Application: Naive Bayes Classifiers
Table of Contents

1 Labeling

2 Statistical modeling

3 A Naive Bayes Classifier
Predicting the world

Things we may want to do with our computer-science powers:
Predicting the world

Things we may want to do with our computer-science powers:

- Diagnose if a person has the flu/a cold/nothing based on their symptoms.
Predicting the world

Things we may want to do with our computer-science powers:

- Diagnose if a person has the flu/a cold/nothing based on their symptoms.
- Filter emails based on whether or not they are likely to be spam.
Classification problems

A lot of problems can be expressed in terms of input data and output labels...
Classification problems

A lot of problems can be expressed in terms of **input** data and **output** labels...

- Data: A list of symptoms for each patient.
  Labels: Flu/Cold/Healthy.
Classification problems

A lot of problems can be expressed in terms of input data and output labels...

- Data: A list of symptoms for each patient.  
  Labels: Flu/Cold/Healthy.
- Data: The words constituting an email.  
  Labels: Spam/Ham.
A lot of problems can be expressed in terms of **input** data and **output** labels...

- Data: A list of symptoms for each patient.
  Labels: Flu/Cold/Healthy.

- Data: The words constituting an email.
  Labels: Spam/Ham.

**Definition**

A **classification problem** is one where we classify data points by giving each one a label.
Classification problems (continued)

What are we actually doing, though?
Classification problems (continued)

What are we actually doing, though?

- Which is most likely, flu/cold/healthy, given the symptoms of a patient?
Classification problems (continued)

What are we actually doing, though?

- Which is most likely, flu/cold/healthy, given the symptoms of a patient?
- Which is most likely, spam/ham, given the words in an email?
Table of Contents

1. Labeling

2. Statistical modeling

3. A Naive Bayes Classifier
A philosophical question...

How does the world work?

Let “→” suggest causation:
A philosophical question...

How does the world work?

Let “→” suggest causation:

- Sickness (the disease/virus) → Symptoms

OR

- Symptoms → Sickness (the name)
A philosophical question...

How does the world work?

Let “→” suggest causation:

- Sickness (the disease/virus) → Symptoms
- OR
- Symptoms → Sickness (the name)
- Spam (the intent) → Email
- OR
- Email → Spam (the meaning)
It’s not clear what direction causality goes!
A philosophical question... (continued)

It’s not clear what direction causality goes!

In practice, use whichever works better/faster/whatever you care about.
You’re sick!

Let’s assume that the fact that someone has the flu causes them to manifest certain symptoms, i.e. $\text{Sickness} \rightarrow \text{Symptoms}$. 
You’re sick!

Let’s assume that the fact that someone has the flu causes them to manifest certain symptoms, i.e. Sickness $\rightarrow$ Symptoms.

We’d expect
You’re sick!

Let’s assume that the fact that someone has the flu causes them to manifest certain symptoms, i.e. Sickness \rightarrow\text{ Symptoms}.

We’d expect

- Healthy people do not have fevers
You’re sick!

Let’s assume that the fact that someone has the flu causes them to manifest certain symptoms, i.e. Sickness $\rightarrow$ Symptoms.

We’d expect

- Healthy people do not have fevers
- People with the flu have upset stomachs and sore throats
You’re sick!

Let’s assume that the fact that someone has the flu causes them to manifest certain symptoms, i.e. Sickness $\rightarrow$ Symptoms.

We’d expect

- Healthy people do not have fevers
- People with the flu have upset stomachs and sore throats
- People with a cold have a runny nose and a cough.
You’re sick! (continued)

We could write code to try and model this:

```python
def diagnose(symptoms):
    diagnoses = {'flu', 'cold', 'healthy'}
    if 'fever' in symptoms:
        # Patient has a fever, can’t be healthy
        diagnoses.remove('healthy')
    if 'upset stomach' and 'sore throat' not in symptoms:
        # Patient doesn’t have an upset stomach or a sore
        # throat, can’t have the flu
        diagnoses.remove('flu')
    if 'runny nose' and 'cough' not in symptoms:
        # Patient doesn’t have a runny nose or a cough,
        # can’t have a cold
        diagnoses.remove('cold')
    return diagnoses
```

This... sucks. Sometimes we’re going to have multiple diagnoses we can’t
differentiate, sometimes we’ll eliminate all the diagnoses.
You’re sick! (continued)

We could write code to try and model this:

```python
def diagnose(symptoms):
    diagnoses = {'flu', 'cold', 'healthy'}
    if 'fever' in symptoms:
        # Patient has a fever, can’t be healthy
        diagnoses.remove('healthy')
    if 'upset stomach' and 'sore throat' not in symptoms:
        # Patient doesn’t have an upset stomach or a sore
        # throat, can’t have the flu
        diagnoses.remove('flu')
    if 'runny nose' and 'cough' not in symptoms:
        # Patient doesn’t have a runny nose or a cough,
        # can’t have a cold
        diagnoses.remove('cold')
    return diagnoses
```

This... sucks. Sometimes we’re going to have multiple diagnoses we can’t differentiate, sometimes we’ll eliminate all the diagnoses.
You’re sick! (continued)

We’re taking 312 – we can deal with this using *probability*!

Example

```python
def diagnose(symptoms):
    diagnoses = {"flu" : 1/3, "cold" : 1/3, "healthy" : 1/3}
    if 'fever' in symptoms:
        # Patient has a fever, very unlikely to be healthy
        diagnoses["healthy"] /= 5.0
        diagnoses.normalize()
    if 'upset stomach' and 'sore throat' not in symptoms:
        # Patient doesn’t have an upset stomach or a sore
        # throat, unlikely to have the flu
        diagnoses["flu"] /= 2.0
        diagnoses.normalize()
    if 'runny nose' and 'cough' not in symptoms:
        # Patient doesn’t have a runny nose or a cough
        # unlikely to have a cold
        diagnoses["cold"] /= 2.0
        diagnoses.normalize()
    return diagnoses
```
You’re sick! (continued)

We’re taking 312 – we can deal with this using probability!

**Example**

```python
def diagnose(symptoms):
    diagnoses = {'flu': 1/3, 'cold': 1/3, 'healthy': 1/3}
    if 'fever' in symptoms:
        # Patient has a fever, very unlikely to be healthy
        diagnoses['healthy'] /= 5.0
        diagnoses.normalize()
    if 'upset stomach' and 'sore throat' not in symptoms:
        # Patient doesn’t have an upset stomach or a sore
        # throat, unlikely to have the flu
        diagnoses['flu'] /= 2.0
        diagnoses.normalize()
    if 'runny nose' and 'cough' not in symptoms:
        # Patient doesn’t have a runny nose or a cough
        # unlikely to have a cold
        diagnoses['cold'] /= 2.0
        diagnoses.normalize()
    return diagnoses
```

Instead of making it impossible for healthy people to have fevers, we just make it less likely the patient is healthy when they have a fever!
You’re sick! (continued)

```python
def diagnose(symptoms):
    diagnoses = {'flu': 1/3, 'cold': 1/3, 'healthy': 1/3}
    if 'fever' in symptoms:
        # Patient has a fever, very unlikely to be healthy
        diagnoses['healthy'] /= 5
    diagnoses.normalize()
    if 'upset stomach' and 'sore throat' not in symptoms:
        # Patient doesn’t have an upset stomach or a sore throat, unlikely to have the flu
        diagnoses['flu'] /= 2
    diagnoses.normalize()
    if 'runny nose' and 'cough' not in symptoms:
        # Patient doesn’t have a runny nose or a cough
        # unlikely to have a cold
        diagnoses['cold'] /= 2
    diagnoses.normalize()
    return diagnoses
```

Let’s enhance our model to check all possible combinations of symptoms. How many if-statements do we have to write?
Welp... We’re screwed.
Intractable

Welp... We’re screwed.

5 symptoms means $2^5 = 32$ if-statements to check all combinations. 10 symptoms would be 1024 if-statements.
Welp... We’re screwed.

5 symptoms means $2^5 = 32$ if-statements to check all combinations. 10 symptoms would be 1024 if-statements.

There isn’t even an upper-bound for the number of words we could have in an email...
Two major problems in statistical modeling

We assumed that \( \text{Sickness} \rightarrow \text{Symptoms} \), so our model predicts the probability certain symptoms manifest \textit{given} we know what the patient has.

<table>
<thead>
<tr>
<th>What we have</th>
<th>What we want</th>
</tr>
</thead>
</table>

We need to somehow address all combinations of symptoms and how they affect each diagnosis.
Two major problems in statistical modeling

1. We assumed that $\text{Sickness} \rightarrow \text{Symptoms}$, so our model predicts the probability certain symptoms manifest *given* we know what the patient has.

<table>
<thead>
<tr>
<th>What we have</th>
<th>What we want</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Pr(\text{symptoms}</td>
<td>\text{diagnosis})$</td>
</tr>
</tbody>
</table>

We want to choose the highest probability diagnosis, so what we want is to compute $\Pr(\text{diagnosis}|\text{symptoms})$ for each diagnosis.

We have the opposite of what we need!

We need to somehow address all combinations of symptoms and how they affect each diagnosis.
Two major problems in statistical modeling

1. We assumed that \textbf{Sickness} \rightarrow \textbf{Symptoms}, so our model predicts the probability certain symptoms manifest \textit{given} we know what the patient has.

<table>
<thead>
<tr>
<th>What we have</th>
<th>What we want</th>
</tr>
</thead>
<tbody>
<tr>
<td>\Pr(\text{symptoms}</td>
<td>\text{diagnosis})</td>
</tr>
</tbody>
</table>

We want to choose the highest probability diagnosis, so what we want is to compute \Pr(\text{diagnosis}|\text{symptoms}) for each diagnosis.
Two major problems in statistical modeling

We assumed that $Sickness \rightarrow Symptoms$, so our model predicts the probability certain symptoms manifest *given* we know what the patient has.

<table>
<thead>
<tr>
<th>What we have</th>
<th>What we want</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{Pr}(\text{symptoms} \mid \text{diagnosis})$</td>
<td>$\text{Pr}(\text{diagnosis} \mid \text{symptoms})$</td>
</tr>
</tbody>
</table>

We want to choose the highest probability diagnosis, so what we want is to compute $\text{Pr}(\text{diagnosis} \mid \text{symptoms})$ for each diagnosis.
Two major problems in statistical modeling

1. We assumed that **Sickness \( \rightarrow \) Symptoms**, so our model predicts the probability certain symptoms manifest *given* we know what the patient has.

<table>
<thead>
<tr>
<th>What we have</th>
<th>What we want</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \Pr(\text{symptoms}</td>
<td>\text{diagnosis}) )</td>
</tr>
</tbody>
</table>

We want to choose the highest probability diagnosis, so what we want is to compute \( \Pr(\text{diagnosis}|\text{symptoms}) \) for each diagnosis.

We have the opposite of what we need!
Two major problems in statistical modeling

1. We assumed that \( Sickness \rightarrow Symptoms \), so our model predicts the probability certain symptoms manifest \textit{given} we know what the patient has.

<table>
<thead>
<tr>
<th>What we have</th>
<th>What we want</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Pr(symptoms</td>
<td>diagnosis) )</td>
</tr>
</tbody>
</table>

We want to choose the highest probability diagnosis, so what we want is to compute \( Pr(diagnosis|symptoms) \) for each diagnosis.

We have the opposite of what we need!

2. We need to somehow address all combinations of symptoms and how they affect each diagnosis.
Maximize!

\[
\max_{\text{diagnosis}} Pr(\text{diagnosis}|\text{symptoms})
\]
Pr(S|D) vs. Pr(D|S)

Maximize!

\[ \max_{\text{diagnosis}} \Pr(\text{diagnosis}|\text{symptoms}) \]

Apply Bayes Rule!

\[ \max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \frac{\Pr(\text{symp}|\text{diag}) \cdot \Pr(\text{diag})}{\Pr(\text{symp})}. \]
Pr(S|D) vs. Pr(D|S)

Maximize!

\[
\max_{\text{diagnosis}} \Pr(\text{diagnosis}|\text{symptoms})
\]

Apply Bayes Rule!

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \frac{\Pr(\text{symp}|\text{diag}) \cdot \Pr(\text{diag})}{\Pr(\text{symp})}.
\]

Notice that if we want to find the most likely diagnosis for a set of symptoms, we vary the diagnosis and compute probability. Since we don’t vary the symptoms, \( \Pr(\text{symptoms}) = c \)
Thus, we see that

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \frac{1}{c} \cdot \Pr(\text{symp}|\text{diag}) \cdot \Pr(\text{diag})
\]

\[
= \frac{1}{c} \max_{\text{diag}} \Pr(\text{symp}|\text{diag}) \cdot \Pr(\text{diag}).
\]
What we’ve derived:

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \frac{1}{c} \cdot \max_{\text{diag}} \Pr(\text{symp}|\text{diag}) \cdot \Pr(\text{diag}).
\]

In English: To maximize the probability of the diagnosis given the symptoms, choose a diagnosis which is likely to cause those symptoms and isn’t too unlikely itself.
Combinations of symptoms

Clearly some symptoms are not actually related to each other given what we are trying to diagnose, e.g. runny nose and swelling of the feet and toes.
Combinations of symptoms

Clearly some symptoms are not actually related to each other given what we are trying to diagnose, e.g. runny nose and swelling of the feet and toes.

It makes our problem **tractable** [read: a lot easier] if we assume that unrelated symptoms are independent given a diagnosis!
Combinations of symptoms

Clearly some symptoms are not actually related to each other given what we are trying to diagnose, e.g. runny nose and swelling of the feet and toes.

It makes our problem tractable [read: a lot easier] if we assume that unrelated symptoms are independent given a diagnosis!

Let’s assume that all the symptoms are independent of one another given the diagnosis.
Combinations of symptoms

Clearly some symptoms are not actually related to each other given what we are trying to diagnose, e.g. runny nose and swelling of the feet and toes.

It makes our problem tractable [read: a lot easier] if we assume that unrelated symptoms are independent given a diagnosis!

Let’s assume that all the symptoms are independent of one another given the diagnosis.

That’s pretty naive, but now we don’t need domain knowledge about which symptoms commonly occur together.
Combinations of symptoms continued

Claim

Conditional independence of all the symptoms allows us to say

\[
Pr(s_1 \cap s_2 \cap \ldots \cap s_n | \text{diagnosis}) = \prod_{i=1}^{n} Pr(s_i | \text{diagnosis}).
\]
Combinations of symptoms continued

Claim

Conditional independence of all the symptoms allows us to say

\[ \Pr(s_1 \cap s_2 \cap \ldots \cap s_n | \text{diagnosis}) = \prod_{i=1}^{n} \Pr(s_i | \text{diagnosis}). \]

See the Naive Bayes notes for a full derivation using the product [chain] rule!
1 Labeling

2 Statistical modeling

3 A Naive Bayes Classifier
Our first classifier

Apply Bayes Rule!
Our first classifier

Apply Bayes Rule!

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \Pr(\text{symp}|\text{diag}) \cdot \Pr(\text{diag}).
\]
Our first classifier

Apply Bayes Rule!

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \Pr(\text{symp}|\text{diag}) \cdot \Pr(\text{diag}).
\]

Apply conditional independence!
Our first classifier

Apply Bayes Rule!

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \Pr(\text{symp}|\text{diag}) \cdot \Pr(\text{diag}).
\]

Apply conditional independence!

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \left[ \Pr(\text{diag}) \cdot \prod_{s \in \text{symp}} \Pr(s|\text{diag}) \right].
\]
Our first classifier

Apply Bayes Rule!

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \Pr(\text{symp}|\text{diag}) \cdot \Pr(\text{diag}).
\]

Apply conditional independence!

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \left[ \Pr(\text{diag}) \cdot \prod_{s \in \text{symp}} \Pr(s|\text{diag}) \right].
\]

That’s it!
Our second classifier

Applying the exact same steps
Our second classifier

Applying the exact same steps

\[
\max_{\text{spam}} \Pr(\text{spam}|\text{email})
\]
Our second classifier

Applying the exact same steps

$$\max_{\text{spam}} \Pr(\text{spam}|\text{email}) = \max_{\text{spam}} \Pr(\text{email}|\text{spam}) \cdot \Pr(\text{spam})$$
Our second classifier

Applying the exact same steps

\[
\max_{\text{spam}} \Pr(\text{spam}|\text{email}) = \max_{\text{spam}} \Pr(\text{email}|\text{spam}) \cdot \Pr(\text{spam}) \\
= \max_{\text{spam}} \left[ \Pr(\text{spam}) \cdot \prod_{\text{word} \in \text{email}} \Pr(\text{word}|\text{spam}) \right].
\]
Training a Naive Bayes Model

What we’ve derived so far:

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \left[ \Pr(\text{diag}) \cdot \prod_{s \in \text{symp}} \Pr(s|\text{diag}) \right].
\]

Suppose we have a database of patients with symptoms, and a doctor has professionally diagnosed all of them.

Our model needs to learn two things in order to make predictions [by computing \( \Pr(\text{diagnosis}|\text{symptoms}) \)]:
Training a Naive Bayes Model

What we’ve derived so far:

\[
\max_{\text{diag}} \Pr(\text{diag}|\text{symp}) = \max_{\text{diag}} \left[ \Pr(\text{diag}) \cdot \prod_{s \in \text{symp}} \Pr(s|\text{diag}) \right].
\]

Suppose we have a database of patients with symptoms, and a doctor has professionally diagnosed all of them.

Our model needs to learn two things in order to make predictions [by computing \( \Pr(\text{diagnosis}|\text{symptoms}) \)]:

- \( \Pr(\text{diagnosis}) \) for each diagnosis.
Training a Naive Bayes Model

What we’ve derived so far:

\[
\max_{\text{diag}} \Pr(\text{diag} | \text{symp}) = \max_{\text{diag}} \left[ \Pr(\text{diag}) \cdot \prod_{s \in \text{symp}} \Pr(s | \text{diag}) \right].
\]

Suppose we have a database of patients with symptoms, and a doctor has professionally diagnosed all of them.

Our model needs to learn two things in order to make predictions [by computing \( \Pr(\text{diagnosis} | \text{symptoms}) \)]:

- \( \Pr(\text{diagnosis}) \) for each diagnosis.
- \( \Pr(\text{symptom} | \text{diagnosis}) \) for each symptom, for each diagnosis.
Consider the training set:

<table>
<thead>
<tr>
<th>Name</th>
<th>Cough</th>
<th>Sore Throat</th>
<th>Fever</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>D</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>E</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Healthy</td>
</tr>
</tbody>
</table>

\[ |\text{Cough} \cap \text{Flu}| = \]
\[ |\text{Flu}| = \]

Pr(\text{Cough}|\text{Flu}) =

Pr(\text{Flu}) =
Training a Naive Bayes Model continued

Example

Consider the training set:

<table>
<thead>
<tr>
<th>Name</th>
<th>Cough</th>
<th>Sore Throat</th>
<th>Fever</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>D</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>E</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Healthy</td>
</tr>
</tbody>
</table>

\[
|\text{Cough} \cap \text{Flu}| = 1 \\
|\text{Flu}| = \\
\Pr(\text{Cough}|\text{Flu}) = \\
\Pr(\text{Flu}) = \\
\]
### Example

Consider the training set:

<table>
<thead>
<tr>
<th>Name</th>
<th>Cough</th>
<th>Sore Throat</th>
<th>Fever</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>D</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>E</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Healthy</td>
</tr>
</tbody>
</table>

|Cough ∩ Flu| = 1
|Flu| = 2

\[
\Pr(\text{Cough}|\text{Flu}) = \]

\[
\Pr(\text{Flu}) = \]
Training a Naive Bayes Model continued

Example

Consider the training set:

<table>
<thead>
<tr>
<th>Name</th>
<th>Cough</th>
<th>Sore Throat</th>
<th>Fever</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>D</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>E</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Healthy</td>
</tr>
</tbody>
</table>

|Cough ∩ Flu| = 1
|Flu| = 2

Pr(Cough|Flu) = \frac{|Cough ∩ Flu|}{|Flu|} = \frac{1}{2}

Pr(Flu) =
Consider the training set:

<table>
<thead>
<tr>
<th>Name</th>
<th>Cough</th>
<th>Sore Throat</th>
<th>Fever</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>D</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>E</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Healthy</td>
</tr>
</tbody>
</table>

\[ \text{Pr}(\text{Cough} \cap \text{Flu}) = \frac{|\text{Cough} \cap \text{Flu}|}{|\text{Flu}|} = \frac{1}{2} \]

\[ \text{Pr}(\text{Flu}) = \frac{|\text{Flu}|}{|\text{Patients}|} = \frac{2}{5} \]
Training a Naive Bayes Model continued

Example

<table>
<thead>
<tr>
<th>Name</th>
<th>Cough</th>
<th>Sore Throat</th>
<th>Fever</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>D</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>E</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Healthy</td>
</tr>
<tr>
<td>F</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>???</td>
</tr>
</tbody>
</table>

What is the probability new patient F is healthy?

\[
\begin{align*}
\Pr(\text{Cough}|\text{Healthy}) &= \\
\Pr(\text{Sore Throat}|\text{Healthy}) &= \\
\Pr(\text{Fever}|\text{Healthy}) &= \\
\Pr(\text{Healthy}) &= 
\end{align*}
\]
Training a Naive Bayes Model continued

**Example**

<table>
<thead>
<tr>
<th>Name</th>
<th>Cough</th>
<th>Sore Throat</th>
<th>Fever</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>D</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>E</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Healthy</td>
</tr>
<tr>
<td>F</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>???</td>
</tr>
</tbody>
</table>

What is the probability new patient F is healthy?

\[
\Pr(\text{Cough}|\text{Healthy}) = 0 \quad \Pr(\text{Sore Throat}|\text{Healthy}) = \\
\Pr(\text{Fever}|\text{Healthy}) = \quad \Pr(\text{Healthy}) =
\]

Problem: This model thinks that it’s impossible for a healthy person to have a sore throat.
### Example

<table>
<thead>
<tr>
<th>Name</th>
<th>Cough</th>
<th>Sore Throat</th>
<th>Fever</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>D</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>E</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Healthy</td>
</tr>
<tr>
<td>F</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>???</td>
</tr>
</tbody>
</table>

What is the probability new patient F is healthy?

\[
\begin{align*}
\Pr(\text{Cough}|\text{Healthy}) &= 0 \\
\Pr(\text{Sore Throat}|\text{Healthy}) &= 0 \\
\Pr(\text{Fever}|\text{Healthy}) &= \text{???} \\
\Pr(\text{Healthy}) &= \text{???}
\end{align*}
\]

Problem: This model thinks that it’s impossible for a healthy person to have a sore throat.
### Example

<table>
<thead>
<tr>
<th>Name</th>
<th>Cough</th>
<th>Sore Throat</th>
<th>Fever</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>B</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Flu</td>
</tr>
<tr>
<td>C</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>D</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Cold</td>
</tr>
<tr>
<td>E</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Healthy</td>
</tr>
<tr>
<td>F</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>??</td>
</tr>
</tbody>
</table>

What is the probability new patient F is healthy?

\[
\begin{align*}
\Pr(\text{Cough}|\text{Healthy}) &= 0 \\
\Pr(\text{Sore Throat}|\text{Healthy}) &= 0 \\
\Pr(\text{Fever}|\text{Healthy}) &= 0 \\
\Pr(\text{Healthy}) &= \frac{1}{5}
\end{align*}
\]

Problem: This model thinks that it’s impossible for a healthy person to have a sore throat.
What is the probability new patient F is healthy?

\[
\begin{align*}
\Pr(\text{Cough} | \text{Healthy}) &= 0 \\
\Pr(\text{Sore Throat} | \text{Healthy}) &= 0 \\
\Pr(\text{Fever} | \text{Healthy}) &= 0 \\
\Pr(\text{Healthy}) &= \frac{1}{5}
\end{align*}
\]

Problem: This model thinks that it's impossible for a healthy person to have a sore throat.
Solution: Let’s add an extra occurrence of everything when computing conditional probabilities! That way we won’t have anything with 0 probability.

**Example**

\[
\begin{align*}
|\text{Sore Throat} \cap \text{Healthy}| & = 0 \\
|\text{Healthy}| & = 1 \\
\Pr(\text{Sore Throat} | \text{Healthy}) & = \frac{|\text{Sore Throat} \cap \text{Healthy}|}{|\text{Healthy}|} = \frac{0}{1}
\end{align*}
\]
Solution: Let’s add an extra occurrence of everything when computing conditional probabilities! That way we won’t have anything with 0 probability.

Example

\[
\begin{align*}
|\text{Sore Throat} \cap \text{Healthy}| &= 0 \\
|\text{Healthy}| &= 1 \\
\Pr(\text{Sore Throat}|\text{Healthy}) &= \frac{|\text{Sore Throat} \cap \text{Healthy}| + 1}{|\text{Healthy}| + 2} = \frac{0 + 1}{1 + 2} = \frac{1}{3}
\end{align*}
\]
Solution: Let’s add an extra occurrence of everything when computing conditional probabilities! That way we won’t have anything with 0 probability.

Example

\[ |\text{Sore Throat} \cap \text{Healthy}| = 0 \]
\[ |\text{Healthy}| = 1 \]

\[ \Pr(\text{Sore Throat} | \text{Healthy}) = \frac{|\text{Sore Throat} \cap \text{Healthy}| + 1}{|\text{Healthy}| + 2} = \frac{0 + 1}{1 + 2} = \frac{1}{3} \]

We add 2 to the denominator [not 1!] because we added one extra patient who was healthy with a sore throat and another extra patient who was healthy and didn’t have a sore throat. That’s 2 total extra healthy people we’ve seen!
Go forth!

Time to try it yourself on the next homework! Make sure to read the Naive Bayes notes for some other computation tricks we have to worry about during implementation!

Have fun!