Naive Bayes Classifier

Section 1

Introduction

Classification: Definition

- Given a collection of records (training set)
  - Each record $(\mathbf{x}, y)$ contains a set of attributes/features/feature variables denoted as $\mathbf{x} \in \mathbb{R}^d$, and one target variable called class $y \in \{0, 1, \ldots, K-1\}$.
  - The entire training set can be denoted as $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$.

Classification: Definition

- Given a collection of records (training set)
  - Each record $(\mathbf{x}, y)$ contains a set of attributes/features/feature variables denoted as $\mathbf{x} \in \mathbb{R}^d$, and one target variable called class $y \in \{0, 1, \dots, K-1\}$.
  - The entire training set can be denoted as $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$.
- Find a model for class as a function of the values of other attributes.

Classification: Definition

- Given a collection of records (training set)
  - Each record $(\mathbf{x}, y)$ contains a set of attributes/features/feature variables denoted as $\mathbf{x} \in \mathbb{R}^d$, and one target variable called class $y \in \{0, 1, \ldots, K-1\}$.
  - The entire training set can be denoted as $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$.
- Find a model for class as a function of the values of other attributes.
- Goal: previously unseen records should be assigned a class as accurately as possible.

Classification: Definition

- Given a collection of records (training set)
  - Each record $(\mathbf{x}, y)$ contains a set of attributes/features/feature variables denoted as $\mathbf{x} \in \mathbb{R}^d$, and one target variable called class $y \in \{0, 1, \ldots, K-1\}$.
  - The entire training set can be denoted as $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$.
- Find a model for class as a function of the values of other attributes.
- Goal: previously unseen records should be assigned a class as accurately as possible.
  - A test set is used to determine the accuracy of the model. Usually, the given data set is divided into training and test sets, with training set used to build the model and test set used to validate it.

Different Approaches to Classification

- Construct a **discriminant function** which assigns each vector $\mathbf{x}$ to a specific class

Different Approaches to Classification

- Construct a **discriminant function** which assigns each vector $\mathbf{x}$ to a specific class
- Model the conditional probability distribution $p(y = k|\mathbf{x})$ in an inference stage, and use this distribution to make optimal decisions

Different Approaches to Classification

- Construct a **discriminant function** which assigns each vector $\mathbf{x}$ to a specific class
- Model the conditional probability distribution $p(y = k|\mathbf{x})$ in an inference stage, and use this distribution to make optimal decisions
  - Two methods to model $p(y = k|\mathbf{x})$
    - Discriminant method
      Example: Representing $p(y = k|\mathbf{x})$ as parametric models and then optimizing the parameters using a training set

Different Approaches to Classification

- Construct a **discriminant function** which assigns each vector $\mathbf{x}$ to a specific class
- Model the conditional probability distribution $p(y = k|\mathbf{x})$ in an inference stage, and use this distribution to make optimal decisions
  - Two methods to model $p(y = k|\mathbf{x})$
    - Discriminant method
      Example: Representing $p(y = k|\mathbf{x})$ as parametric models and then optimizing the parameters using a training set
    - Generative method
      Model the class-conditional densities $p(\mathbf{x}|y = k)$ and prior probabilities $p(y = k)$, and compute $p(y = k|\mathbf{x})$ using Bayes theorem

$$p(y = k|\mathbf{x}) = \frac{p(\mathbf{x}|y = k)p(y = k)}{p(\mathbf{x})}$$

Decision Rule for Generative Method

- Assign the class with largest probability to $\mathbf{x}$:

$$y(x) = \text{argmax}_k \left\{ p(y = k|\mathbf{x}) = \frac{p(\mathbf{x}|y = k)p(y = k)}{p(\mathbf{x})} \right\}$$

Decision Rule for Generative Method

- Assign the class with largest probability to $\mathbf{x}$:

$$y(x) = \operatorname{argmax}_k \left\{ p(y=k|\mathbf{x}) = \frac{p(\mathbf{x}|y=k)p(y=k)}{p(\mathbf{x})} \right\}$$

- Do we need to calculate $p(\mathbf{x})$?

Decision Rule for Generative Method

- Assign the class with largest probability to $\mathbf{x}$:

$$y(x) = \text{argmax}_k \left\{ p(y = k|\mathbf{x}) = \frac{p(\mathbf{x}|y = k)p(y = k)}{p(\mathbf{x})} \right\}$$

- Do we need to calculate $p(\mathbf{x})$?

$$y(x) = \text{argmax}_k \, p(\mathbf{x}|y = k)p(y = k)$$

Decision Rule for Generative Method

- Assign the class with largest probability to $\mathbf{x}$:

$$y(x) = \text{argmax}_k \left\{ p(y = k|\mathbf{x}) = \frac{p(\mathbf{x}|y = k)p(y = k)}{p(\mathbf{x})} \right\}$$

- Do we need to calculate $p(\mathbf{x})$?

$$y(x) = \text{argmax}_k \, p(\mathbf{x}|y = k)p(y = k)$$

We need to estimate $p(\mathbf{x}|y = k)$ and $p(y = k)$ for all k.

Section 2

Naive Bayes Classifier

An Example: Tennis

- Binary Classification: Play or Not ($y \in \{0, 1\}$).

An Example: Tennis

- Binary Classification: Play or Not ($y \in \{0, 1\}$).
- Categorical Features:
  - $\mathbf{x}[0]$: **O**utlook
    - S(unny)
    - R(ainy)
    - O(vercast)
  - $\mathbf{x}[1]$: **T**emperature
    - H(ot)
    - M(edium)
    - C(ool)
  - $\mathbf{x}[2]$: **H**umidity
    - H(igh)
    - N(ormal)
    - L(ow)
  - $\mathbf{x}[3]$: **W**ind
    - S(trong)
    - W(eak)

An Example: Tennis

- Binary Classification: Play or Not ($y \in \{0, 1\}$).
- Categorical Features:
  - $\mathbf{x}[0]$: **O**utlook
    - S(unny)
    - R(ainy)
    - O(vercast)
  - $\mathbf{x}[1]$: **T**emperature
    - H(ot)
    - M(edium)
    - C(ool)
  - $\mathbf{x}[2]$: **H**umidity
    - H(igh)
    - N(ormal)
    - L(ow)
  - $\mathbf{x}[3]$: **W**ind
    - S(trong)
    - W(eak)

	O	**T**	**H**	**W**	**Play?**
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

- Model $p(y)$ is easy
  - A single parameter: $p(y = 1) = \theta$
  - Why only 1 parameter?

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

- Model $p(y)$ is easy
  - A single parameter: $p(y = 1) = \theta$
  - Why only 1 parameter?
- How about $p(\mathbf{x}|y)$?
  - There are 4 features
  - How many possible assignments of $\mathbf{x}$?

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

- Model $p(y)$ is easy
  - A single parameter: $p(y = 1) = \theta$
  - Why only 1 parameter?
- How about $p(\mathbf{x}|y)$?
  - There are 4 features
  - How many possible assignments of $\mathbf{x}$?
    - $3 \times 3 \times 3 \times 2$
  - To model $p(\mathbf{x}|y = 1)$, we need a value to model the probability of each possible assignment of $\mathbf{x}$
    - How many parameters?

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

- Model $p(y)$ is easy
  - A single parameter: $p(y = 1) = \theta$
  - Why only 1 parameter?
- How about $p(\mathbf{x}|y)$?
  - There are 4 features
  - How many possible assignments of $\mathbf{x}$?
    - $3 \times 3 \times 3 \times 2$
  - To model $p(\mathbf{x}|y = 1)$, we need a value to model the probability of each possible assignment of $\mathbf{x}$
    - How many parameters? $3 \times 3 \times 3 \times 2 - 1$

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

More generally, assuming we have K labels in total

- We need $K - 1$ parameters for modelling $p(y)$
  - One parameter for modelling each $p(y = k)$
  - $\sum_{k=0}^{K-1} p(y = k) = 1$

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

More generally, assuming we have K labels in total

- We need $K - 1$ parameters for modelling $p(y)$
  - One parameter for modelling each $p(y = k)$
  - $\sum_{k=0}^{K-1} p(y = k) = 1$
- How about $p(\mathbf{x}|y)$?
  - Assuming there are d features and all of them are **Boolean**

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

More generally, assuming we have K labels in total

- We need $K - 1$ parameters for modelling $p(y)$
  - One parameter for modelling each $p(y = k)$
  - $\sum_{k=0}^{K-1} p(y = k) = 1$
- How about $p(\mathbf{x}|y)$?
  - Assuming there are d features and all of them are **Boolean**
  - How many possible assignments of $\mathbf{x}$?

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

More generally, assuming we have K labels in total

- We need $K - 1$ parameters for modelling $p(y)$
  - One parameter for modelling each $p(y = k)$
  - $\sum_{k=0}^{K-1} p(y = k) = 1$
- How about $p(\mathbf{x}|y)$?
  - Assuming there are d features and all of them are **Boolean**
  - How many possible assignments of $\mathbf{x}$?
    - 2^d

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

More generally, assuming we have K labels in total

- We need $K - 1$ parameters for modelling $p(y)$
  - One parameter for modelling each $p(y = k)$
  - $\sum_{k=0}^{K-1} p(y = k) = 1$
- How about $p(\mathbf{x}|y)$?
  - Assuming there are d features and all of them are **Boolean**
  - How many possible assignments of $\mathbf{x}$?
    - 2^d
  - For each class $y = k$, to model $p(\mathbf{x}|y = k)$, we need $2^d - 1$ parameters

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

More generally, assuming we have K labels in total

- We need $K - 1$ parameters for modelling $p(y)$
  - One parameter for modelling each $p(y = k)$
  - $\sum_{k=0}^{K-1} p(y = k) = 1$
- How about $p(\mathbf{x}|y)$?
  - Assuming there are d features and all of them are **Boolean**
  - How many possible assignments of $\mathbf{x}$?
    - 2^d
  - For each class $y = k$, to model $p(\mathbf{x}|y = k)$, we need $2^d - 1$ parameters
    - One parameter for each assignment of $\mathbf{x}$
    - Summation of the probability of all possible assignments equals to q

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

More generally, assuming we have K labels in total

- We need $K - 1$ parameters for modelling $p(y)$
  - One parameter for modelling each $p(y = k)$
  - $\sum_{k=0}^{K-1} p(y = k) = 1$
- How about $p(\mathbf{x}|y)$?
  - Assuming there are d features and all of them are **Boolean**
  - How many possible assignments of $\mathbf{x}$?
    - 2^d
  - For each class $y = k$, to model $p(\mathbf{x}|y = k)$, we need $2^d - 1$ parameters
    - One parameter for each assignment of $\mathbf{x}$
    - Summation of the probability of all possible assignments equals to q
  - In total, we need $K(2^d - 1)$ parameters

How Difficult is it to Model $p(\mathbf{x}|y = k)$?

More generally, assuming we have K labels in total

- We need $K - 1$ parameters for modelling $p(y)$
  - One parameter for modelling each $p(y = k)$
  - $\sum_{k=0}^{K-1} p(y = k) = 1$
- How about $p(\mathbf{x}|y)$?
  - Assuming there are d features and all of them are **Boolean**
  - How many possible assignments of $\mathbf{x}$?
    - 2^d
  - For each class $y = k$, to model $p(\mathbf{x}|y = k)$, we need $2^d - 1$ parameters
    - One parameter for each assignment of $\mathbf{x}$
    - Summation of the probability of all possible assignments equals to q
  - In total, we need $K(2^d - 1)$ parameters

Need a lot of data to estimate all these parameters

Recall: Conditional Independence

- Event A and B are conditionally independent given C in case

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

Recall: Conditional Independence

- Event A and B are conditionally independent given C in case

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

- A set of events $\{A_i\}$ is conditionally independent given C in case

$$p(\cup_i A_i|C) = \prod_i p(A_i|C)$$

Naive Bayes Assumption

Modeling $p(\mathbf{x}|y)$ requires $K(2^d - 1)$ parameters.
What if all the features were conditionally independent given the label?

Naive Bayes Assumption

Modeling $p(\mathbf{x}|y)$ requires $K(2^d - 1)$ parameters.

What if all the features were conditionally independent given the label?

- Naive Bayes Assumption

Naive Bayes Assumption

Modeling $p(\mathbf{x}|y)$ requires $K(2^d - 1)$ parameters.
What if all the features were conditionally independent given the label?

- Naive Bayes Assumption

$$p(\mathbf{x}|y) = p(\mathbf{x}[0]|y)p(\mathbf{x}[1]|y)\cdots p(\mathbf{x}[d-1]|y) = \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y)$$

Naive Bayes Assumption

Modeling $p(\mathbf{x}|y)$ requires $K(2^d - 1)$ parameters.
What if all the features were conditionally independent given the label?

- Naive Bayes Assumption

$$p(\mathbf{x}|y) = p(\mathbf{x}[0]|y)p(\mathbf{x}[1]|y) \cdots p(\mathbf{x}[d-1]|y) = \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y)$$

- For each class $y = k$, how many parameters are needed for modeling $p(\mathbf{x}[i]|y = k)$? (recall: we assume all features are Boolean)

Naive Bayes Assumption

Modeling $p(\mathbf{x}|y)$ requires $K(2^d - 1)$ parameters.
What if all the features were conditionally independent given the label?

- Naive Bayes Assumption

$$p(\mathbf{x}|y) = p(\mathbf{x}[0]|y)p(\mathbf{x}[1]|y) \cdots p(\mathbf{x}[d-1]|y) = \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y)$$

- For each class $y = k$, how many parameters are needed for modeling $p(\mathbf{x}[i]|y = k)$? (recall: we assume all features are Boolean)
    - One parameter

Naive Bayes Assumption

Modeling $p(\mathbf{x}|y)$ requires $K(2^d - 1)$ parameters.
What if all the features were conditionally independent given the label?

- Naive Bayes Assumption

$$p(\mathbf{x}|y) = p(\mathbf{x}[0]|y)p(\mathbf{x}[1]|y)\cdots p(\mathbf{x}[d-1]|y) = \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y)$$

- For each class $y = k$, how many parameters are needed for modeling $p(\mathbf{x}[i]|y = k)$? (recall: we assume all features are Boolean)
  - One parameter
- Kd parameters are needed to mode $p(\mathbf{x}|y)$ with the *Naive Bayes Assumption*
  - Much smaller than $K(2^d - 1)$

The Naive Bayes Classifier

- Assumption: Features are conditionally independent given the label y

$$p(\mathbf{x}|y) = \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y)$$

The Naive Bayes Classifier

- Assumption: Features are conditionally independent given the label y

$$p(\mathbf{x}|y) = \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y)$$

- Decision Rule

$$\begin{aligned} y(x) &= \mathrm{argmax}_k\, p(\mathbf{x}|y=k)p(y=k) \\ &= \mathrm{argmax}_k \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=k)p(y=k) \end{aligned}$$

The Naive Bayes Classifier

- Assumption: Features are conditionally independent given the label y

$$p(\mathbf{x}|y) = \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y)$$

- Decision Rule

$$\begin{aligned} y(x) &= \mathrm{argmax}_k \, p(\mathbf{x}|y=k)p(y=k) \\ &= \mathrm{argmax}_k \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=k)p(y=k) \end{aligned}$$

- We need to estimate $p(\mathbf{x}[i]|y=k)$ and $p(y=k)$ for all $i = 0, \ldots, d-1$ and $k = 0, \ldots, K$.

Section 3

Learning the Naive Bayes Classifier

How to Parameterize $p(\mathbf{x}[i]|y = k)$ and $p(y = k)$?

- $p(y = k) = \theta_k$ with $\sum_{k=0}^{K} \theta_k = 1$

How to Parameterize $p(\mathbf{x}[i]|y=k)$ and $p(y=k)$?

- $p(y=k) = \theta_k$ with $\sum_{k=0}^{K} \theta_k = 1$
- $p(\mathbf{x}[i]|y=k)$

How to Parameterize $p(\mathbf{x}[i]|y = k)$ and $p(y = k)$?

- $p(y = k) = \theta_k$ with $\sum_{k=0}^{K} \theta_k = 1$
- $p(\mathbf{x}[i]|y = k)$
  - Categorical features: $\mathbf{x}[i] \in \{0, 1, \ldots, J_i - 1\}$
    - $p(\mathbf{x}[i] = j|y = k) = \mathbf{w}_{ijk}$
    - $\sum_{j=0}^{J_i-1} \mathbf{w}_{i,j,k} = 1$

How to Parameterize $p(\mathbf{x}[i]|y=k)$ and $p(y=k)$?

- $p(y=k) = \theta_k$ with $\sum_{k=0}^{K} \theta_k = 1$
- $p(\mathbf{x}[i]|y=k)$
  - Categorical features: $\mathbf{x}[i] \in \{0, 1, \ldots, J_i - 1\}$
    - $p(\mathbf{x}[i] = j|y=k) = \mathbf{w}_{ijk}$
    - $\sum_{j=0}^{J_i-1} \mathbf{w}_{i,j,k} = 1$

How to Parameterize $p(\mathbf{x}[i]|y = k)$ and $p(y = k)$?

- $p(y = k) = \theta_k$ with $\sum_{k=0}^{K} \theta_k = 1$
- $p(\mathbf{x}[i]|y = k)$
  - Categorical features: $\mathbf{x}[i] \in \{0, 1, \ldots, J_i - 1\}$
    - $p(\mathbf{x}[i] = j|y = k) = \mathbf{w}_{ijk}$
    - $\sum_{j=0}^{J_i-1} \mathbf{w}_{i,j,k} = 1$
  - Continuous features: $\mathbf{x}[i] \in \mathcal{R}$
    - Model $p(\mathbf{x}[i]|y = k)$ with Gaussian distribution.

$$p(\mathbf{x}[i]|y = k) = \mathcal{N}(\mu_{ik}, \sigma^2_{ik})$$

How to Parameterize $p(\mathbf{x}[i]|y=k)$ and $p(y=k)$?

- $p(y=k)=\theta_k$ with $\sum_{k=0}^{K}\theta_k=1$
- $p(\mathbf{x}[i]|y=k)$
  - Categorical features: $\mathbf{x}[i]\in\{0,1,\ldots,J_i-1\}$
    - $p(\mathbf{x}[i]=j|y=k)=\mathbf{w}_{ijk}$
    - $\sum_{j=0}^{J_i-1}\mathbf{w}_{i,j,k}=1$
  - Continuous features: $\mathbf{x}[i]\in\mathcal{R}$
    - Model $p(\mathbf{x}[i]|y=k)$ with Gaussian distribution.

$$p(\mathbf{x}[i]|y=k)=\mathcal{N}(\mu_{ik},\sigma_{ik}^2)$$

- We need to estimate these parameters.

Estimation method : MLE

How to Parameterize $p(\mathbf{x}[i]|y = k)$ and $p(y = k)$?

- $p(y = k) = \theta_k$ with $\sum_{k=0}^{K} \theta_k = 1$
- $p(\mathbf{x}[i]|y = k)$
  - Categorical features: $\mathbf{x}[i] \in \{0, 1, \ldots, J_i - 1\}$
    - $p(\mathbf{x}[i] = j|y = k) = \mathbf{w}_{ijk}$
    - $\sum_{j=0}^{J_i-1} \mathbf{w}_{i,j,k} = 1$
  - Continuous features: $\mathbf{x}[i] \in \mathcal{R}$
    - Model $p(\mathbf{x}[i]|y = k)$ with Gaussian distribution.

$$p(\mathbf{x}[i]|y = k) = \mathcal{N}(\mu_{ik}, \sigma_{ik}^2)$$

- We need to estimate these parameters.
- For convenience, we summarize all parameters (for either case) as $\mathcal{W}$
  - Categorical features: $\mathcal{W} = \{\theta_k, \mathbf{w}_{ijk}\}$
  - Continuous features: $\mathcal{W} = \{\theta_k, \mu_{ik}, \sigma_{ik}\}$

Estimating the Parameters

We include $\mathcal{W}$ into the formulation of $p(y)$ and $p(\mathbf{x}[i]|y)$ to emphasize they are functions of $\mathcal{W}$:

$$p(y) = p(y; \mathcal{W})$$
$$p(\mathbf{x}[i]|y) = p(\mathbf{x}[i]|y; \mathcal{W})$$

Estimating the Parameters

We include $\mathcal{W}$ into the formulation of $p(y)$ and $p(\mathbf{x}[i]|y)$ to emphasize they are functions of $\mathcal{W}$:

$$p(y) = p(y; \mathcal{W})$$

$$p(\mathbf{x}[i]|y) = p(\mathbf{x}[i]|y; \mathcal{W})$$

Training set: $\mathcal{D} = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$ with N samples. The data points are i.i.d

Estimating the Parameters

We include $\mathcal{W}$ into the formulation of $p(y)$ and $p(\mathbf{x}[i]|y)$ to emphasize they are functions of $\mathcal{W}$:

$$p(y) = p(y; \mathcal{W})$$

$$p(\mathbf{x}[i]|y) = p(\mathbf{x}[i]|y; \mathcal{W})$$

Training set: $\mathcal{D} = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$ with N samples. The data points are i.i.d

Likelihood: (We denote $p(\mathbf{x} = \mathbf{x}_n, y = y_n)$ as $p(\mathbf{x}_n, y_n)$)

$$\begin{aligned} p(\mathcal{D}|\mathcal{W}) &= \prod_{n=1}^{N} p(\mathbf{x}_n, y_n; \mathcal{W}) \\ &= \prod_{n=1}^{N} p(\mathbf{x}_n|y_n; \mathcal{W}) p(y_n; \mathcal{W}) \\ &= \prod_{n=1}^{N} \left\{ p(y_n; \mathcal{W}) \prod_{i=0}^{d-1} p(\mathbf{x}_n[i]|y_n; \mathcal{W}) \right\} \end{aligned}$$

Maximum Likelihood Estimation

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \prod_{n=1}^{N} \left\{ p(y_n; \mathcal{W}) \prod_{i=0}^{d-1} p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Maximum Likelihood Estimation

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \prod_{n=1}^{N} \left\{ p(y_n; \mathcal{W}) \prod_{i=0}^{d-1} p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Again, take the logarithm

$$\begin{aligned}
\mathcal{W}_{ML} &= \text{argmax}_{\mathcal{W}} \sum_{n=1}^{N} \left\{ \log p(y_n; \mathcal{W}) + \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\} \\
&= \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}
\end{aligned}$$

MLE: Categorical Features

$$\mathcal{W}_{ML} = \mathrm{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

MLE: Categorical Features

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Recall

- $p(y = k) = \theta_k$ with $\sum_{k=0}^{K} \theta_k = 1$
- $p(\mathbf{x}[i] | y = k)$
  - Categorical features: $\mathbf{x}[i] \in \{0, 1, \ldots, J_i - 1\}$
    - $p(\mathbf{x}[i] = j | y = k) = \mathbf{w}_{ijk}$
    - $\sum_{j=0}^{J_i - 1} \mathbf{w}_{i,j,k} = 1$

MLE: Categorical Features

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Recall

- $p(y = k) = \theta_k$ with $\sum_{k=0}^{K} \theta_k = 1$
- $p(\mathbf{x}[i] | y = k)$
    - Categorical features: $\mathbf{x}[i] \in \{0, 1, \ldots, J_i - 1\}$
        - $p(\mathbf{x}[i] = j | y = k) = \mathbf{w}_{ijk}$
        - $\sum_{j=0}^{J_i - 1} \mathbf{w}_{i,j,k} = 1$

Let us consider a simple case for illustration

- Assume $y \in \{0, 1\}$: $p(y = 1) = \theta$ and $p(y = 0) = \theta$

MLE: Categorical Features

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Recall

- $p(y = k) = \theta_k$ with $\sum_{k=0}^{K} \theta_k = 1$
- $p(\mathbf{x}[i] | y = k)$
  - Categorical features: $\mathbf{x}[i] \in \{0, 1, \ldots, J_i - 1\}$
    - $p(\mathbf{x}[i] = j | y = k) = \mathbf{w}_{ijk}$
    - $\sum_{j=0}^{J_i - 1} \mathbf{w}_{i,j,k} = 1$

Let us consider a simple case for illustration

- Assume $y \in \{0, 1\}$: $p(y = 1) = \theta$ and $p(y = 0) = 1$-θ
- Assume all features are **Boolean**: $p(\mathbf{x}[i] = 1 | y = k) = w_{ik}$ and $p(\mathbf{x}[i] = 0 | y = k) = 1 - w_{ik}$

MLE: Categorical Features

$$\mathcal{W}_{ML} = \operatorname{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Let us consider a simple case for illustration

- Assume $y \in \{0, 1\}$: $p(y = 1) = \theta$ and $p(y = 0) = 1 - \theta$.
- Assume all features are **Boolean**, i.e, $\mathbf{x}[i] \in \{0, 1\}$:
  $p(\mathbf{x}[i] = 1 | y = k) = w_{ik}$ and $p(\mathbf{x}[i] = 0 | y = k) = 1 - w_{ik}$

MLE: Categorical Features

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Let us consider a simple case for illustration

- Assume $y \in \{0, 1\}$: $p(y = 1) = \theta$ and $p(y = 0) = 1 - \theta$.
- Assume all features are **Boolean**, i.e, $\mathbf{x}[i] \in \{0, 1\}$:
  $p(\mathbf{x}[i] = 1 | y = k) = w_{ik}$ and $p(\mathbf{x}[i] = 0 | y = k) = 1 - w_{ik}$

More concisely

$$p(y_n; \mathcal{W}) = \theta^{y_n} (1 - \theta)^{1 - y_n}$$

MLE: Categorical Features

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Let us consider a simple case for illustration

- Assume $y \in \{0, 1\}$: $p(y = 1) = \theta$ and $p(y = 0) = 1 - \theta$.
- Assume all features are **Boolean**, i.e, $\mathbf{x}[i] \in \{0, 1\}$: $p(\mathbf{x}[i] = 1 | y = k) = w_{ik}$ and $p(\mathbf{x}[i] = 0 | y = k) = 1 - w_{ik}$

More concisely

$$p(y_n; \mathcal{W}) = \theta^{y_n} (1 - \theta)^{1 - y_n}$$

$$p(\mathbf{x}_n[i] | y_n; \mathcal{W}) = \left(w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1 - \mathbf{x}_n[i]} \right)^{y_n} \left(w_{i0}^{\mathbf{x}_n[i]} (1 - w_{i0})^{1 - \mathbf{x}_n[i]} \right)^{1 - y_n}$$

MLE: Categorical Features

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Let us consider a simple case for illustration

- Assume $y \in \{0, 1\}$: $p(y = 1) = \theta$ and $p(y = 0) = 1 - \theta$.
- Assume all features are **Boolean**, i.e, $\mathbf{x}[i] \in \{0, 1\}$: $p(\mathbf{x}[i] = 1 | y = k) = w_{ik}$ and $p(\mathbf{x}[i] = 0 | y = k) = 1 - w_{ik}$

More concisely

$$p(y_n; \mathcal{W}) = \theta^{y_n} (1 - \theta)^{1 - y_n}$$

$$p(\mathbf{x}_n[i] | y_n; \mathcal{W}) = \left(w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1 - \mathbf{x}_n[i]} \right)^{y_n} \left(w_{i0}^{\mathbf{x}_n[i]} (1 - w_{i0})^{1 - \mathbf{x}_n[i]} \right)^{1 - y_n}$$

The two formulations depends on different parameters.

MLE: Categorical Features

$$\mathcal{W}_{ML} = \mathrm{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Let us consider a simple case for illustration

- Assume $y \in \{0, 1\}$: $p(y = 1) = \theta$ and $p(y = 0) = 1 - \theta$.
- Assume all features are **Boolean**, i.e, $\mathbf{x}[i] \in \{0, 1\}$: $p(\mathbf{x}[i] = 1 | y = k) = w_{ik}$ and $p(\mathbf{x}[i] = 0 | y = k) = 1 - w_{ik}$

More concisely

$$p(y_n; \mathcal{W}) = \theta^{y_n} (1 - \theta)^{1 - y_n}$$

$$p(\mathbf{x}_n[i] | y_n; \mathcal{W}) = \left(w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1 - \mathbf{x}_n[i]} \right)^{y_n} \left(w_{i0}^{\mathbf{x}_n[i]} (1 - w_{i0})^{1 - \mathbf{x}_n[i]} \right)^{1 - y_n}$$

The two formulations depends on different parameters.

Estimate them separately!

MLE: Categorical Features

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Let us consider a simple case for illustration

- Assume $y \in \{0, 1\}$: $p(y = 1) = \theta$ and $p(y = 0) = 1 - \theta$.
- Assume all features are **Boolean**, i.e, $\mathbf{x}[i] \in \{0, 1\}$: $p(\mathbf{x}[i] = 1 | y = k) = w_{ik}$ and $p(\mathbf{x}[i] = 0 | y = k) = 1 - w_{ik}$

More concisely

$$p(y_n; \theta) = \theta^{y_n} (1 - \theta)^{1 - y_n}$$

$$p(\mathbf{x}_n[i] | y_n; \{w_{i1}, w_{i0}\}) = \left(w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1 - \mathbf{x}_n[i]} \right)^{y_n} \left(w_{i0}^{\mathbf{x}_n[i]} (1 - w_{i0})^{1 - \mathbf{x}_n[i]} \right)$$

The two formulations depends on different parameters.

Estimate them separately!

Estimating θ for $p(y)$

$$p(y_n; \theta) = \theta^{y_n} (1 - \theta)^{1 - y_n}$$

Estimating θ for $p(y)$

$$p(y_n;\theta) = \theta^{y_n}(1-\theta)^{1-y_n}$$

$$\begin{aligned}
\theta_{ML} &= \operatorname{argmax}_\theta \sum_{n=1}^{N} \log p(y_n;\theta) \\
&= \operatorname{argmax}_\theta \sum_{n=1}^{N} \log p(y_n;\theta) \\
&= \operatorname{argmax}_\theta \sum_{n=1}^{N} \log\left(\theta^{y_n}(1-\theta)^{1-y_n}\right) \\
&= \operatorname{argmax}_\theta \sum_{n=1}^{N} \left(y_n \log\theta + (1-y_n)\log(1-\theta)\right)
\end{aligned}$$

Estimating θ for $p(y)$

$$\begin{aligned}
\mathcal{L}(\theta) &= \sum_{n=1}^{N} \left(y_n \log \theta + (1 - y_n) \log(1 - \theta)\right) \\
&= \log \theta \sum_{n=1}^{N} y_n + \log(1 - \theta) \sum_{n=1}^{N} (1 - y_n)
\end{aligned}$$

Estimating θ for $p(y)$

$$\begin{aligned}\mathcal{L}(\theta) &= \sum_{n=1}^{N} (y_n \log \theta + (1 - y_n) \log(1 - \theta)) \\ &= \log \theta \sum_{n=1}^{N} y_n + \log(1 - \theta) \sum_{n=1}^{N} (1 - y_n)\end{aligned}$$

We aim to maximize $\mathcal{L}(\theta)$ with respect to θ.

- Calculate the derivative:

$$\frac{d\mathcal{L}(\theta)}{d\theta} = \frac{1}{\theta} \sum_{n=1}^{N} (y_n) - \frac{1}{1 - \theta} \sum_{n=1}^{N} (1 - y_n)$$

Estimating θ for $p(y)$

$$\begin{aligned}\mathcal{L}(\theta) &= \sum_{n=1}^{N} \left(y_n \log \theta + (1 - y_n) \log(1 - \theta)\right) \\ &= \log \theta \sum_{n=1}^{N} y_n + \log(1 - \theta) \sum_{n=1}^{N} (1 - y_n)\end{aligned}$$

We aim to maximize $\mathcal{L}(\theta)$ with respect to θ.

- Calculate the derivative:

$$\frac{d\mathcal{L}(\theta)}{d\theta} = \frac{1}{\theta} \sum_{n=1}^{N} (y_n) - \frac{1}{1-\theta} \sum_{n=1}^{N} (1 - y_n)$$

- Set the derivative to 0:

$$\theta_{ML} = \frac{1}{N} \sum_{n=1}^{N} y_n = \frac{\#\text{Samples with } y_n = 1}{N}$$

Estimating w_{i1}, w_{i0}

Recall

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Estimating w_{i1}, w_{i0}

Recall

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Now we focus on maximizing the second part.

Estimating w_{i1}, w_{i0}

Recall

$$\mathcal{W}_{ML} = \mathrm{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Now we focus on maximizing the second part.

$$p(\mathbf{x}_n[i] | y_n; \mathcal{W}) = \left(\underbrace{w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1-\mathbf{x}_n[i]}}_{f_n(w_{i1})} \right)^{y_n} \left(\underbrace{w_{i0}^{\mathbf{x}_n[i]} (1 - w_{i0})^{1-\mathbf{x}_n[i]}}_{f_n(w_{i0})} \right)^{1-y_n}$$

Estimating w_{i1}, w_{i0}

Recall

$$\mathcal{W}_{ML} = \text{argmax}_{\mathcal{W}} \left\{ \sum_{n=1}^{N} \log p(y_n; \mathcal{W}) + \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log p(\mathbf{x}_n[i] | y_n; \mathcal{W}) \right\}$$

Now we focus on maximizing the second part.

$$p(\mathbf{x}_n[i] | y_n; \mathcal{W}) = \left(\underbrace{w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1-\mathbf{x}_n[i]}}_{f_n(w_{i1})} \right)^{y_n} \left(\underbrace{w_{i0}^{\mathbf{x}_n[i]} (1 - w_{i0})^{1-\mathbf{x}_n[i]}}_{f_n(w_{i0})} \right)^{1-y_n}$$

The second part:

$$\mathcal{L}(\mathcal{W}) = \sum_{n=1}^{N} \sum_{i=0}^{d-1} \log \left(f_n^{y_n}(w_{i1}) f^{(1-y_n)}(w_{i0}) \right)$$

Estimating w_{i1}, w_{i0}

$$
\begin{aligned}
\mathcal{L}(\mathcal{W}) &= \sum_{n=1}^{N}\sum_{i=0}^{d-1} \log\left(f_n^{y_n}(w_{i1}) f^{(1-y_n)}(w_{i0})\right) \\
&= \sum_{i=0}^{d-1}\sum_{n=1}^{N} \log\left(f_n^{y_n}(w_{i1}) f^{(1-y_n)}(w_{i0})\right) \\
&= \sum_{i=0}^{d-1}\sum_{n=1}^{N} (y_n \log f_n(w_{i1}) + (1-y_n)\log f_n(w_{i0})) \\
&= \sum_{i=0}^{d-1}(\underbrace{\sum_{n=1}^{N} y_n \log f_n(w_{i1})}_{\mathcal{L}_{i,1}(w_{i1})} + \underbrace{\sum_{n=1}^{N}(1-y_n)\log f_n(w_{i0}))}_{\mathcal{L}_{i,0}(w_{i,0})}
\end{aligned}
$$

Estimating w_{i1}, w_{i0}

$$\mathcal{L}(\mathcal{W}) = \sum_{n=1}^{N}\sum_{i=0}^{d-1} \log\left(f_n^{y_n}(w_{i1}) f^{(1-y_n)}(w_{i0})\right)$$

$$= \sum_{i=0}^{d-1}\sum_{n=1}^{N} \log\left(f_n^{y_n}(w_{i1}) f^{(1-y_n)}(w_{i0})\right)$$

$$= \sum_{i=0}^{d-1}\sum_{n=1}^{N} (y_n \log f_n(w_{i1}) + (1-y_n)\log f_n(w_{i0}))$$

$$= \sum_{i=0}^{d-1}(\underbrace{\sum_{n=1}^{N} y_n \log f_n(w_{i1})}_{\mathcal{L}_{i,1}(w_{i1})} + \underbrace{\sum_{n=1}^{N}(1-y_n)\log f_n(w_{i0}))}_{\mathcal{L}_{i,0}(w_{i,0})}$$

- Maximize for each feature separately

Estimating w_{i1}, w_{i0}

$$
\begin{aligned}
\mathcal{L}(\mathcal{W}) &= \sum_{n=1}^{N}\sum_{i=0}^{d-1}\log\left(f_n^{y_n}(w_{i1})f^{(1-y_n)}(w_{i0})\right) \\
&= \sum_{i=0}^{d-1}\sum_{n=1}^{N}\log\left(f_n^{y_n}(w_{i1})f^{(1-y_n)}(w_{i0})\right) \\
&= \sum_{i=0}^{d-1}\sum_{n=1}^{N}\left(y_n \log f_n(w_{i1}) + (1-y_n)\log f_n(w_{i0})\right) \\
&= \sum_{i=0}^{d-1}\Big(\underbrace{\sum_{n=1}^{N} y_n \log f_n(w_{i1})}_{\mathcal{L}_{i,1}(w_{i1})} + \underbrace{\sum_{n=1}^{N}(1-y_n)\log f_n(w_{i0})\Big)}_{\mathcal{L}_{i,0}(w_{i,0})}
\end{aligned}
$$

- Maximize for each feature separately
- For each feature, maximize $\mathcal{L}_{i,1}(w_{i,1})$ and $\mathcal{L}_{i,0}(w_{i,0})$ separately

Estimating w_{i1}

$$\mathcal{L}_{i,1}(w_{i,1}) = \sum_{n=1}^{N} y_n \log f_n(w_{i1}) \text{ with } f_n(w_{i1}) = w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1-\mathbf{x}_n[i]}$$

Estimating w_{i1}

$$\mathcal{L}_{i,1}(w_{i,1}) = \sum_{n=1}^{N} y_n \log f_n(w_{i1}) \text{ with } f_n(w_{i1}) = w_{i1}^{\mathbf{x}_n[i]}(1 - w_{i1})^{1-\mathbf{x}_n[i]}$$

Estimating w_{i1}

$$\mathcal{L}_{i,1}(w_{i,1}) = \sum_{n=1}^{N} y_n \log f_n(w_{i1}) \text{ with } f_n(w_{i1}) = w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1-\mathbf{x}_n[i]}$$

Hence,

$$\begin{aligned} \mathcal{L}_{i,1}(w_{i,1}) &= \sum_{n=1}^{N} y_n \log \left(w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1-\mathbf{x}_n[i]} \right) \\ &= \sum_{n=1}^{N} y_n \left\{ \mathbf{x}_n[i] \log w_{i1} + (1 - \mathbf{x}_n[i]) \log(1 - w_{i1}) \right\} \end{aligned}$$

Estimating w_{i1}

$$\mathcal{L}_{i,1}(w_{i,1}) = \sum_{n=1}^{N} y_n \log f_n(w_{i1}) \text{ with } f_n(w_{i1}) = w_{i1}^{\mathbf{x}_n[i]} \left(1 - w_{i1}\right)^{1-\mathbf{x}_n[i]}$$

Hence,

$$\begin{aligned}\mathcal{L}_{i,1}(w_{i,1}) &= \sum_{n=1}^{N} y_n \log \left(w_{i1}^{\mathbf{x}_n[i]} \left(1 - w_{i1}\right)^{1-\mathbf{x}_n[i]} \right) \\ &= \sum_{n=1}^{N} y_n \left\{ \mathbf{x}_n[i] \log w_{i1} + (1 - \mathbf{x}_n[i]) \log(1 - w_{i1}) \right\}\end{aligned}$$

Does it look familiar?

Estimating w_{i1}

$$\mathcal{L}_{i,1}(w_{i,1}) = \sum_{n=1}^{N} y_n \log f_n(w_{i1}) \text{ with } f_n(w_{i1}) = w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1-\mathbf{x}_n[i]}$$

Hence,

$$\begin{aligned}\mathcal{L}_{i,1}(w_{i,1}) &= \sum_{n=1}^{N} y_n \log \left(w_{i1}^{\mathbf{x}_n[i]} (1 - w_{i1})^{1-\mathbf{x}_n[i]} \right) \\ &= \sum_{n=1}^{N} y_n \left\{ \mathbf{x}_n[i] \log w_{i1} + (1 - \mathbf{x}_n[i]) \log(1 - w_{i1}) \right\}\end{aligned}$$

Does it look familiar? Recall

$$\mathcal{L}(\theta) = \sum_{n=1}^{N} (y_n \log \theta + (1 - y_n) \log(1 - \theta))$$

Estimating w_{i1}

Maximizing

$$\mathcal{L}_{i,1}(w_{i,1}) = \sum_{n=1}^{N} y_n \left\{ \mathbf{x}_n[i] \log w_{i1} + (1 - \mathbf{x}_n[i]) \log(1 - w_{i1}) \right\}$$

The solution:

$$w_{i1\,ML} = \frac{\sum_{n=1}^{N} y_n \mathbf{x}_n[i]}{\sum_{n=1}^{N} y_n} = \frac{\#\text{Samples with } \mathbf{x}_n[i] = 1 \text{ and } y_n = 1}{\#\text{Samples with } y_n = 1}$$

Similarly for Estimating w_{i0}

Maximizing

$$\mathcal{L}_{i,1}(w_{i,0}) = \sum_{n=1}^{N} (1 - y_n) \left\{ \mathbf{x}_n[i] \log w_{i0} + (1 - \mathbf{x}_n[i]) \log(1 - w_{i0}) \right\}$$

The solution:

$$w_{i0\,ML} = \frac{\sum_{n=1}^{N} (1 - y_n)\mathbf{x}_n[i]}{\sum_{n=1}^{N} (1 - y_n)} = \frac{\#\text{Samples with } \mathbf{x}_n[i] = 1 \text{ and } y_n = 0}{\#\text{Samples with } y_n = 0}$$

In Summary

$$\theta_{ML} = \frac{1}{N}\sum_{n=1}^{N} y_n = \frac{\#\text{Samples with } y_n = 1}{N}$$

$$w_{i1\,ML} = \frac{\sum_{n=1}^{N} y_n \mathbf{x}_n[i]}{\sum_{n=1}^{N} y_n} = \frac{\#\text{Samples with } \mathbf{x}_n[i] = 1 \text{ and } y_n = 1}{\#\text{Samples with } y_n = 1}$$

$$w_{i0\,ML} = \frac{\sum_{n=1}^{N} (1 - y_n)\mathbf{x}_n[i]}{\sum_{n=1}^{N} (1 - y_n)} = \frac{\#\text{Samples with } \mathbf{x}_n[i] = 1 \text{ and } y_n = 0}{\#\text{Samples with } y_n = 0}$$

Naive Bayes with Categorical Features

- Learning
  - Calculate fraction of Samples with each label
  - Count how often features occur with each label.
  - It can generalize to multi-classes
- Prediction: Use learned probabilities to find highest scoring label

Back to the Example: Tennis

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

- $p(y=1) = \frac{9}{14}; p(y=0) = \frac{5}{14}$

Back to the Example: Tennis

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

- $p(y = 1) = \frac{9}{14}; p(y = 0) = \frac{5}{14}$
- $p(\mathbf{O}|y = 1)$
  - $p(\mathbf{O} = S|y = 1) = \frac{2}{9}$

Back to the Example: Tennis

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

- $p(y=1) = \frac{9}{14}; p(y=0) = \frac{5}{14}$
- $p(\mathbf{O}|y=1)$
  - $p(\mathbf{O} = S|y=1) = \frac{2}{9}$
  - $p(\mathbf{O} = R|y=1) = \frac{3}{9}$

Back to the Example: Tennis

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

- $p(y=1) = \frac{9}{14}; p(y=0) = \frac{5}{14}$
- $p(\mathbf{O}|y=1)$
  - $p(\mathbf{O}=S|y=1) = \frac{2}{9}$
  - $p(\mathbf{O}=R|y=1) = \frac{3}{9}$
  - $p(\mathbf{O}=O|y=1) = \frac{4}{9}$

Back to the Example: Tennis

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$\mathbf{x} = [\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2], \mathbf{x}[3]]$$

- $p(y=1) = \frac{9}{14}; p(y=0) = \frac{5}{14}$
- $p(\mathbf{O}|y=1)$
  - $p(\mathbf{O}=S|y=1) = \frac{2}{9}$
  - $p(\mathbf{O}=R|y=1) = \frac{3}{9}$
  - $p(\mathbf{O}=O|y=1) = \frac{4}{9}$
- And so on, for other attributes and also for $y = 0$...

Decision Boundary

Decision Rule

$$
\begin{aligned}
y(x) &= \operatorname{argmax}_k p(\mathbf{x}|y=k)p(y=k) \\
&= \operatorname{argmax}_k \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=k)p(y=k)
\end{aligned}
$$

Decision Boundary

Decision Rule

$$
\begin{aligned}
y(x) &= \operatorname{argmax}_k p(\mathbf{x}|y=k)p(y=k) \\
&= \operatorname{argmax}_k \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=k)p(y=k)
\end{aligned}
$$

Decision Boundary (for the binary classification case, i.e, $K = 2$) is defined by

$$
\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=1)p(y=1) = \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=0)p(y=0)
$$

Decision Boundary

Decision Rule

$$y(x) = \text{argmax}_k \, p(\mathbf{x}|y=k)p(y=k)$$
$$= \text{argmax}_k \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=k)p(y=k)$$

Decision Boundary (for the binary classification case, i.e, $K = 2$) is defined by

$$\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=1)p(y=1) = \prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=0)p(y=0)$$

$$\Leftrightarrow \frac{\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=1)p(y=1)}{\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=0)p(y=0)} = 1$$

Decision Boundary

$$\frac{\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=1)p(y=1)}{\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=0)p(y=0)} = 1 \tag{1}$$

Decision Boundary

$$\frac{\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=1)p(y=1)}{\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=0)p(y=0)} = 1 \tag{1}$$

Recall

- $y \in \{0,1\}$: $p(y=1) = \theta$ and $p(y=0) = 1-\theta$
- Assume all features are **Boolean**: $p(\mathbf{x}[i] = 1|y=k) = w_{ik}$ and $p(\mathbf{x}[i] = 0|y=k) = 1 - w_{ik}$

$$p(\mathbf{x}_n[i]|y) = \left(w_{i1}^{\mathbf{x}[i]}(1-w_{i1})^{1-\mathbf{x}[i]}\right)^y \left(w_{i0}^{\mathbf{x}[i]}(1-w_{i0})^{1-\mathbf{x}[i]}\right)^{1-y}$$

Decision Boundary

$$\frac{\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=1)p(y=1)}{\prod_{i=0}^{d-1} p(\mathbf{x}[i]|y=0)p(y=0)} = 1 \tag{1}$$

Recall

- $y \in \{0,1\}$: $p(y=1) = \theta$ and $p(y=0) = 1-\theta$
- Assume all features are **Boolean**: $p(\mathbf{x}[i]=1|y=k) = w_{ik}$ and $p(\mathbf{x}[i]=0|y=k) = 1-w_{ik}$

$$p(\mathbf{x}_n[i]|y) = \left(w_{i1}^{\mathbf{x}[i]}(1-w_{i1})^{1-\mathbf{x}[i]}\right)^y \left(w_{i0}^{\mathbf{x}[i]}(1-w_{i0})^{1-\mathbf{x}[i]}\right)^{1-y}$$

Substitute them into Eq. (1):

$$\frac{\theta\prod_{i=0}^{d-1}\left(w_{i1}^{\mathbf{x}[i]}(1-w_{i1})^{1-\mathbf{x}[i]}\right)}{(1-\theta)\prod_{i=0}^{d-1}\left(w_{i0}^{\mathbf{x}[i]}(1-w_{i0})^{1-\mathbf{x}[i]}\right)} = 1 \tag{2}$$

Decision Boundary

$$\frac{\theta \prod_{i=0}^{d-1}\left(w_{i1}^{\mathbf{x}[i]}(1-w_{i1})^{1-\mathbf{x}[i]}\right)}{(1-\theta)\prod_{i=0}^{d-1}\left(w_{i0}^{\mathbf{x}[i]}(1-w_{i0})^{1-\mathbf{x}[i]}\right)} = 1 \tag{3}$$

Decision Boundary

$$\frac{\theta \prod_{i=0}^{d-1} \left(w_{i1}^{\mathbf{x}[i]} (1 - w_{i1})^{1-\mathbf{x}[i]} \right)}{(1-\theta) \prod_{i=0}^{d-1} \left(w_{i0}^{\mathbf{x}[i]} (1 - w_{i0})^{1-\mathbf{x}[i]} \right)} = 1 \tag{3}$$

Collect the constants together:

$$\left(\frac{\theta}{1-\theta} \prod_{j=0}^{d-1} \frac{1 - w_{i1}}{1 - w_{i0}} \right) \cdot \prod_{i=0}^{d-1} \left(\frac{w_{i1}}{w_{i0}} \cdot \frac{1 - w_{i0}}{1 - w_{i1}} \right)^{\mathbf{x}[i]} = 1$$

Decision Boundary

$$\frac{\theta \prod_{i=0}^{d-1} \left(w_{i1}^{\mathbf{x}[i]} (1 - w_{i1})^{1-\mathbf{x}[i]} \right)}{(1-\theta) \prod_{i=0}^{d-1} \left(w_{i0}^{\mathbf{x}[i]} (1 - w_{i0})^{1-\mathbf{x}[i]} \right)} = 1 \tag{3}$$

Collect the constants together:

$$\left(\frac{\theta}{1-\theta} \prod_{j=0}^{d-1} \frac{1 - w_{i1}}{1 - w_{i0}} \right) \cdot \prod_{i=0}^{d-1} \left(\frac{w_{i1}}{w_{i0}} \cdot \frac{1 - w_{i0}}{1 - w_{i1}} \right)^{\mathbf{x}[i]} = 1$$

Take logarithm:

$$\log \left(\frac{\theta}{1-\theta} \prod_{j=0}^{d-1} \frac{1 - w_{i1}}{1 - w_{i0}} \right) + \sum_{i=0}^{d-1} \log \left(\frac{w_{i1}}{w_{i0}} \cdot \frac{1 - w_{i0}}{1 - w_{i1}} \right) \cdot \mathbf{x}[i] = 1$$

Decision Boundary

$$\log \left(\frac{\theta}{1-\theta} \prod_{j=0}^{d-1} \frac{1-w_{i1}}{1-w_{i0}} \right) + \sum_{i=0}^{d-1} \log \left(\frac{w_{i1}}{w_{i0}} \cdot \frac{1-w_{i0}}{1-w_{i1}} \right) \cdot \mathbf{x}[i] = 1 \tag{4}$$

Decision Boundary

$$\log\left(\frac{\theta}{1-\theta}\prod_{j=0}^{d-1}\frac{1-w_{i1}}{1-w_{i0}}\right)+\sum_{i=0}^{d-1}\log\left(\frac{w_{i1}}{w_{i0}}\cdot\frac{1-w_{i0}}{1-w_{i1}}\right)\cdot\mathbf{x}[i]=1 \qquad (4)$$

Denote:

$$b=\log\left(\frac{\theta}{1-\theta}\prod_{j=0}^{d-1}\frac{1-w_{i1}}{1-w_{i0}}\right)$$

$$\mathbf{w}_i=\log\left(\frac{w_{i1}}{w_{i0}}\cdot\frac{1-w_{i0}}{1-w_{i1}}\right)$$

Decision Boundary

$$\log\left(\frac{\theta}{1-\theta}\prod_{j=0}^{d-1}\frac{1-w_{i1}}{1-w_{i0}}\right)+\sum_{i=0}^{d-1}\log\left(\frac{w_{i1}}{w_{i0}}\cdot\frac{1-w_{i0}}{1-w_{i1}}\right)\cdot\mathbf{x}[i]=1 \qquad (4)$$

Denote:

$$b=\log\left(\frac{\theta}{1-\theta}\prod_{j=0}^{d-1}\frac{1-w_{i1}}{1-w_{i0}}\right)$$

$$\mathbf{w}_i=\log\left(\frac{w_{i1}}{w_{i0}}\cdot\frac{1-w_{i0}}{1-w_{i1}}\right)$$

Rewrite Eq.(4) as

$$b+\sum_{i=0}^{d-1}\mathbf{w}_i\mathbf{x}[i]=1$$

Decision Boundary

$$\log\left(\frac{\theta}{1-\theta}\prod_{j=0}^{d-1}\frac{1-w_{i1}}{1-w_{i0}}\right)+\sum_{i=0}^{d-1}\log\left(\frac{w_{i1}}{w_{i0}}\cdot\frac{1-w_{i0}}{1-w_{i1}}\right)\cdot\mathbf{x}[i]=1 \tag{4}$$

Denote:

$$b=\log\left(\frac{\theta}{1-\theta}\prod_{j=0}^{d-1}\frac{1-w_{i1}}{1-w_{i0}}\right)$$

$$\mathbf{w}_i=\log\left(\frac{w_{i1}}{w_{i0}}\cdot\frac{1-w_{i0}}{1-w_{i1}}\right)$$

Rewrite Eq.(4) as

$$b+\sum_{i=0}^{d-1}\mathbf{w}_i\mathbf{x}[i]=1$$

The decision boundary is linear

Section 4

Practical Concerns

Important Caveats with Naive Bayes

- Features need not be conditionally independent given the label
  - The Naive Bayes assumption does not always hold
  - And yet, very often used in practice because of simplicity
  - Works reasonably well even when the assumption is violated
- Not enough training data to get good estimates of the probabilities from counts
  - What if we never see a particular feature with a particular label?

Example: Spam Filtering

- Samples: Text documents (such as email)
- Labels: Spam or NotSpam

Example: Spam Filtering

- Samples: Text documents (such as email)
- Labels: Spam or NotSpam

Goal: To learn a function that can predict whether a new document is Spam or NotSpam

How to build a Naive Bayes Classifier?

Example: Spam Filtering

- Samples: Text documents (such as email)
- Labels: Spam or NotSpam

Goal: To learn a function that can predict whether a new document is Spam or NotSpam

How to build a Naive Bayes Classifier?

- How to represent documents?
- How to estimate probabilities?

Example: Spam Filtering

Represent documents by a vector of words

- Each feature is corresponding to a word in the vocabulary $\mathcal{V}$
- Each feature is **Boolean**: whether the word appear in the document
- Total number of features:

Example: Spam Filtering

Represent documents by a vector of words

- Each feature is corresponding to a word in the vocabulary $\mathcal{V}$
- Each feature is **Boolean**: whether the word appear in the document
- Total number of features: the size of vocabulary $|\mathcal{V}|$

Example: Spam Filtering

Represent documents by a vector of words

- Each feature is corresponding to a word in the vocabulary $\mathcal{V}$
- Each feature is **Boolean**: whether the word appear in the document
- Total number of features: the size of vocabulary $|\mathcal{V}|$

Learning from N labeled documents

Example: Spam Filtering

Represent documents by a vector of words

- Each feature is corresponding to a word in the vocabulary $\mathcal{V}$
- Each feature is **Boolean**: whether the word appear in the document
- Total number of features: the size of vocabulary $|\mathcal{V}|$

Learning from N labeled documents

- Estimating $p(y)$:

$$P(\text{Spam}) = \frac{\text{Count }(\text{Spam})}{N}; P(\text{NotSpam}) = 1 - P(\text{Spam})$$

Example: Spam Filtering

Represent documents by a vector of words

- Each feature is corresponding to a word in the vocabulary $\mathcal{V}$
- Each feature is **Boolean**: whether the word appear in the document
- Total number of features: the size of vocabulary $|\mathcal{V}|$

Learning from N labeled documents

- Estimating $p(y)$:

$$P(\text{Spam}) = \frac{\text{Count }(\text{Spam})}{N}; P(\text{NotSpam}) = 1 - P(\text{Spam})$$

- Estimating $p(\mathbf{x}[i]|y)$ (for all i); assume the i-th feature is corresponding to the word $v \in \mathcal{V}$

$$P(v|\text{Spam}) = \frac{\text{Count }(v, \text{Spam})}{\text{Count }(\text{Spam})}$$

$$P(v|\text{NotSpam}) = \frac{\text{Count }(v, \text{NotSpam})}{\text{Count }(\text{NotSpam})}$$

Smoothing

$$P(v|\text{Spam}) = \frac{\text{Count }(v, \text{Spam})}{\text{Count }(\text{Spam})}$$

$$P(v|\text{NotSpam}) = \frac{\text{Count }(v, \text{NotSpam})}{\text{Count }(\text{NotSpam})}$$

Smoothing

$$P(v|\text{Spam}) = \frac{\text{Count}\ (v, \text{Spam})}{\text{Count}\ (\text{Spam})}$$

$$P(v|\text{NotSpam}) = \frac{\text{Count}\ (v, \text{NotSpam})}{\text{Count}\ (\text{NotSpam})}$$

What if there are some words never appeared in the training data?

Smoothing

$$P(v|\text{Spam}) = \frac{\text{Count }(v, \text{Spam})}{\text{Count }(\text{Spam})}$$

$$P(v|\text{NotSpam}) = \frac{\text{Count }(v, \text{NotSpam})}{\text{Count }(\text{NotSpam})}$$

What if there are some words never appeared in the training data?
$P(v|\text{Spam})$ and $P(v|\text{NotSpam})$ will both be zero!

Smoothing

$$P(v|\text{Spam}) = \frac{\text{Count }(v, \text{Spam})}{\text{Count }(\text{Spam})}$$
$$P(v|\text{NotSpam}) = \frac{\text{Count }(v, \text{NotSpam})}{\text{Count }(\text{NotSpam})}$$

What if there are some words never appeared in the training data?
$P(v|\text{Spam})$ and $P(v|\text{NotSpam})$ will both be zero!

Solution: Smoothing

- Add pseudocounts (α) to each word in the vocabulary (very small numbers so that the counts are not zero)

Smoothing

$$P(v|\text{Spam}) = \frac{\text{Count }(v, \text{Spam})}{\text{Count }(\text{Spam})}$$
$$P(v|\text{NotSpam}) = \frac{\text{Count }(v, \text{NotSpam})}{\text{Count }(\text{NotSpam})}$$

What if there are some words never appeared in the training data?
$P(v|\text{Spam})$ and $P(v|\text{NotSpam})$ will both be zero!

Solution: Smoothing

- Add pseudocounts (α) to each word in the vocabulary (very small numbers so that the counts are not zero)

$$P(v|\text{Spam}) = \frac{\text{Count }(v, \text{Spam}) + \alpha}{\text{Count }(\text{Spam}) + \alpha \cdot |\mathcal{V}|}$$
$$P(v|\text{NotSpam}) = \frac{\text{Count }(v, \text{NotSpam}) + \alpha}{\text{Count }(\text{NotSpam}) + \alpha \cdot |\mathcal{V}|}$$

Section 5

Additional Notes

MLE: Continuous Features

- $p(\mathbf{x}[i]|y=k)$
  - Continuous features: $\mathbf{x}[i] \in \mathcal{R}$
    - Model $p(\mathbf{x}[i]|y=k)$ with Gaussian distribution.

$$p(\mathbf{x}[i]|y=k) = \mathcal{N}(\mu_{ik}, \sigma_{ik}^2)$$

- Estimating μ_{ik} and σ_{ik}^2
  - Estimating for each feature separately, i.e, each i.
  - MLE solution:

$$\mu_{ik}^* = \frac{1}{\sum_{n=1}^N \mathbb{1}\left(y_n = k\right)} \sum_{n=1}^N \mathbf{x}_n[i] \mathbb{1} \cdot \left(y_n = k\right)$$

$$\sigma_{ik}^{*\,2} = \frac{1}{\sum_{n=1}^N \mathbb{1}\left(y_n = k\right)} \sum_{n=1}^N \left(\mathbf{x}_n[i] - \mu_{ik}^*\right)^2 \cdot \mathbb{1}\left(y_n = k\right)$$

  - Check more details at `https://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf`