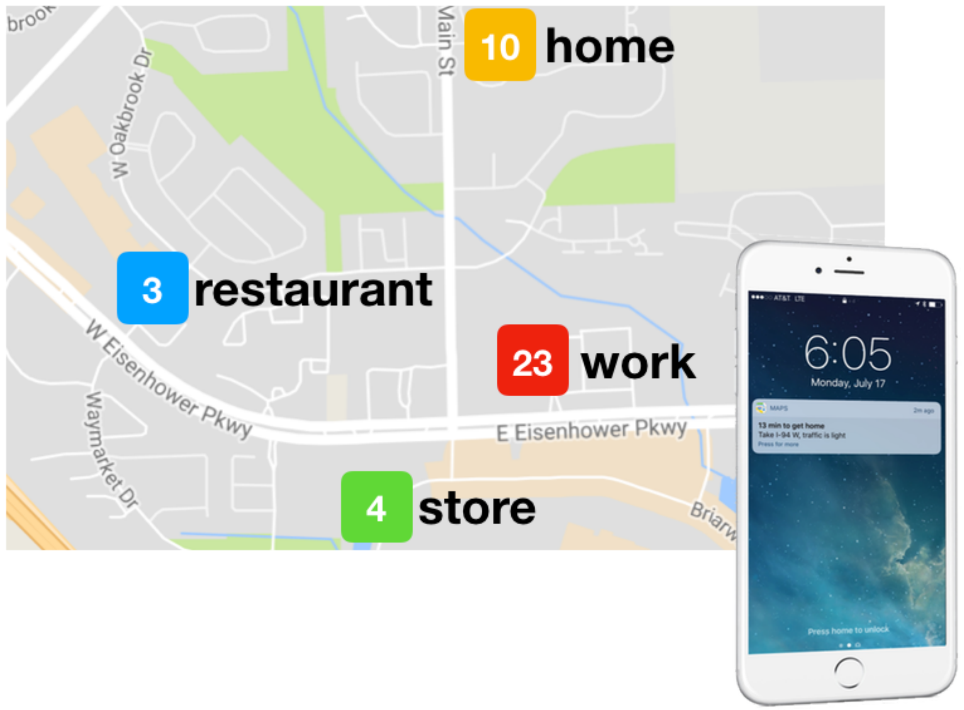
Understanding Bayesian methods

SUPERVISED LEARNING IN R: CLASSIFICATION

Brett Lantz
Instructor

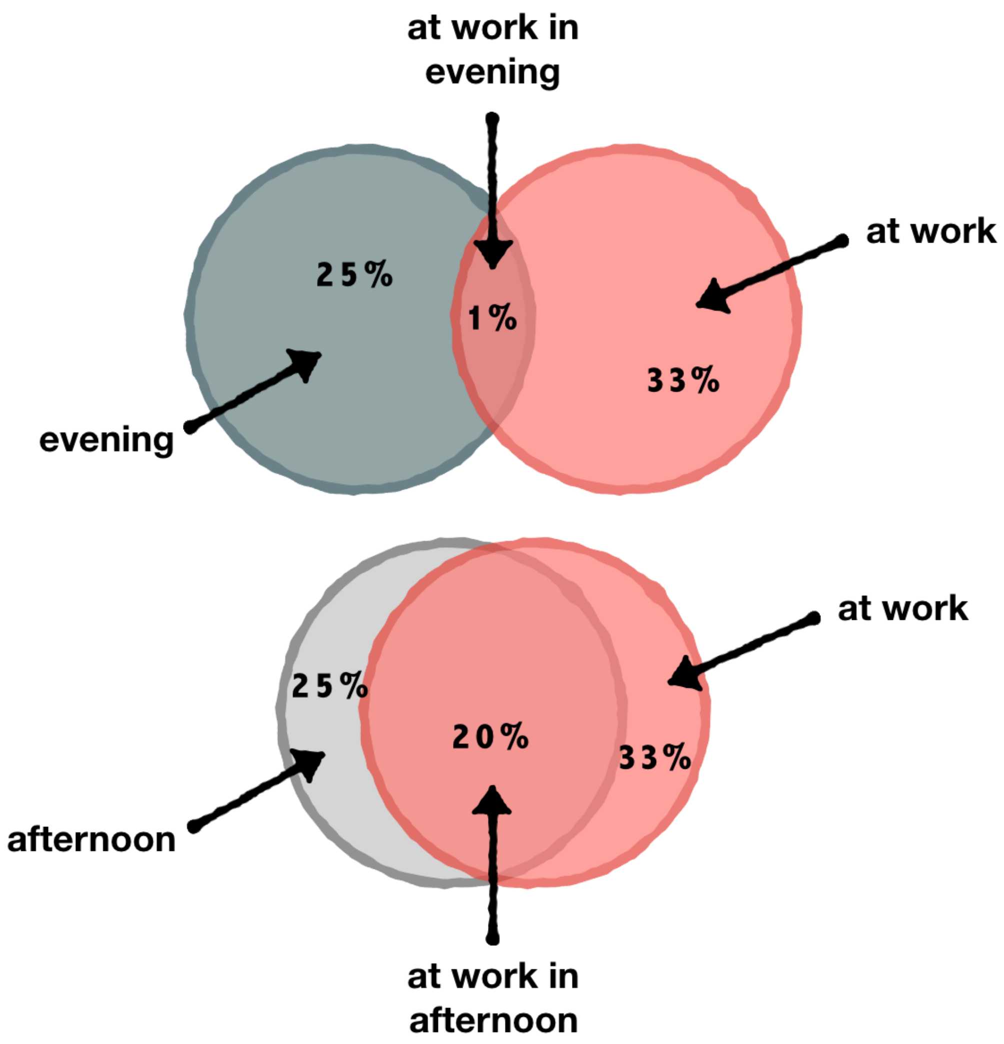
Estimating probability



The **probability** of A is denoted P(A)

- P(work) = 23 / 40 = 57.5%
- P(store) = 4 / 40 = 10.0%

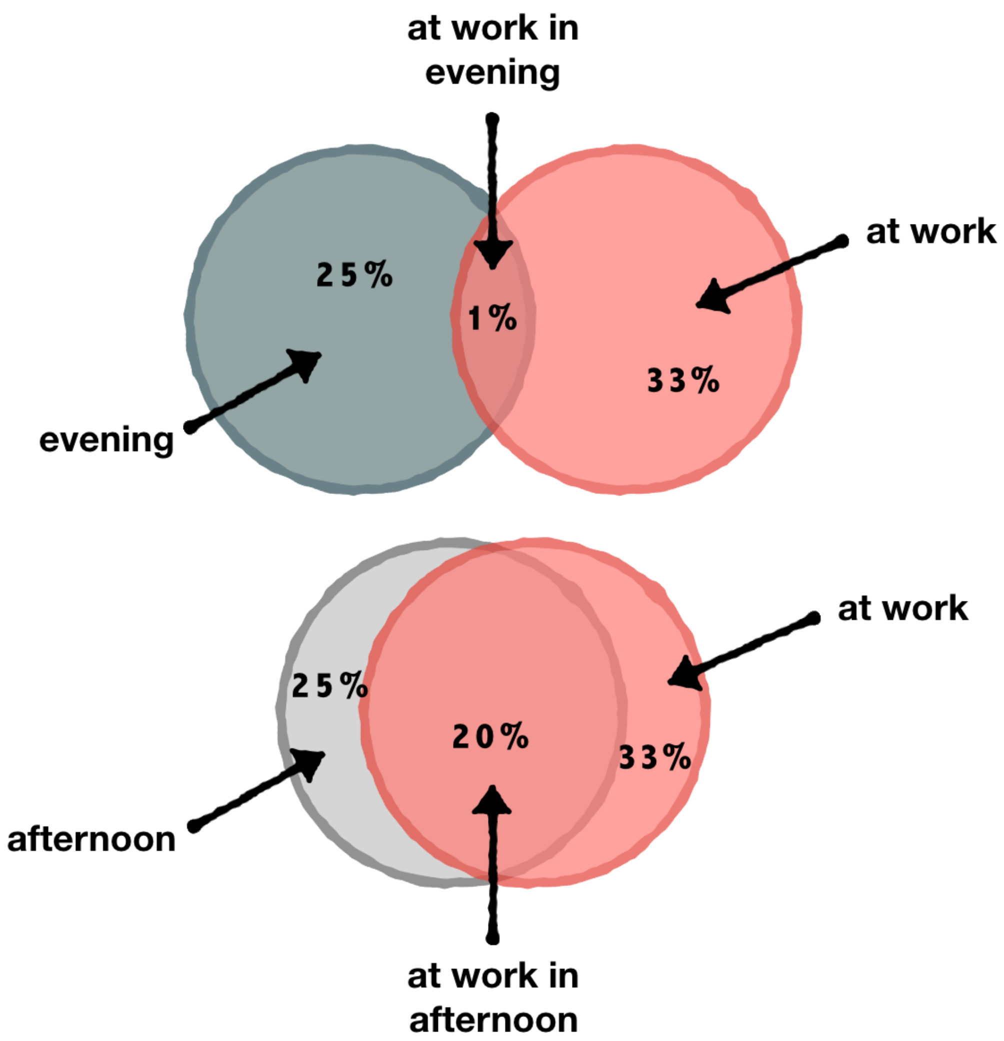
Joint probability and independent events



The **joint probability** of events A and B is denoted P(A and B)

- P(work and evening) = 1%
- P(work and afternoon) = 20%

Conditional probability and dependent events



The **conditional probability** of events A and B is denoted P(A | B)

- P(A | B) = P(A and B) / P(B)
- P(work | evening) = 1 / 25 = 4%
- P(work | afternoon) = 20 / 25 = 80%

Making predictions with Naive Bayes

```
# building a Naive Bayes model
library(naivebayes)
m <- naive_bayes(location ~ time_of_day, data = location_history)
```

```
# making predictions with Naive Bayes
future_location <- predict(m, future_conditions)
```

Let's practice!

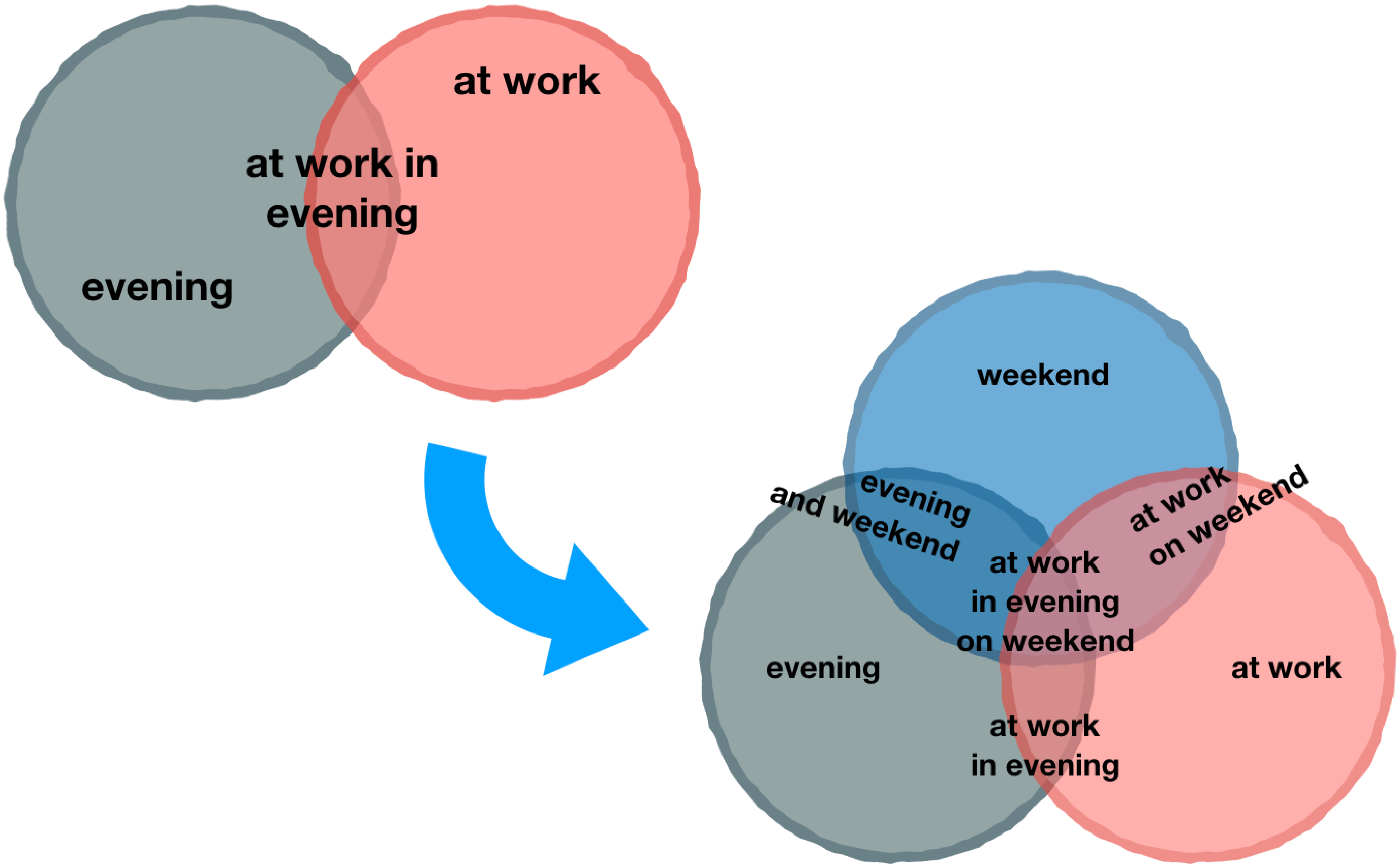
SUPERVISED LEARNING IN R: CLASSIFICATION

Understanding NB's "naivety"

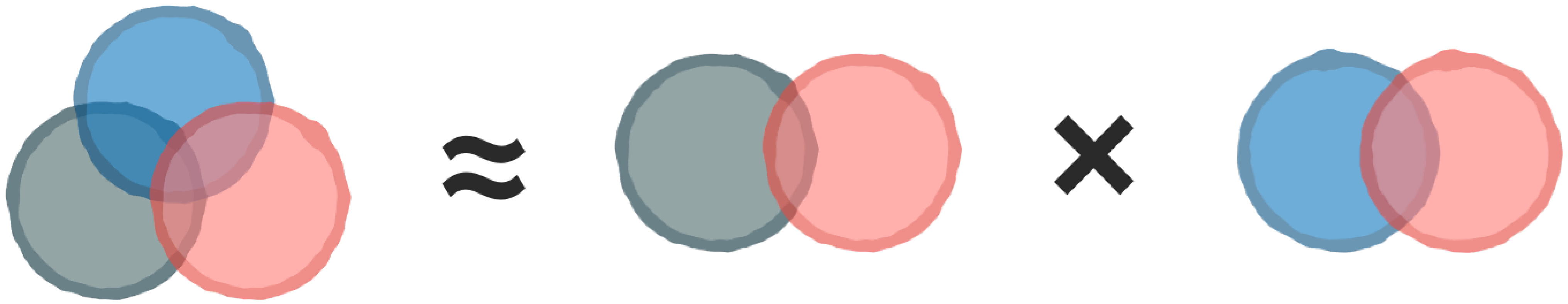
SUPERVISED LEARNING IN R: CLASSIFICATION

Brett Lantz
Instructor

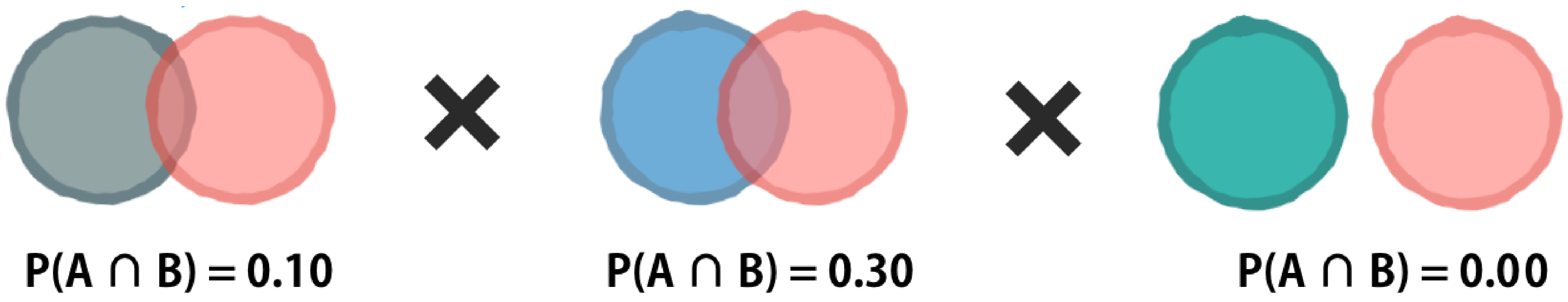
The challenge of multiple predictors



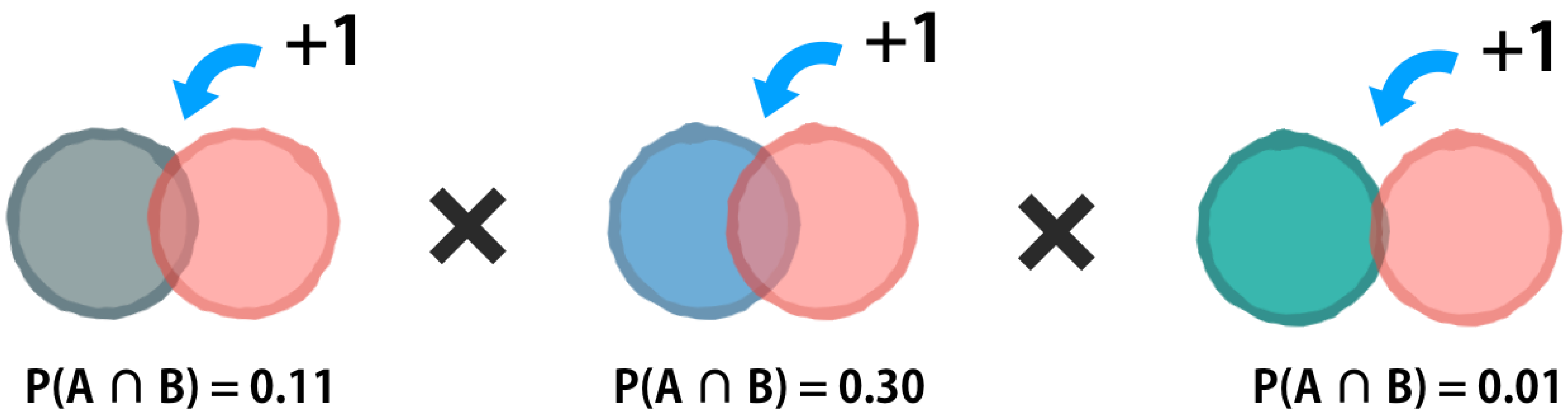
A "naive" simplification



An "infrequent" problem



The Laplace correction



Let's practice!

SUPERVISED LEARNING IN R: CLASSIFICATION

Applying Naive Bayes to other problems

SUPERVISED LEARNING IN R: CLASSIFICATION

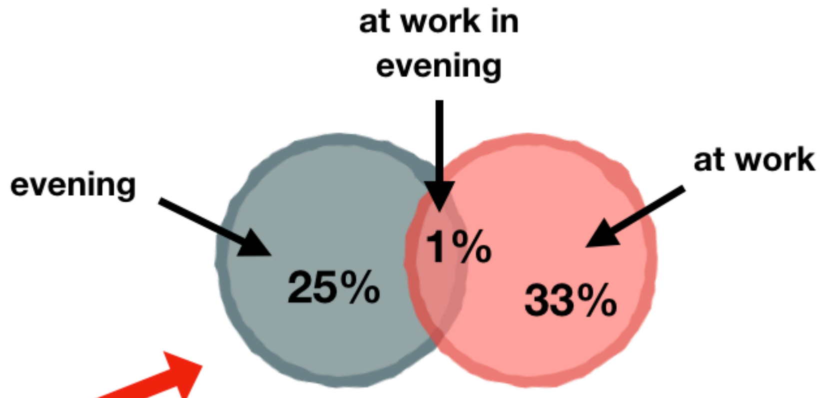
Brett Lantz
Instructor

How Naive Bayes uses data

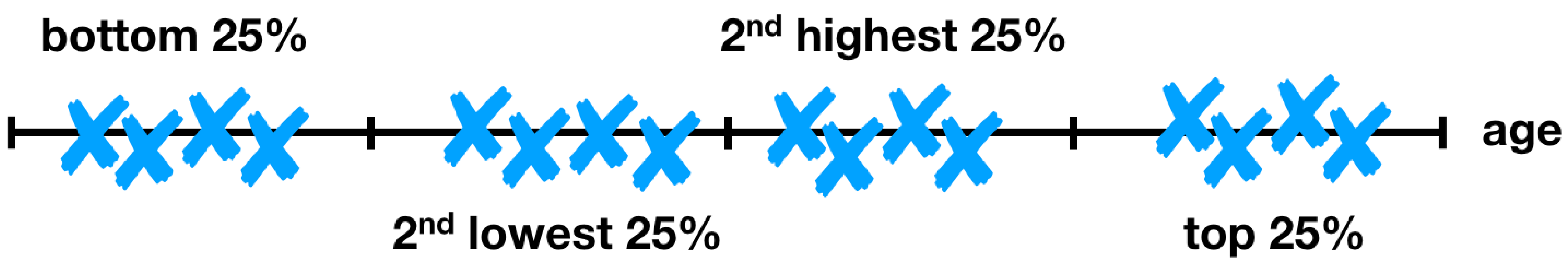
locations

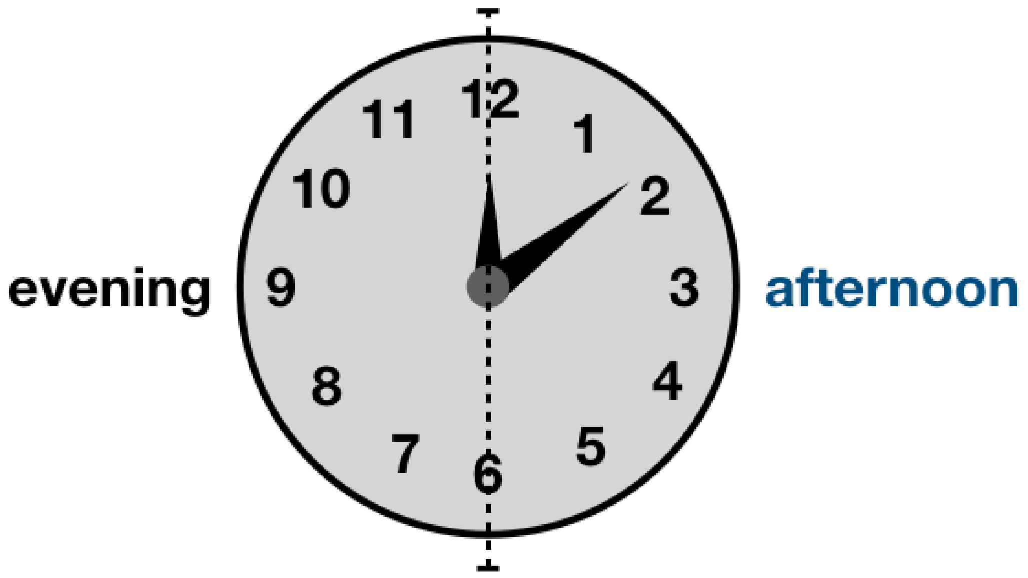
	month	day	weekday	daytype	hour	hourtype	location
1	1	4	wednesday	weekday	0	night	home
2	1	4	wednesday	weekday	1	night	home
3	1	4	wednesday	weekday	2	night	home
4	1	4	wednesday	weekday	3	night	home
5	1	4	wednesday	weekday	4	night	home
6	1	4	wednesday	weekday	5	night	home
7	1	4	wednesday	weekday	6	morning	home
8	1	4	wednesday	weekday	7	morning	home
9	1	4	wednesday	weekday	8	morning	home
10	1	4	wednesday	weekday	9	morning	office
11	1	4	wednesday	weekday	10	morning	office
12	1	4	wednesday	weekday	11	morning	office
13	1	4	wednesday	weekday	12	afternoon	office

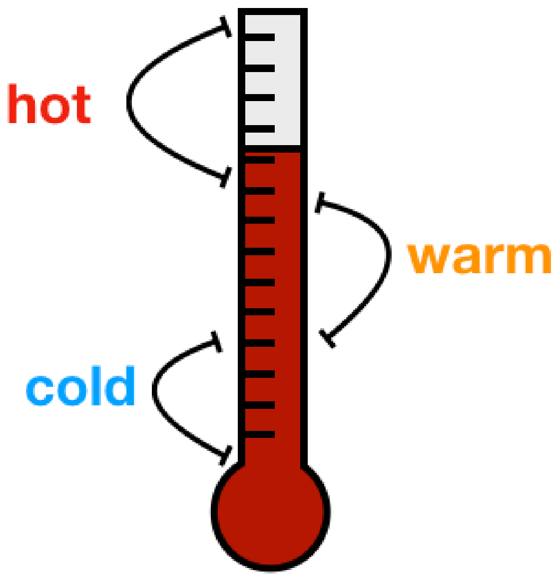
	not evening	evening
at work	705	22
not at work	933	524



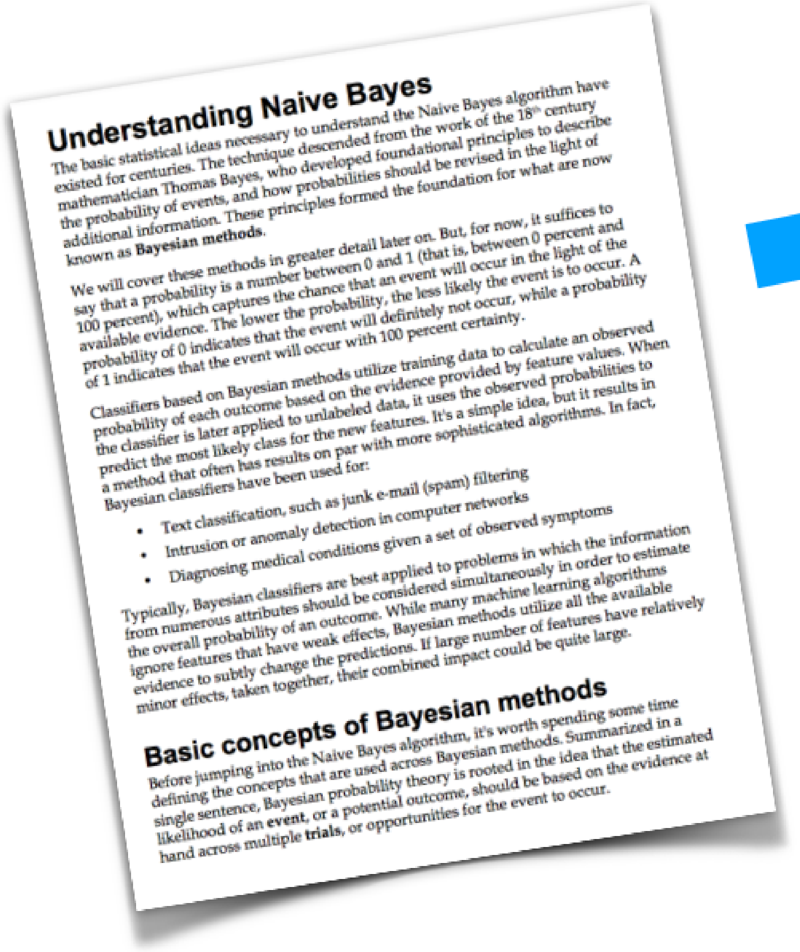
Binning numeric data for Naive Bayes







Preparing text data for Naive Bayes



Understanding Naive Bayes

The basic statistical ideas necessary to understand the Naive Bayes algorithm have existed for centuries. The technique descended from the work of the 18th century mathematician Thomas Bayes, who developed foundational principles to describe the probability of events, and how probabilities should be revised in the light of additional information. These principles formed the foundation for what are now known as **Bayesian methods**.

We will cover these methods in greater detail later on. But, for now, it suffices to say that a probability is a number between 0 and 1 (that is, between 0 percent and 100 percent), which captures the chance that an event will occur in the light of the available evidence. The lower the probability, the less likely the event is to occur. A probability of 0 indicates that the event will definitely not occur, while a probability of 1 indicates that the event will occur with 100 percent certainty.

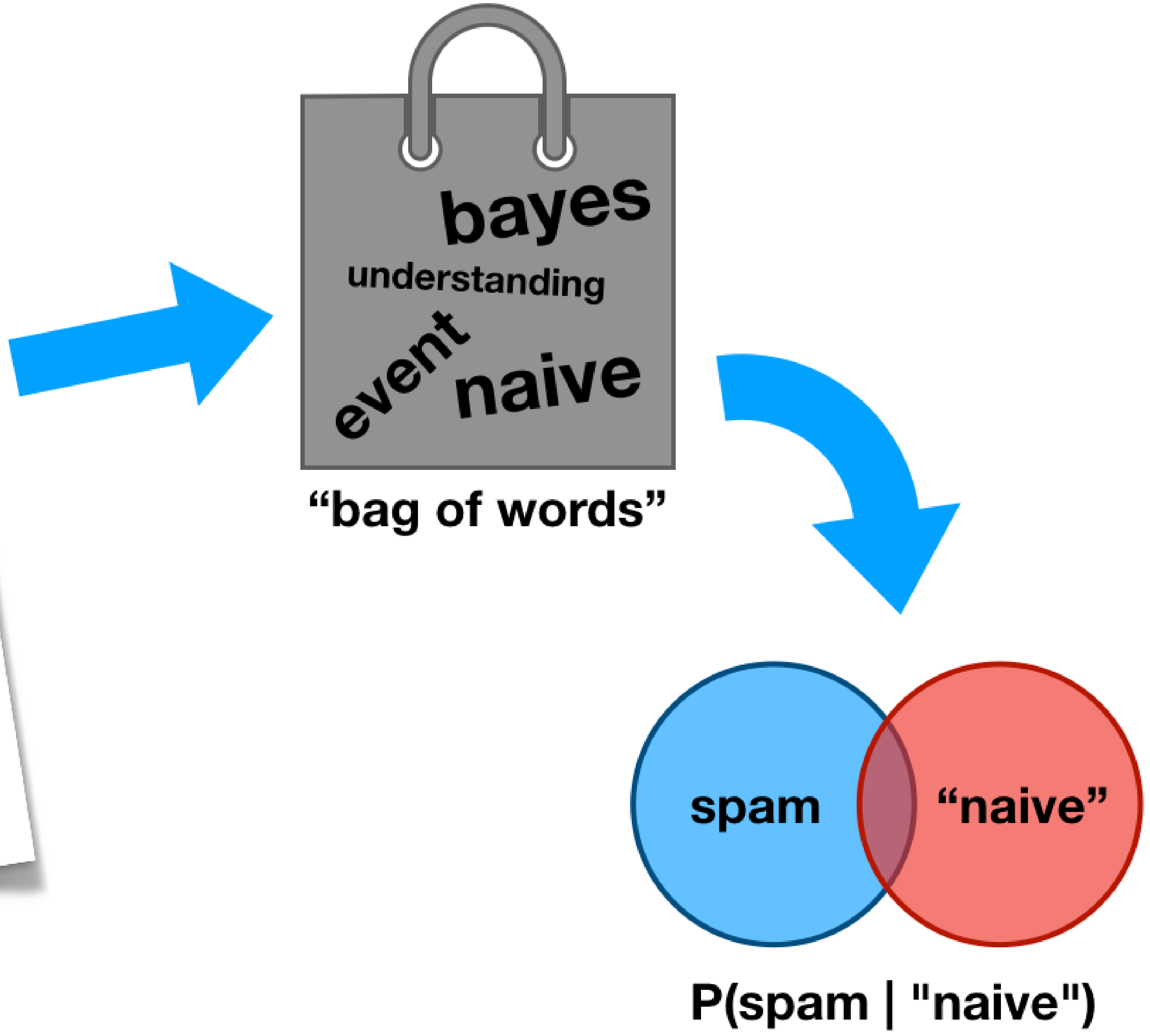
Classifiers based on Bayesian methods utilize training data to calculate an observed probability of each outcome based on the evidence provided by feature values. When the classifier is later applied to unlabeled data, it uses the observed probabilities to predict the most likely class for the new features. It's a simple idea, but it results in a method that often has results on par with more sophisticated algorithms. In fact, Bayesian classifiers have been used for:

- Text classification, such as junk e-mail (spam) filtering
- Intrusion or anomaly detection in computer networks
- Diagnosing medical conditions given a set of observed symptoms

Typically, Bayesian classifiers are best applied to problems in which the information from numerous attributes should be considered simultaneously in order to estimate the overall probability of an outcome. While many machine learning algorithms ignore features that have weak effects, Bayesian methods utilize all the available evidence to subtly change the predictions. If large number of features have relatively minor effects, taken together, their combined impact could be quite large.

Basic concepts of Bayesian methods

Before jumping into the Naive Bayes algorithm, it's worth spending some time defining the concepts that are used across Bayesian methods. Summarized in a single sentence, Bayesian probability theory is rooted in the idea that the estimated likelihood of an **event**, or a potential outcome, should be based on the evidence at hand across multiple **trials**, or opportunities for the event to occur.



Let's practice!

SUPERVISED LEARNING IN R: CLASSIFICATION