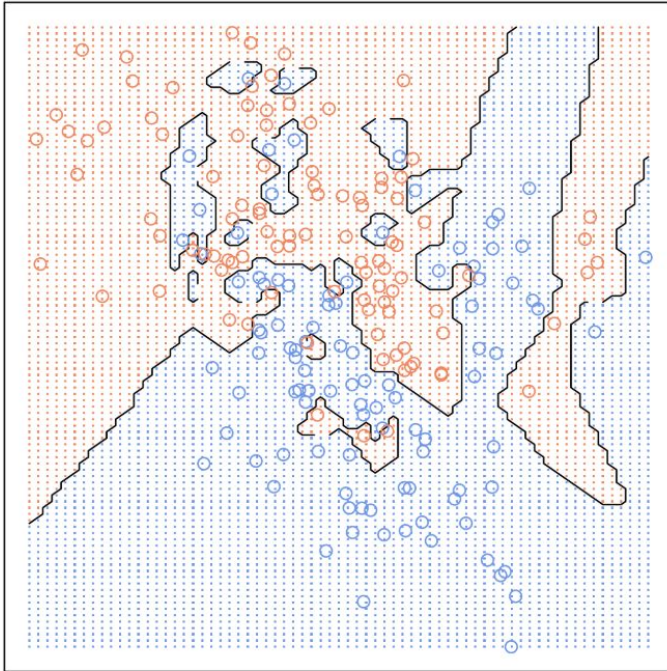 movies (training set)

Overfitting

Increasing *k*

1-nearest neighbours

20-nearest neighbours