**Examining "raw" probabilities**

**Instructions**

* Print the locmodel object to the console to view the computed *a priori* and conditional probabilities.
* Use the predict() function similarly to the previous exercise, but with type = "prob" to see the predicted probabilities for Thursday at 9am.
* Compare these to the predicted probabilities for Saturday at 9am.

Code

# Examine the location prediction model

locmodel

# Obtain the predicted probabilities for Thursday at 9am

predict(locmodel, thursday9am, type = "prob")

# Obtain the predicted probabilities for Saturday at 9am

predict(locmodel, saturday9am, type = "prob")

**Understanding independence**

Understanding the idea of event independence will become important as you learn more about how "naive" Bayes got its name. Which of the following is true about independent events?

* The events cannot occur at the same time.
* A Venn diagram will always show no intersection.
* Knowing the outcome of one event does not help predict the other. (**TRUE**)
* At least one of the events is completely random.

**Why 3rd one is TRUE**

**The events cannot occur at the same time**: This is incorrect. Independent events can occur simultaneously. For instance, getting heads on a coin flip and rolling a 4 on a die are independent, yet both can occur in the same trial.

**The Venn diagram** shows an intersection if the events can occur together, but this doesn't mean they're dependent.

**At least one of the events is completely random**: This is incorrect. The concept of independence doesn’t require one event to be random in the sense of being unpredictable. Both events can be deterministic but still independent.

**Why 3rd is TRUE**

**No Impact on Probability**: If AAA and BBB are independent, knowing whether AAA has occurred or not does not provide any additional information about whether BBB will occur. This lack of influence is what defines their independence.

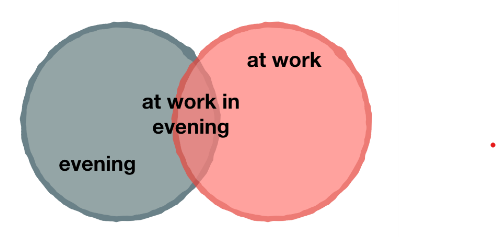
**why it’s called “naive”**

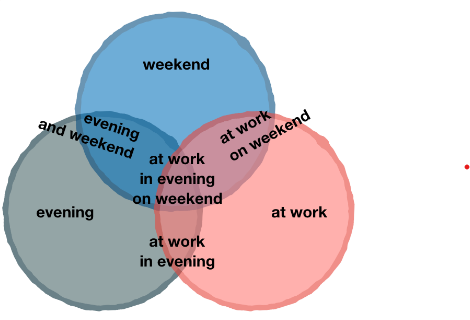
1. **Starting Simple with One Predictor**

* Imagine you’re predicting where someone will go based on one thing, like the weather. If it's sunny, you predict they’ll go to the park. This is pretty straightforward and easy to handle.

1. **Adding More Predictors**

* Now, say you want to use more information, like the weather, day of the week, and if they have an appointment. When you try to consider all these factors together, it gets complicated, like trying to draw overlapping circles in a Venn diagram. With many factors, the diagram can get very messy, and figuring out how everything overlaps becomes tough for a computer.





1. **The “Naive” Assumption**

 To simplify this, Naive Bayes makes a “naive” assumption: it treats each factor as if it were completely independent of the others. This means it assumes the weather, day of the week, and appointment don’t influence each other.

 Because of this, instead of dealing with a complicated Venn diagram, the algorithm just multiplies the probabilities of each factor happening. This makes calculations easier

and faster.

**The Simplified Calculation Approach Imagine You Have Three Predictors:**

1. **Weather (Sunny or Not Sunny)**
2. **Day of the Week (Weekday or Weekend)**
3. **Appointment (Yes or No)**

You want to predict where someone might go based on these three factors. If you tried to understand how all these factors interact with each other, the diagram and calculations would get really complex.

**Complicated Approach**

If you were to use a detailed approach, you'd need to:

* Look at all possible combinations of these factors.
* Check how often each combination happens together.
* Calculate the probability based on these combinations.

For instance, you’d need to figure out the probability of it being sunny **and** a weekday **and** having an appointment all at once.

**Naive Bayes Simplification**

**Naive Bayes** simplifies this by making a key assumption:

* **Assumption**: Each factor is independent of the others. This means it assumes that knowing it’s sunny doesn’t change the probability of it being a weekday or having an appointment.

**How It Works:**

1. **Calculate Each Factor’s Probability**:
   * Probability of it being sunny.
   * Probability of it being a weekday.
   * Probability of having an appointment.
2. **Multiply These Probabilities Together**:
   * Instead of dealing with complicated overlaps, you just multiply the individual probabilities:



**Why This Works:**

* **Simplified Calculation**: By assuming independence, the math becomes straightforward. You don’t have to worry about how factors influence each other directly. Just multiply their chances.
* **Faster and Easier**: It’s easier for the computer to handle these simple multiplications rather than working out all the overlaps in a complex Venn diagram.

**Example**

Let’s say:

* Probability of sunny weather = 0.6
* Probability of it being a weekday = 0.5
* Probability of having an appointment = 0.4

Using Naive Bayes, you would calculate the combined probability of all three factors being true by multiplying:



This simple multiplication is much easier and quicker than calculating how these factors interact in all possible combinations.

1. **The Problem with Zeroes**

 Here’s a hiccup: If a particular combination of factors (like working on a weekend) has never happened before, the model thinks it’s impossible (i.e., it assigns a probability of zero).

 When probabilities are multiplied and one of them is zero, the whole result becomes zero. So if the model has never seen someone working on a weekend, it’ll always predict zero chance of this happening, even if it might in the future.

1. **Fixing the Problem with Laplace Correction**

* To solve this problem, we use something called the Laplace correction. This means adding a tiny value (like 1) to each factor so that no probability is exactly zero.
* With this correction, even if something hasn’t been seen before, the model will still give it a small chance of happening. This way, it doesn’t completely rule out possibilities just because they haven’t been observed before.

**Summary**

* **Naive Bayes** simplifies complex predictions by assuming all factors are independent.
* **The problem** is that if some combinations of factors haven’t been seen, the model might predict a zero chance for them.
* **Laplace Correction** fixes this by adding a small value to ensure that every possible outcome has at least a tiny chance of happening.

**A more sophisticated location model**

**Instructions**

* **Use the R formula interface to build a model where location depends on both daytype and hourtype. Recall that the function naive\_bayes() takes 2 arguments: formula and data.**
* **Predict Brett's location on a weekday afternoon using the data frame weekday\_afternoon and the predict() function.**
* **Do the same for a weekday\_evening.**

**CODE**

locmodel <- naive\_bayes(location ~ daytype + hourtype, data = locations)

weekday\_evening <- read\_csv("weekday\_evening.csv")

predict(locmodel,weekday\_evening)

weekday\_afternoon <- read\_csv("weekday\_afternoon.csv")

predict(locmodel,weekday\_afternoon)

**Preparing for unforeseen circumstances**

**Instructions**

* Use the locmodel to output predicted probabilities for a weekend afternoon by using the predict() function. Remember to set the type argument.
* Create a new naive Bayes model with the Laplace smoothing parameter set to 1. You can do this by setting the laplace argument in your call to naive\_bayes(). Save this as locmodel2.
* See how the new predicted probabilities compare by using the predict() function on your new model.

**CODE**

# Observe the predicted probabilities for a weekend afternoon

predict(locmodel, weekend\_afternoon, type = "prob")

# Build a new model using the Laplace correction

locmodel2 <- naive\_bayes(location ~ daytype + hourtype, data = locations, laplace = 1)

# Observe the new predicted probabilities for a weekend afternoon

predict(locmodel2, weekend\_afternoon, type = "prob")

**QUESTION:**

**Understanding the Laplace correction**

By default, the naive\_bayes() function in the naivebayes package does not use the Laplace correction. What is the risk of leaving this parameter unset?

* Some potential outcomes may be predicted to be impossible. (**TRUE**)
* The algorithm may have a divide by zero error.
* Naive Bayes will ignore features with zero values.
* The model may not estimate probabilities for some cases.