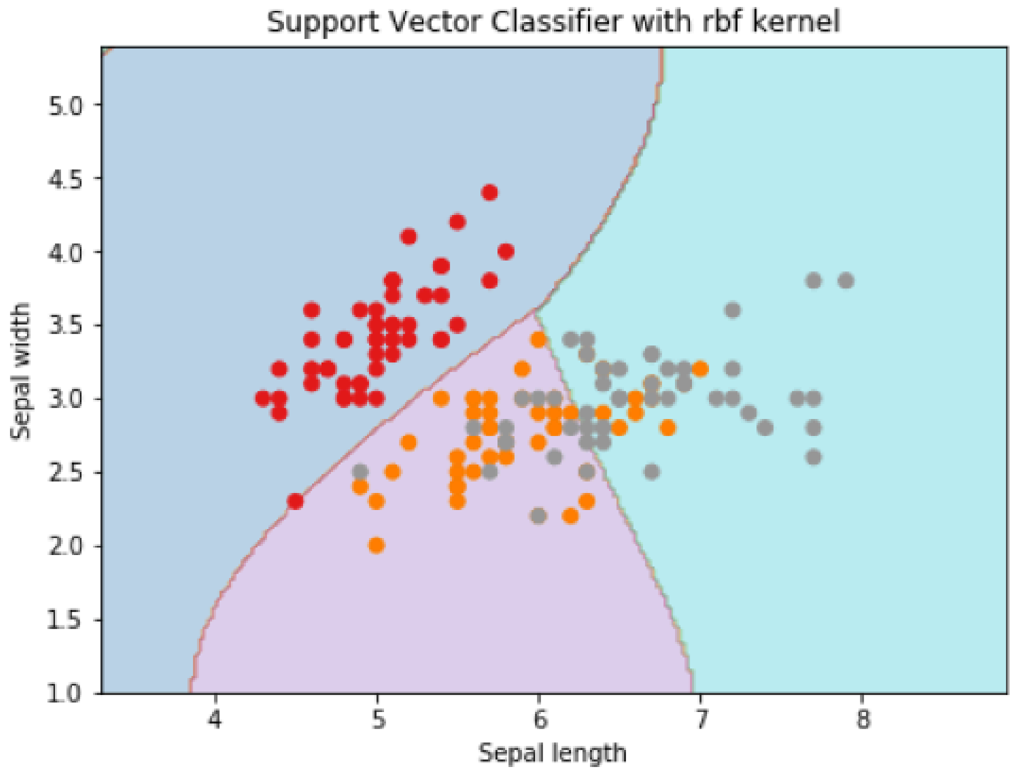
Understanding Artificial Intelligence

Day Three

Classifiers



Overview

- Classifiers are used to predict what **category** a given data point belongs to
  - Spam email, not spam email
  - Species of animal
- Very dependent upon domain, no one-size-fits-all

Uses

Many things can be formulated as a classification problem:

- Email spam detection
- Filtering candidates for hiring
- Loan application decisions
- Credit scoring
- Handwriting recognition

Algorithms

There are many classification algorithms, including:

- Decision trees
- K-nearest neighbors
- Naive Bayes
- Support vector machines
- Neural networks

Approach: K-Nearest Neighbors

- Start with a training data set where each data point is labeled with a category
- Compare a new piece of data to the k nearest data points
- Take the category that is in the majority of those k

Interactive example:

https://tinyurl.com/knn-demo

http://vision.stanford.edu/teaching/cs231n-demos/knn/

K-Nearest Neighbors

- Generally very accurate
- Insensitive to outliers
- No assumptions about your data
- Simple to implement and understand
- Computationally expensive[1]
- Needs extra tuning if classes are skewed

[1] Specialized storage, like vector databases, are required for good scaling and performance on large datasets

Accuracy

In the previous slide, we said that the algorithm is generally very **accurate**. Accuracy for classifiers is oftem measured using a **confusion matrix**.

Confusion Matrix

	Predicted Negative	Predicted Positive
Actual Negative	True Negative	False Positive
Actual Positive	False Negative	True Positive

Confusion Matrix: Example

Hypothetical classifier for “cancer” (1) or “no cancer” (0):

Individual Number	1	2	3	4	5	6	7	8	9	10	11	12
Actual Classification	1	1	1	1	1	1	1	1	0	0	0	0
Predicted Classification	0	0	1	1	1	1	1	1	1	0	0	0

	Predicted Negative	Predicted Positive
Actual Negative		
Actual Positive		

Accuracy

There are many accuracy measurements. A basic one is:

$$\text{Accuracy} = \frac{\text{True positives} + \text{True negatives}}{\text{Total}}$$

It's unwise to rely on this alone. Consider the case when 95% of the data set does not have cancer - classifying **everyone** as negative for cancer would give a 95% accuracy.

F_1 Score

Good for when there's class imbalance and false negatives (FN) are especially undesirable:

$$F_1 = \frac{2 * \text{True positives}}{2 * \text{True positives} + \text{False positives} + \text{False negatives}}$$

φ Coefficient (MCC)

φ (phi) or Matthews correlation coefficient (MCC), good general alternative to plain accuracy, especially for impalanced classes.

$$\varphi = \mathrm{MCC} = \frac{\mathrm{TP} * \mathrm{TN} - \mathrm{FP} * \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}}$$

Group Exercise

- Read the provided article
- Split into groups and discuss:
  - How was a classifier likely used in this situation?
  - How did bias in the data used to train the classification model manifest in the output?
  - How could discrimination be detected and avoided?
- We'll then regroup and discuss as a class

Bayesian Statistics

Figure 2: The Reverend Thomas Bayes (maybe)

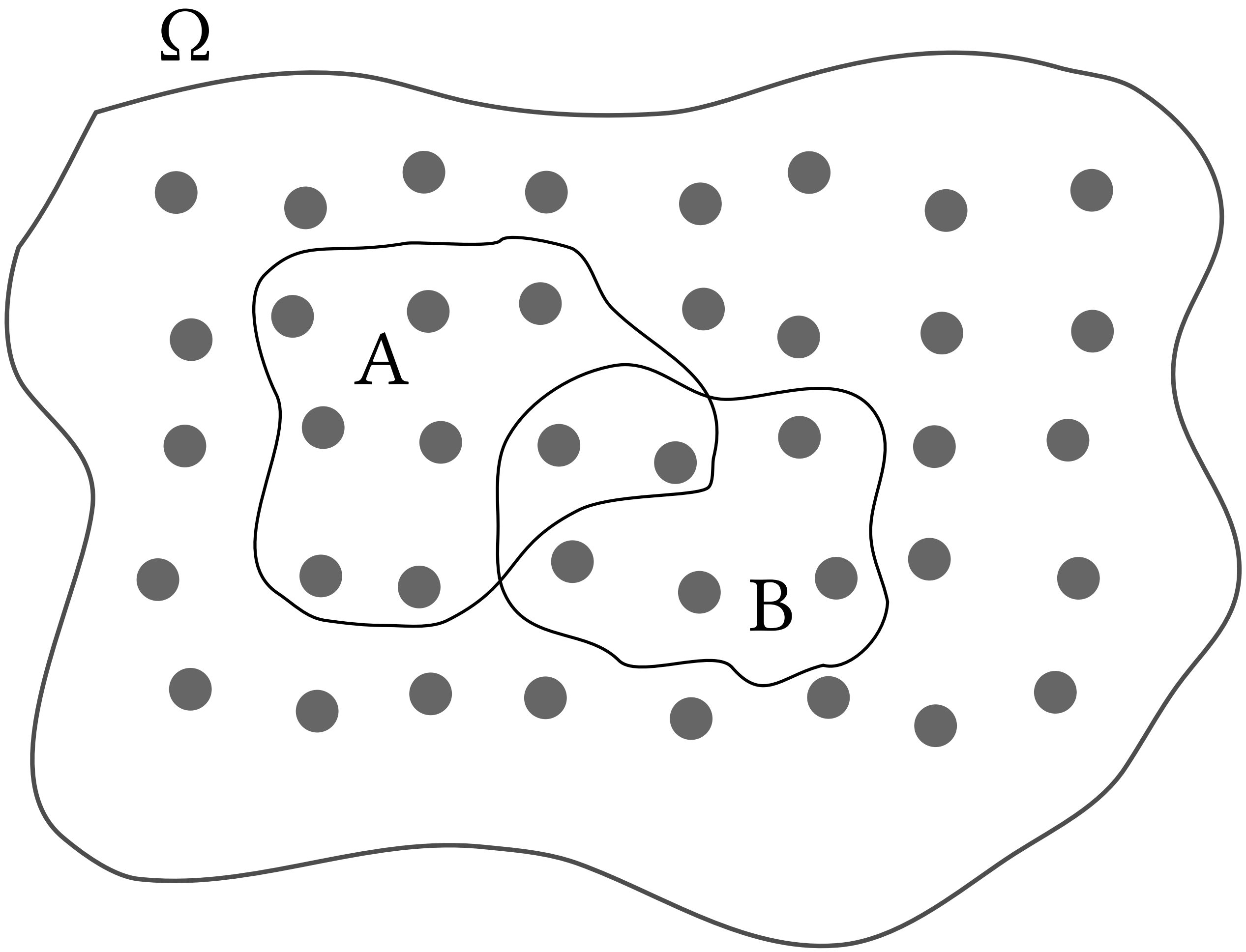
Overview

- **Bayesian Statistics** is based on the interpretation of probability as a *degree of belief* in an event.
- This is different than **frequentist statistics** which interprets probability as how often an event happens if the situation were to occur many times in a row.
- Useful for AI and machine learning because you can update your best guess when new data arrives.

Bayes' Theorem

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

where $P(B) \neq 0$.



$P(A) = \ldots$

$P(B) = \ldots$

$P(A|B) = \ldots$

$P(B|A) = \ldots$

$\frac{P(B|A)P(A)}{P(B)} = \ldots$

$\frac{P(A|B)P(B)}{P(A)} = \ldots$

40 total events

Bayes' Theorem: Everyday Example

- Suppose you think there's an 80% chance the belief "my friend is mad at me" is true.
- You get some new information: Your friend texted you to hang out.
- How does your *degree of belief* change?

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$$\begin{aligned}
P(A) &= 80\% = \text{Prior probability} \\
P(B) &= 90\% = \text{Marginal probability} \\
P(B|A) &= \text{Odds friend texted you if they are mad} \\
&= 40\% \\
P(A|B) &= \tfrac{40\% * 80\%}{90\%} \approx 35.6\% \Rightarrow \text{Friend is probably not mad}
\end{aligned}$$

Classifiers

Approach: Naive Bayes

- Utilize Bayes' Theorem to calculate the **probability** that a data point is one of multiple categories
- Choose the category with the highest probability

Naive Bayes

#	a_1	a_2	y
1	Yes	Yes	Yes
2	Yes	No	Yes
3	Yes	Yes	No
4	No	Yes	Yes
5	Yes	No	No

$$\text{Predicted } y = \hat{y} = \max_y p(y|\boldsymbol{x})$$

$$p(y|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|y)p(y)}{p(\boldsymbol{x})} \propto p(\boldsymbol{x}|y)p(y)$$

$$p(y) = \frac{\#\text{ of } y}{\text{Total}} = \frac{|y|}{|X|}$$

$$\hat{y} = \max_y \frac{|y| * p(\boldsymbol{x}|y)}{|X|}$$

Naive Bayes

$p(\boldsymbol{x}|y)$ is not easy to estimate. We make a **naive** assumption, that all attributes are **independent**.

In probability, if two events A and B are independent, then:

$$p(A|B) = p(B|A) = p(A) = p(B)$$

$$p(A \text{ and } B) = p(A) * p(B)$$

Naive Bayes

A data point is a vector of events combined with an "and":

$$\begin{aligned}
\boldsymbol{x} &= (a_1, a_2, ..., a_n) \\
p(\boldsymbol{x}) &= p(a_1 \text{ and } a_2 \text{ and } ... \text{ and } a_n) \\
&= p(a_1) * p(a_2) * ... * p(a_n)
\end{aligned}$$

Naive Bayes

If probabilities are **conditioned** on another event, y, that ends up getting applied to all the event probabilities:

$$
\begin{aligned}
p(\boldsymbol{x}|y) &= p((a_1, a_2, \ldots, a_n)|y) \\
&= p(a_1|y) * p(a_2|y) * \ldots * p(a_n|y) \\
&= \prod_{i=1}^{n} p(a_i|y)
\end{aligned}
$$

Naive Bayes

$$\hat{y} = \max_{y} \frac{|y|}{|X|} p(\boldsymbol{x}|y) = \max_{y} \frac{|y|}{|X|} * \prod_{i=1}^{n} p(a_i|y)$$

Estimating $p(a_i|y)$ is easy:

$$p(a_i|y) = \frac{\text{\# of data points with attribute } a_i \text{ and } y}{\text{Total number of data points with } y}$$

Example

#	a_1	a_2	y
1	Yes	Yes	Yes
2	Yes	No	Yes
3	Yes	Yes	No
4	No	Yes	Yes
5	Yes	No	No

A new data point has (a_1, a_2) and we are predicting if y = Yes or y = No.

#	a_1	a_2	y
1	Yes	Yes	Yes
2	Yes	No	Yes
3	Yes	Yes	No
4	No	Yes	Yes
5	Yes	No	No

	y = Yes	y = No
$p(a_1 = \text{Yes}\|y)$	2/3	1
$p(a_1 = \text{No}\|y)$	1/3	0
$p(a_2 = \text{Yes}\|y)$	2/3	1/2
$p(a_2 = \text{No}\|y)$	1/3	1/2

Consider: $x_{\text{new}} = (a_1 = \text{Yes}, a_2 = \text{No}) = (\text{Yes}, \text{No})$

	y = Yes	y = No
$p(a_1 = \text{Yes}\|y)$	2/3	1
$p(a_1 = \text{No}\|y)$	1/3	0
$p(a_2 = \text{Yes}\|y)$	2/3	1/2
$p(a_2 = \text{No}\|y)$	1/3	1/2

$$x_{\text{new}} = (\text{Yes}, \text{No})$$

$$p(y = \text{Yes}|x_{\text{new}}) =$$

$$\frac{|y=\text{Yes}|}{|X|} \prod_{i=1}^{n} p\Big(x_{a_i}|y = \text{Yes}\Big)$$

$$\prod_{i=1}^{n} p\Big(x_{a_i}|y = \text{Yes}\Big) = p(a_1 = \text{Yes}|y = \text{Yes}) * p(a_2 = \text{No}|y = \text{Yes})$$

$$= \frac{2}{3} * \frac{1}{3} = \frac{2}{9}$$

$$p(y = \text{Yes}|x_{\text{new}}) = \frac{|y=\text{Yes}|}{|X|} \prod_{i=1}^{n} p\Big(x_{a_i}|y\Big) = \frac{3}{5} * \frac{2}{9} = \frac{2}{15}$$

	y = Yes	y = No
$p(a_1 = \text{Yes}\|y)$	2/3	1
$p(a_1 = \text{No}\|y)$	1/3	0
$p(a_2 = \text{Yes}\|y)$	2/3	1/2
$p(a_2 = \text{No}\|y)$	1/3	1/2

$$x_{\text{new}} = (\text{Yes}, \text{No})$$

$$p(y = \text{No}|x_{\text{new}}) = \frac{|y=\text{No}|}{|X|} \prod_{i=1}^{n} p\left(x_{a_i}|y = \text{No}\right)$$

$$\prod_{i=1}^{n} p\left(x_{a_i}|y = \text{No}\right) = p(a_1 = \text{Yes}|y = \text{No}) * p(a_2 = \text{No}|y = \text{No})$$

$$= 1 * \frac{1}{2} = \frac{1}{2}$$

$$p(y = \text{No}|x_{\text{new}}) = \frac{|y=\text{No}|}{|X|} \prod_{i=1}^{n} p\left(x_{a_i}|y\right) = \frac{2}{5} * \frac{1}{2} = \frac{1}{5}$$

Naive Bayes Example

$$p(y = \text{Yes} | x_{\text{new}}) = \frac{2}{15}$$

$$p(y = \text{No} | x_{\text{new}}) = \frac{1}{5}$$

We choose the max, so we predict y = “No” for x_{new}.