Probabilistic Models

Shan-Hung Wu
shwu@cs.nthu.edu.tw

Department of Computer Science,
National Tsing Hua University, Taiwan

Machine Learning

Outline

Outline

Predictions based on Probability

- Supervised learning, we are given a training set $\mathbb{X} = \{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}_{i=1}^{N}$

Predictions based on Probability

- Supervised learning, we are given a training set $\mathbb{X} = \{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}_{i=1}^{N}$
- Model $\mathbb{F}$: a collection of functions parametrized by Θ

Predictions based on Probability

- Supervised learning, we are given a training set $\mathbb{X} = \{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}_{i=1}^{N}$
- Model $\mathbb{F}$: a collection of functions parametrized by Θ
- Goal: to train a function f such that, given a new data point $\boldsymbol{x}'$, the output value

$$\hat{\boldsymbol{y}} = f(\boldsymbol{x}'; \Theta)$$

is closest to the correct label $\boldsymbol{y}'$

Predictions based on Probability

- Supervised learning, we are given a training set $\mathbb{X} = \{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}_{i=1}^{N}$
- Model $\mathbb{F}$: a collection of functions parametrized by Θ
- Goal: to train a function f such that, given a new data point $\boldsymbol{x}'$, the output value
$$\hat{\boldsymbol{y}} = f(\boldsymbol{x}'; \Theta)$$
is closest to the correct label $\boldsymbol{y}'$
- Examples in $\mathbb{X}$ are usually assumed to be i.i.d. sampled from random variables $(\mathbf{x}, \mathbf{y})$ following some data generating distribution $\mathrm{P}(\mathbf{x}, \mathbf{y})$

Predictions based on Probability

- Supervised learning, we are given a training set $\mathbb{X} = \{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}_{i=1}^{N}$
- Model $\mathbb{F}$: a collection of functions parametrized by Θ
- Goal: to train a function f such that, given a new data point $\boldsymbol{x}'$, the output value

  $$\hat{\boldsymbol{y}} = f(\boldsymbol{x}'; \Theta)$$

  is closest to the correct label $\boldsymbol{y}'$
- Examples in $\mathbb{X}$ are usually assumed to be i.i.d. sampled from random variables $(\mathbf{x}, \mathbf{y})$ following some data generating distribution $\mathrm{P}(\mathbf{x}, \mathbf{y})$
- In probabilistic models, f is replaced by $\mathrm{P}(\mathbf{y} = \boldsymbol{y} \mid \mathbf{x} = \boldsymbol{x}')$ and a prediction is made by:

  $$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\mathbf{y} = \boldsymbol{y} \mid \mathbf{x} = \boldsymbol{x}'; \Theta)$$

Predictions based on Probability

- Supervised learning, we are given a training set $\mathbb{X} = \{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}_{i=1}^N$
- Model $\mathbb{F}$: a collection of functions parametrized by Θ
- Goal: to train a function f such that, given a new data point $\boldsymbol{x}'$, the output value

$$\hat{\boldsymbol{y}} = f(\boldsymbol{x}'; \Theta)$$

  is closest to the correct label $\boldsymbol{y}'$
- Examples in $\mathbb{X}$ are usually assumed to be i.i.d. sampled from random variables $(\mathbf{x}, \mathbf{y})$ following some data generating distribution $\mathrm{P}(\mathbf{x}, \mathbf{y})$
- In probabilistic models, f is replaced by $\mathrm{P}(\mathbf{y} = \boldsymbol{y} \mid \mathbf{x} = \boldsymbol{x}')$ and a prediction is made by:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\mathbf{y} = \boldsymbol{y} \mid \mathbf{x} = \boldsymbol{x}'; \Theta)$$

- How to find Θ?

Function (Θ) as Point Estimate

- Regard Θ (f) as an estimate of the "true" Θ^* (f^*)
  - Mapped from the training set $\mathbb{X}$

Function (Θ) as Point Estimate

- Regard Θ (f) as an estimate of the "true" Θ^* (f^*)
  - Mapped from the training set $\mathbb{X}$
- ***Maximum a posteriori (MAP) estimation***:

$$\arg\max_{\boldsymbol{\Theta}} \mathrm{P}(\boldsymbol{\Theta} \,|\, \mathbb{X}) = \arg\max_{\boldsymbol{\Theta}} \mathrm{P}(\mathbb{X} \,|\, \boldsymbol{\Theta})\mathrm{P}(\boldsymbol{\Theta})$$

  - By Bayes' rule ($\mathrm{P}(\mathbb{X})$ is irrelevant)

Function (Θ) as Point Estimate

- Regard Θ (f) as an estimate of the "true" Θ^* (f^*)
  - Mapped from the training set $\mathbb{X}$
- ***Maximum a posteriori (MAP) estimation***:

$$\arg\max_{\Theta} \mathrm{P}(\Theta \,|\, \mathbb{X}) = \arg\max_{\Theta} \mathrm{P}(\mathbb{X} \,|\, \Theta)\mathrm{P}(\Theta)$$

  - By Bayes' rule ($\mathrm{P}(\mathbb{X})$ is irrelevant)
  - Solves Θ first, then uses it as a constant in $\mathrm{P}(\boldsymbol{y} \,|\, \boldsymbol{x}; \Theta)$ to get $\hat{\boldsymbol{y}}$

Function (Θ) as Point Estimate

- Regard Θ (f) as an estimate of the "true" Θ^* (f^*)
  - Mapped from the training set $\mathbb{X}$
- *Maximum a posteriori (MAP) estimation*:

$$\arg\max_{\Theta} \mathrm{P}(\Theta \,|\, \mathbb{X}) = \arg\max_{\Theta} \mathrm{P}(\mathbb{X} \,|\, \Theta)\mathrm{P}(\Theta)$$

  - By Bayes' rule ($\mathrm{P}(\mathbb{X})$ is irrelevant)
  - Solves Θ first, then uses it as a constant in $\mathrm{P}(\boldsymbol{y} \,|\, \boldsymbol{x};\Theta)$ to get $\hat{\boldsymbol{y}}$
- *Maximum likelihood (ML) estimation*:

$$\arg\max_{\Theta} \mathrm{P}(\mathbb{X} \,|\, \Theta)$$

  - Assumes uniform $\mathrm{P}(\Theta)$ and does not prefer particular Θ

Outline

Outline

Probability Interpretation

- Assumption: $y = f^*(\boldsymbol{x}) + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \beta^{-1})$

Probability Interpretation

- Assumption: $y = f^*(\boldsymbol{x}) + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \beta^{-1})$
- The unknown deterministic function is defined as

$$f^*(\boldsymbol{x}; \boldsymbol{w}^*) = \boldsymbol{w}^{*\top}\boldsymbol{x}$$

  - All variables are z-normalized, so no bias term (b)

Probability Interpretation

- Assumption: $y = f^*(\boldsymbol{x}) + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \beta^{-1})$
- The unknown deterministic function is defined as

$$f^*(\boldsymbol{x}; \boldsymbol{w}^*) = \boldsymbol{w}^{*\top}\boldsymbol{x}$$

  - All variables are z-normalized, so no bias term (b)
- We have $(\mathrm{y} \,|\, \boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{w}^{*\top}\boldsymbol{x}, \beta^{-1})$

Probability Interpretation

- Assumption: $y = f^*(\boldsymbol{x}) + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \beta^{-1})$
- The unknown deterministic function is defined as

$$f^*(\boldsymbol{x}; \boldsymbol{w}^*) = \boldsymbol{w}^{*\top}\boldsymbol{x}$$

  - All variables are z-normalized, so no bias term (b)
- We have $(\mathrm{y} \,|\, \boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{w}^{*\top}\boldsymbol{x}, \beta^{-1})$
- So, out goal is to find $\boldsymbol{w}$ as close to $\boldsymbol{w}^*$ as possible such that:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y} \,|\, \mathbf{x} = \boldsymbol{x}; \boldsymbol{w}) = \boldsymbol{w}^\top \boldsymbol{x}$$

  - Note that $\hat{y}$ is irrelevant to β, so we don't need to solve β

Probability Interpretation

- Assumption: $y = f^*(\boldsymbol{x}) + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \beta^{-1})$
- The unknown deterministic function is defined as

$$f^*(\boldsymbol{x}; \boldsymbol{w}^*) = \boldsymbol{w}^{*\top}\boldsymbol{x}$$

  - All variables are z-normalized, so no bias term (b)
- We have $(\mathrm{y} \,|\, \boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{w}^{*\top}\boldsymbol{x}, \beta^{-1})$
- So, out goal is to find $\boldsymbol{w}$ as close to $\boldsymbol{w}^*$ as possible such that:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y} \,|\, \mathbf{x} = \boldsymbol{x}; \boldsymbol{w}) = \boldsymbol{w}^\top \boldsymbol{x}$$

  - Note that $\hat{y}$ is irrelevant to β, so we don't need to solve β
- ML estimation:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X} \,|\, \boldsymbol{w})$$

ML Estimation I

- Problem:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X} \,|\, \boldsymbol{w})$$

- Since we assume i.i.d. samples, we have

$$\mathrm{P}(\mathbb{X} \,|\, \boldsymbol{w}) = \prod_{i=1}^{N} \mathrm{P}(\boldsymbol{x}^{(i)}, y^{(i)} \,|\, \boldsymbol{w}) = \prod_{i=1}^{N} \mathrm{P}(y^{(i)} \,|\, \boldsymbol{x}^{(i)}, \boldsymbol{w}) \mathrm{P}(\boldsymbol{x}^{(i)} \,|\, \boldsymbol{w})$$

ML Estimation I

- Problem:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})$$

- Since we assume i.i.d. samples, we have

$$\begin{aligned}\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) &= \prod_{i=1}^{N} \mathrm{P}(\boldsymbol{x}^{(i)}, y^{(i)}\,|\,\boldsymbol{w}) = \prod_{i=1}^{N} \mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}\,|\,\boldsymbol{w}) \\ &= \prod_{i=1}^{N} \mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}) = \prod_{i} \mathcal{N}(y^{(i)}; \boldsymbol{w}^\top \boldsymbol{x}^{(i)}, \sigma^2)\mathrm{P}(\boldsymbol{x}^{(i)})\end{aligned}$$

ML Estimation I

- Problem:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})$$

- Since we assume i.i.d. samples, we have

$$\begin{aligned}\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) &= \prod_{i=1}^{N}\mathrm{P}(\boldsymbol{x}^{(i)},y^{(i)}\,|\,\boldsymbol{w}) = \prod_{i=1}^{N}\mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)},\boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}\,|\,\boldsymbol{w})\\ &= \prod_{i=1}^{N}\mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)},\boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}) = \prod_{i}\mathcal{N}(y^{(i)};\boldsymbol{w}^\top\boldsymbol{x}^{(i)},\sigma^2)\mathrm{P}(\boldsymbol{x}^{(i)})\\ &= \prod_{i}\sqrt{\frac{\beta}{2\pi}}\exp\left(-\frac{\beta}{2}(y^{(i)}-\boldsymbol{w}^\top\boldsymbol{x}^{(i)})^2\right)\mathrm{P}(\boldsymbol{x}^{(i)})\end{aligned}$$

ML Estimation I

- Problem:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})$$

- Since we assume i.i.d. samples, we have

$$\begin{aligned}
\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) &= \textstyle\prod_{i=1}^{N} \mathrm{P}(\boldsymbol{x}^{(i)}, y^{(i)}\,|\,\boldsymbol{w}) = \prod_{i=1}^{N} \mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}\,|\,\boldsymbol{w}) \\
&= \textstyle\prod_{i=1}^{N} \mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}) = \prod_i \mathcal{N}(y^{(i)}; \boldsymbol{w}^\top \boldsymbol{x}^{(i)}, \sigma^2)\mathrm{P}(\boldsymbol{x}^{(i)}) \\
&= \textstyle\prod_i \sqrt{\frac{\beta}{2\pi}} \exp\left(-\frac{\beta}{2}(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2\right) \mathrm{P}(\boldsymbol{x}^{(i)})
\end{aligned}$$

- To make the problem tractable, we prefer "sums" over "products"

ML Estimation I

- Problem:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})$$

- Since we assume i.i.d. samples, we have

$$\begin{aligned}\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) &= \textstyle\prod_{i=1}^{N} \mathrm{P}(\boldsymbol{x}^{(i)}, y^{(i)}\,|\,\boldsymbol{w}) = \prod_{i=1}^{N} \mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}\,|\,\boldsymbol{w}) \\ &= \textstyle\prod_{i=1}^{N} \mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}) = \prod_i \mathcal{N}(y^{(i)}; \boldsymbol{w}^\top \boldsymbol{x}^{(i)}, \sigma^2)\mathrm{P}(\boldsymbol{x}^{(i)}) \\ &= \textstyle\prod_i \sqrt{\frac{\beta}{2\pi}} \exp\left(-\frac{\beta}{2}(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2\right) \mathrm{P}(\boldsymbol{x}^{(i)})\end{aligned}$$

- To make the problem tractable, we prefer "sums" over "products"
- We can instead maximize the ***log likelihood***

$$\arg\max_{\boldsymbol{w}} \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})$$

- The optimal point does not change since $\log$ is monotone increasing

ML Estimation I

- Problem:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})$$

- Since we assume i.i.d. samples, we have

$$\begin{aligned}\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) = \prod_{i=1}^{N} \mathrm{P}(\boldsymbol{x}^{(i)}, y^{(i)}\,|\,\boldsymbol{w}) &= \prod_{i=1}^{N} \mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}\,|\,\boldsymbol{w}) \\ &= \prod_{i=1}^{N} \mathrm{P}(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w})\mathrm{P}(\boldsymbol{x}^{(i)}) = \prod_i \mathcal{N}(y^{(i)}; \boldsymbol{w}^\top \boldsymbol{x}^{(i)}, \sigma^2)\mathrm{P}(\boldsymbol{x}^{(i)}) \\ &= \prod_i \sqrt{\frac{\beta}{2\pi}} \exp\left(-\frac{\beta}{2}(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2\right) \mathrm{P}(\boldsymbol{x}^{(i)})\end{aligned}$$

- To make the problem tractable, we prefer "sums" over "products"
- We can instead maximize the ***log likelihood***

$$\begin{aligned}&\arg\max_{\boldsymbol{w}} \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) \\ &\quad = \arg\max_{\boldsymbol{w}} \log\left[\prod_i \sqrt{\frac{\beta}{2\pi}} \exp\left(-\frac{\beta}{2}(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2\right) \mathrm{P}(\boldsymbol{x}^{(i)})\right]\end{aligned}$$

- The optimal point does not change since $\log$ is monotone increasing

ML Estimation I

- Problem:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X} \,|\, \boldsymbol{w})$$

- Since we assume i.i.d. samples, we have

$$\begin{aligned}
\mathrm{P}(\mathbb{X} \,|\, \boldsymbol{w}) &= \prod_{i=1}^{N} \mathrm{P}(\boldsymbol{x}^{(i)}, y^{(i)} \,|\, \boldsymbol{w}) = \prod_{i=1}^{N} \mathrm{P}(y^{(i)} \,|\, \boldsymbol{x}^{(i)}, \boldsymbol{w}) \mathrm{P}(\boldsymbol{x}^{(i)} \,|\, \boldsymbol{w}) \\
&= \prod_{i=1}^{N} \mathrm{P}(y^{(i)} \,|\, \boldsymbol{x}^{(i)}, \boldsymbol{w}) \mathrm{P}(\boldsymbol{x}^{(i)}) = \prod_i \mathcal{N}(y^{(i)}; \boldsymbol{w}^\top \boldsymbol{x}^{(i)}, \sigma^2) \mathrm{P}(\boldsymbol{x}^{(i)}) \\
&= \prod_i \sqrt{\frac{\beta}{2\pi}} \exp\left(-\frac{\beta}{2}(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2\right) \mathrm{P}(\boldsymbol{x}^{(i)})
\end{aligned}$$

- To make the problem tractable, we prefer "sums" over "products"
- We can instead maximize the ***log likelihood***

$$\begin{aligned}
&\arg\max_{\boldsymbol{w}} \log \mathrm{P}(\mathbb{X} \,|\, \boldsymbol{w}) \\
&\quad = \arg\max_{\boldsymbol{w}} \log \left[\prod_i \sqrt{\frac{\beta}{2\pi}} \exp\left(-\frac{\beta}{2}(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2\right) \mathrm{P}(\boldsymbol{x}^{(i)}) \right] \\
&\quad = \arg\max_{\boldsymbol{w}} N \sqrt{\frac{\beta}{2\pi}} - \frac{\beta}{2} \sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2 + \sum_i \mathrm{P}(\boldsymbol{x}^{(i)})
\end{aligned}$$

- The optimal point does not change since $\log$ is monotone increasing

ML Estimation II

$$\arg\max_{\boldsymbol{w}} \underbrace{N\sqrt{\frac{\beta}{2\pi}}}_{\text{W無關}} \underbrace{-\frac{\beta}{2}\sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2}_{\text{唯一有關W的}} + \underbrace{\sum_i \mathrm{P}(\boldsymbol{x}^{(i)})}_{\text{W無關}}$$

- Ignoring terms irrelevant to $\boldsymbol{w}$, we have

$$\arg\min_{\boldsymbol{w}} \sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2$$

ML Estimation II

$$\arg\max_{\boldsymbol{w}} N\sqrt{\frac{\beta}{2\pi}} - \frac{\beta}{2}\sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2 + \sum_i \mathrm{P}(\boldsymbol{x}^{(i)})$$

- Ignoring terms irrelevant to $\boldsymbol{w}$, we have

$$\arg\min_{\boldsymbol{w}} \sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2$$

- In other words, we seek for $\boldsymbol{w}$ by ***minimizing the SSE*** (sum of square errors), as we have done before
  - By, e.g., the stochastic gradient descent algorithm

ML Estimation II

$$\arg\max_{\boldsymbol{w}} N\sqrt{\frac{\beta}{2\pi}} - \frac{\beta}{2}\sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2 + \sum_i \mathrm{P}(\boldsymbol{x}^{(i)})$$

- Ignoring terms irrelevant to $\boldsymbol{w}$, we have

$$\arg\min_{\boldsymbol{w}} \sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2$$

- In other words, we seek for $\boldsymbol{w}$ by ***minimizing the SSE*** (sum of square errors), as we have done before
    - By, e.g., the stochastic gradient descent algorithm
- This new perspective explains our ad hoc choice of SSE for empirical risk minimization
    - Checking assumptions helps understand when model works the best

ML Estimation II

$$\arg\max_{\boldsymbol{w}} N\sqrt{\frac{\beta}{2\pi}} - \frac{\beta}{2}\sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2 + \sum_i \mathrm{P}(\boldsymbol{x}^{(i)})$$

- Ignoring terms irrelevant to $\boldsymbol{w}$, we have

$$\arg\min_{\boldsymbol{w}} \sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2$$

- In other words, we seek for $\boldsymbol{w}$ by ***minimizing the SSE*** (sum of square errors), as we have done before
  - By, e.g., the stochastic gradient descent algorithm
- This new perspective explains our ad hoc choice of SSE for empirical risk minimization
  - Checking assumptions helps understand when model works the best
- Also motivates new models

ML Estimation II

$$\arg\max_{\boldsymbol{w}} N\sqrt{\frac{\beta}{2\pi}} - \frac{\beta}{2}\sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2 + \sum_i \mathrm{P}(\boldsymbol{x}^{(i)})$$

- Ignoring terms irrelevant to $\boldsymbol{w}$, we have

$$\arg\min_{\boldsymbol{w}} \sum_i (y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)})^2$$

- In other words, we seek for $\boldsymbol{w}$ by ***minimizing the SSE*** (sum of square errors), as we have done before
    - By, e.g., the stochastic gradient descent algorithm
- This new perspective explains our ad hoc choice of SSE for empirical risk minimization
    - Checking assumptions helps understand when model works the best
- Also motivates new models. Probabilistic model for classification?

Outline

Probabilistic Models for Binary Classification

- Probabilistic models:

$$\hat{y} = \arg\max_{y} \mathrm{P}(y \,|\, \boldsymbol{x}; \Theta)$$

Probabilistic Models for Binary Classification

- Probabilistic models:

$$\hat{y} = \arg\max_{y} \mathrm{P}(y \mid \boldsymbol{x}; \Theta)$$

- In regression, we assume $(\mathrm{y} \mid \boldsymbol{x}) \sim \mathcal{N}$ (based on $y = f^*(\boldsymbol{x}) + \varepsilon$)

Probabilistic Models for Binary Classification

- Probabilistic models:

$$\hat{y} = \arg\max_{y} \mathrm{P}(y \,|\, \boldsymbol{x}; \Theta)$$

- In regression, we assume $(\mathrm{y} \,|\, \boldsymbol{x}) \sim \mathcal{N}$ (based on $y = f^*(\boldsymbol{x}) + \varepsilon$)
- However, Gaussian distribution is ***not*** applicable to binary classification
  - The values of y should concentrate in either 1 or -1

Probabilistic Models for Binary Classification

- Probabilistic models:

$$\hat{y} = \arg\max_{y} \mathrm{P}(y \,|\, \boldsymbol{x}; \Theta)$$

- In regression, we assume $(\mathrm{y} \,|\, \boldsymbol{x}) \sim \mathcal{N}$ (based on $y = f^*(\boldsymbol{x}) + \varepsilon$)
- However, Gaussian distribution is ***not*** applicable to binary classification
  - The values of y should concentrate in either 1 or -1
- Which distribution to assume?

Probabilistic Models for Binary Classification

- Probabilistic models:

$$\hat{y} = \arg\max_{y} \mathrm{P}(y \,|\, \boldsymbol{x}; \Theta)$$

- In regression, we assume $(\mathrm{y} \,|\, \boldsymbol{x}) \sim \mathcal{N}$ (based on $y = f^*(\boldsymbol{x}) + \varepsilon$)
- However, Gaussian distribution is ***not*** applicable to binary classification
  - The values of y should concentrate in either 1 or -1
- Which distribution to assume?
- Coin flipping: $(\mathrm{y} \,|\, \boldsymbol{x}) \sim \mathrm{Bernoulli}(\rho)$, where

$$\mathrm{P}(y \,|\, \boldsymbol{x}; \rho) = \rho^{y'} (1-\rho)^{(1-y')}, \text{ where } y' = \frac{y+1}{2}$$

Probabilistic Models for Binary Classification

- Probabilistic models:

$$\hat{y} = \arg\max_{y} \mathrm{P}(y \,|\, \boldsymbol{x}; \Theta)$$

- In regression, we assume $(\mathrm{y} \,|\, \boldsymbol{x}) \sim \mathcal{N}$ (based on $y = f^*(\boldsymbol{x}) + \varepsilon$)
- However, Gaussian distribution is ***not*** applicable to binary classification
  - The values of y should concentrate in either 1 or -1
- Which distribution to assume?
- Coin flipping: $(\mathrm{y} \,|\, \boldsymbol{x}) \sim \mathrm{Bernoulli}(\rho)$, where

$$\mathrm{P}(y \,|\, \boldsymbol{x}; \rho) = \rho^{y'} (1-\rho)^{(1-y')}, \text{ where } y' = \frac{y+1}{2}$$

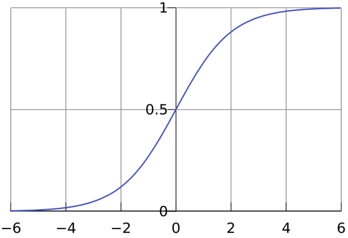
- How to relate $\boldsymbol{x}$ to ρ?

Logistic Function

- Recall that the ***logistic function***

$$\sigma(z) = \frac{\exp(z)}{\exp(z)+1} = \frac{1}{1+\exp(-z)}$$

is commonly used as a parametrizing function of the Bernoulli distribution



Logistic Function

- Recall that the *logistic function*

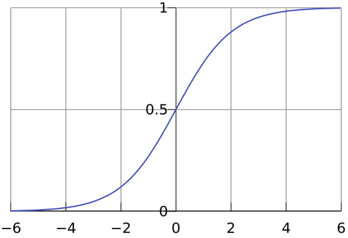
$$\sigma(z) = \frac{\exp(z)}{\exp(z)+1} = \frac{1}{1+\exp(-z)}$$

  is commonly used as a parametrizing function of the Bernoulli distribution

- We have

$$\mathrm{P}(y \mid \boldsymbol{x}; z) = \sigma(z)^{y'}\,(1-\sigma(z))^{(1-y')}$$

- The larger z, the higher chance we get a "positive flip"



Logistic Function

- Recall that the *logistic function*

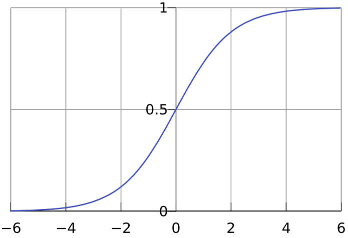
$$\sigma(z) = \frac{\exp(z)}{\exp(z)+1} = \frac{1}{1+\exp(-z)}$$

  is commonly used as a parametrizing function of the Bernoulli distribution

- We have

$$\mathrm{P}(y\,|\,\boldsymbol{x};z) = \sigma(z)^{y'}\,(1-\sigma(z))^{(1-y')}$$

- The larger z, the higher chance we get a "positive flip"
- How to relate $\boldsymbol{x}$ to z?



Logistic Regression

- In ***logistic regression***, we let

$$z = \boldsymbol{w}^\top \boldsymbol{x}$$

  - Basically, z is the projection of $\boldsymbol{x}$ along the direction $\boldsymbol{w}$

Logistic Regression

- In *logistic regression*, we let

$$\boxed{z = \boldsymbol{w}^\top \boldsymbol{x}}$$ x多維 map到 z一維

  - Basically, z is the projection of $\boldsymbol{x}$ along the direction $\boldsymbol{w}$
- We have

$$\mathrm{P}(y\,|\,\boldsymbol{x};\boldsymbol{w}) = \sigma(\boldsymbol{w}^\top \boldsymbol{x})^{y'}[1-\sigma(\boldsymbol{w}^\top \boldsymbol{x})]^{(1-y')}$$

- Prediction:

$$\hat{y} = \arg\max_y \mathrm{P}(y\,|\,\boldsymbol{x};\boldsymbol{w})$$

Logistic Regression

- In ***logistic regression***, we let

$$z = \boldsymbol{w}^\top \boldsymbol{x}$$

  - Basically, z is the projection of $\boldsymbol{x}$ along the direction $\boldsymbol{w}$
- We have

$$\mathrm{P}(y \mid \boldsymbol{x}; \boldsymbol{w}) = \sigma(\boldsymbol{w}^\top \boldsymbol{x})^{y'} [1 - \sigma(\boldsymbol{w}^\top \boldsymbol{x})]^{(1-y')}$$

- Prediction:

$$\hat{y} = \arg\max_{y} \mathrm{P}(y \mid \boldsymbol{x}; \boldsymbol{w}) = \mathrm{sign}(\boldsymbol{w}^\top \boldsymbol{x})$$

Logistic Regression

- In ***logistic regression***, we let

$$z = \boldsymbol{w}^\top \boldsymbol{x}$$

  - Basically, z is the projection of $\boldsymbol{x}$ along the direction $\boldsymbol{w}$

- We have

$$\mathrm{P}(y \,|\, \boldsymbol{x}; \boldsymbol{w}) = \underset{(A)}{\boxed{\sigma(\boldsymbol{w}^\top \boldsymbol{x})}}^{y'} \underset{(B)}{\boxed{[1 - \sigma(\boldsymbol{w}^\top \boldsymbol{x})]}}^{(1-y')}$$

- Prediction:

$$\hat{y} = \arg\max_y \mathrm{P}(y \,|\, \boldsymbol{x}; \boldsymbol{w}) = \mathrm{sign}(\boldsymbol{w}^\top \boldsymbol{x})$$

$= \arg\max_{y} \{(A), (B)\}$

- How to learn $\boldsymbol{w}$ from $\mathbb{X}$?

Logistic Regression

- In *logistic regression*, we let

$$z = \boldsymbol{w}^\top \boldsymbol{x}$$

  - Basically, z is the projection of $\boldsymbol{x}$ along the direction $\boldsymbol{w}$
- We have

$$\mathrm{P}(y \mid \boldsymbol{x}; \boldsymbol{w}) = \sigma(\boldsymbol{w}^\top \boldsymbol{x})^{y'} [1 - \sigma(\boldsymbol{w}^\top \boldsymbol{x})]^{(1-y')}$$

- Prediction:

$$\hat{y} = \arg\max_{y} \mathrm{P}(y \mid \boldsymbol{x}; \boldsymbol{w}) = \mathrm{sign}(\boldsymbol{w}^\top \boldsymbol{x})$$

- How to learn $\boldsymbol{w}$ from $\mathbb{X}$?
- ML estimation:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X} \mid \boldsymbol{w})$$

ML Estimation

- Log-likelihood:

$$
\begin{aligned}
\log \mathrm{P}(\mathbb{X} \,|\, \boldsymbol{w}) &= \log \prod_{i=1}^{N} \mathrm{P}\left(\boldsymbol{x}^{(i)}, y^{(i)} \,|\, \boldsymbol{w}\right) \\
&= \log \prod_{i} \mathrm{P}\left(y^{(i)} \,|\, \boldsymbol{x}^{(i)}, \boldsymbol{w}\right) \mathrm{P}\left(\boldsymbol{x}^{(i)} \,|\, \boldsymbol{w}\right)
\end{aligned}
$$

ML Estimation

- Log-likelihood:

$$\begin{aligned}\log \mathrm{P}(\mathbb{X} \,|\, \boldsymbol{w}) &= \log \prod_{i=1}^{N} \mathrm{P}\left(\boldsymbol{x}^{(i)}, y^{(i)} \,|\, \boldsymbol{w}\right) \\ &= \log \prod_{i} \mathrm{P}\left(y^{(i)} \,|\, \boldsymbol{x}^{(i)}, \boldsymbol{w}\right) \mathrm{P}\left(\boldsymbol{x}^{(i)} \,|\, \boldsymbol{w}\right) \\ &\propto \log \prod_{i} \sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})^{y'^{(i)}} [1 - \sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})]^{(1-y'^{(i)})}\end{aligned}$$

ML Estimation

- Log-likelihood:

$$\begin{aligned}\log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) &= \log \textstyle\prod_{i=1}^{N} \mathrm{P}\left(\boldsymbol{x}^{(i)}, y^{(i)}\,|\,\boldsymbol{w}\right)\\ &= \log \textstyle\prod_{i} \mathrm{P}\left(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w}\right) \mathrm{P}\left(\boldsymbol{x}^{(i)}\,|\,\boldsymbol{w}\right)\\ &\propto \log \textstyle\prod_{i} \sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})^{y'^{(i)}} [1-\sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})]^{(1-y'^{(i)})}\\ &= \textstyle\sum_{i} y'^{(i)} \boldsymbol{w}^\top \boldsymbol{x}^{(i)} - \log(1+e^{\boldsymbol{w}^\top \boldsymbol{x}^{(i)}}) \text{ [Homework]}\end{aligned}$$

- Unlike in linear regression, we cannot solve $\boldsymbol{w}$ analytically in a closed form via

$$\nabla_{\boldsymbol{w}} \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) = \sum_{t=1}^{N} [y'^{(i)} - \sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})]\boldsymbol{x}^{(i)} = \mathbf{0}$$

ML Estimation

- Log-likelihood:

$$\begin{aligned}\log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) &= \log \textstyle\prod_{i=1}^{N} \mathrm{P}\left(\boldsymbol{x}^{(i)}, y^{(i)}\,|\,\boldsymbol{w}\right) \\ &= \log \textstyle\prod_i \mathrm{P}\left(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w}\right) \mathrm{P}\left(\boldsymbol{x}^{(i)}\,|\,\boldsymbol{w}\right) \\ &\propto \log \textstyle\prod_i \sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})^{y'^{(i)}} [1-\sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})]^{(1-y'^{(i)})} \\ &= \underline{\textstyle\sum_i y'^{(i)} \boldsymbol{w}^\top \boldsymbol{x}^{(i)} - \log(1+e^{\boldsymbol{w}^\top \boldsymbol{x}^{(i)}})} \text{ [Homework]}\end{aligned}$$

- Unlike in linear regression, we cannot solve $\boldsymbol{w}$ analytically in a closed form via 做偏微 取gradient

$$\nabla_{\boldsymbol{w}} \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) = \sum_{t=1}^{N} [y'^{(i)} - \sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})] \boldsymbol{x}^{(i)} = \boldsymbol{0}$$

不好解

- However, we can still evaluate $\nabla_{\boldsymbol{w}} \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})$ and use the iterative methods to solve $\boldsymbol{w}$
  - E.g., stochastic gradient descent

ML Estimation

- Log-likelihood:

$$\begin{aligned}\log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) &= \log \textstyle\prod_{i=1}^{N} \mathrm{P}\left(\boldsymbol{x}^{(i)}, y^{(i)}\,|\,\boldsymbol{w}\right)\\ &= \log \textstyle\prod_i \mathrm{P}\left(y^{(i)}\,|\,\boldsymbol{x}^{(i)}, \boldsymbol{w}\right) \mathrm{P}\left(\boldsymbol{x}^{(i)}\,|\,\boldsymbol{w}\right)\\ &\propto \log \textstyle\prod_i \sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})^{y'^{(i)}} [1-\sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})]^{(1-y'^{(i)})}\\ &= \textstyle\sum_i y'^{(i)} \boldsymbol{w}^\top \boldsymbol{x}^{(i)} - \log(1+e^{\boldsymbol{w}^\top \boldsymbol{x}^{(i)}}) \text{ [Homework]}\end{aligned}$$

- Unlike in linear regression, we cannot solve $\boldsymbol{w}$ analytically in a closed form via

$$\nabla_{\boldsymbol{w}} \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) = \sum_{t=1}^{N} [y'^{(i)} - \sigma(\boldsymbol{w}^\top \boldsymbol{x}^{(i)})] \boldsymbol{x}^{(i)} = \boldsymbol{0}$$

- However, we can still evaluate $\nabla_{\boldsymbol{w}} \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})$ and use the iterative methods to solve $\boldsymbol{w}$
  - E.g., stochastic gradient descent
- It can be shown that $\log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})$ is concave in terms of $\boldsymbol{w}$ [1]
  - So, iterative algorithms converges

Outline

MAP Estimation

- So far, we solve $\boldsymbol{w}$ by ML estimation:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X} \,|\, \boldsymbol{w})$$

MAP Estimation

- So far, we solve $\boldsymbol{w}$ by ML estimation:

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X} \mid \boldsymbol{w})$$ 沒有preference W是uniform的

- In MAP estimation, we solve 較符合直覺 根據一組X 去推測最好的W

$$\arg\max_{\boldsymbol{w}} \mathrm{P}(\boldsymbol{w} \mid \mathbb{X}) = \arg\max_{\boldsymbol{w}} \mathrm{P}(\mathbb{X} \mid \boldsymbol{w}) \mathrm{P}(\boldsymbol{w})$$

- $\mathrm{P}(\boldsymbol{w})$ models our ***preference*** or ***prior knowledge*** about $\boldsymbol{w}$

MAP Estimation for Linear Regression

- MAP estimation in linear regression:

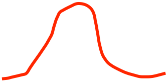
$$\arg\max_{\boldsymbol{w}} \log[\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})\mathrm{P}(\boldsymbol{w})]$$

MAP Estimation for Linear Regression

- MAP estimation in linear regression:

$$\arg\max_{\boldsymbol{w}} \log[\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})\mathrm{P}(\boldsymbol{w})]$$

- If we assume that $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \beta^{-1}\boldsymbol{I})$ 假設W在0的機率最高



0

$$\log[\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})\mathrm{P}(\boldsymbol{w})] = \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) + \log \mathrm{P}(\boldsymbol{w})$$

MAP Estimation for Linear Regression

- MAP estimation in linear regression:

$$\arg\max_{\boldsymbol{w}} \log[\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})\mathrm{P}(\boldsymbol{w})]$$

- If we assume that $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \beta^{-1}\boldsymbol{I})$

$$\begin{aligned}\log[\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})\mathrm{P}(\boldsymbol{w})] = \log\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) + \log\mathrm{P}(\boldsymbol{w}) &\propto -\sum_i \left(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)}\right)^2 \\ &+ \log\sqrt{\tfrac{1}{(2\pi)^D \det(\beta^{-1}\boldsymbol{I})}} \exp\left[-\tfrac{1}{2}(\boldsymbol{w}-\boldsymbol{0})^\top(\beta^{-1}\boldsymbol{I})^{-1}(\boldsymbol{w}-\boldsymbol{0})\right]\end{aligned}$$

MAP Estimation for Linear Regression

- MAP estimation in linear regression:

$$\arg\max_{\boldsymbol{w}} \log[\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})\mathrm{P}(\boldsymbol{w})]$$

- If we assume that $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \beta^{-1}\boldsymbol{I})$

$$\begin{aligned}\log[\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})\mathrm{P}(\boldsymbol{w})] = \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) + \log \mathrm{P}(\boldsymbol{w}) \propto -\textstyle\sum_i \left(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)}\right)^2 \\ + \log \sqrt{\tfrac{1}{(2\pi)^D \det(\beta^{-1}\boldsymbol{I})}} \exp\left[-\tfrac{1}{2}(\boldsymbol{w}-\boldsymbol{0})^\top (\beta^{-1}\boldsymbol{I})^{-1}(\boldsymbol{w}-\boldsymbol{0})\right] \\ \propto -\textstyle\sum_i \left(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)}\right)^2 - \beta \boldsymbol{w}^\top \boldsymbol{w}\end{aligned}$$

- $\mathrm{P}(\boldsymbol{w})$ corresponds to the ***weight decay*** term in Ridge regression

MAP Estimation for Linear Regression

- MAP estimation in linear regression:

$$\arg\max_{\boldsymbol{w}} \log[\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})\mathrm{P}(\boldsymbol{w})]$$

- If we assume that $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \beta^{-1}\boldsymbol{I})$

$$\begin{aligned}\log[\mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w})\mathrm{P}(\boldsymbol{w})] = \log \mathrm{P}(\mathbb{X}\,|\,\boldsymbol{w}) + \log \mathrm{P}(\boldsymbol{w}) &\propto -\textstyle\sum_i \left(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)}\right)^2 \\ &+ \log \sqrt{\tfrac{1}{(2\pi)^D \det(\beta^{-1}\boldsymbol{I})}} \exp\left[-\tfrac{1}{2}(\boldsymbol{w}-\boldsymbol{0})^\top (\beta^{-1}\boldsymbol{I})^{-1}(\boldsymbol{w}-\boldsymbol{0})\right] \\ \propto -\textstyle\sum_i \left(y^{(i)} - \boldsymbol{w}^\top \boldsymbol{x}^{(i)}\right)^2 - \beta \boldsymbol{w}^\top \boldsymbol{w}\end{aligned}$$

- $\mathrm{P}(\boldsymbol{w})$ corresponds to the ***weight decay*** term in Ridge regression
- MAP estimation provides a way to design complicated yet interpretable regularization terms
  - E.g., we have LASSO by letting $\mathrm{P}(\boldsymbol{w}) \sim \mathrm{Laplace}(0, b)$ [Proof]
  - We can also let $\mathrm{P}(\boldsymbol{w})$ be a mixture of Gaussians

Remarks on ML and MAP Estimation

Theorem (Consistency)

*The ML estimator Θ_{ML} is **consistent**, i.e., $\lim_{N\to\infty}\Theta_{ML} \xrightarrow{\Pr} \Theta^*$ as long as the "true" $\mathrm{P}(y\,|\,\boldsymbol{x};\Theta^*)$ lies within our model $\mathbb{F}$.*

Remarks on ML and MAP Estimation

Theorem (Consistency)

*The ML estimator Θ_{ML} is **consistent**, i.e., $\lim_{N\to\infty}\Theta_{ML} \xrightarrow{\mathrm{Pr}} \Theta^*$ as long as the "true" $\mathrm{P}(y\,|\,\boldsymbol{x};\Theta^*)$ lies within our model $\mathbb{F}$.*

Theorem (Cramér-Rao Lower Bound [2])

At a fixed (large) number N of examples, no consistent estimator of Θ^* has a lower expected MSE (mean square error) than the ML estimator Θ_{ML}.

- That is, Θ_{ML} has a low sample complexity (or is statistic efficient)

Remarks on ML and MAP Estimation

Theorem (Consistency)

The ML estimator Θ_{ML} *is* **consistent**, *i.e.,* $\lim_{N\to\infty}\Theta_{ML} \xrightarrow{\mathrm{Pr}} \Theta^*$ *as long as the "true"* $\mathrm{P}(y\,|\,\boldsymbol{x};\Theta^*)$ *lies within our model* $\mathbb{F}$.

Theorem (Cramér-Rao Lower Bound [2])

At a fixed (large) number N *of examples, no consistent estimator of* Θ^* *has a lower expected MSE (mean square error) than the ML estimator* Θ_{ML}.

- That is, Θ_{ML} has a low sample complexity (or is statistic efficient)
- ML estimation is popular due to its consistency and efficiency

有著夠大的data set ＭＬ可以learn的最好

Remarks on ML and MAP Estimation

Theorem (Consistency)

The ML estimator Θ_{ML} is ***consistent****, i.e., $\lim_{N\rightarrow\infty}\Theta_{ML}\xrightarrow{\mathrm{Pr}}\Theta^*$ as long as the "true" $\mathrm{P}(y\,|\,\boldsymbol{x};\Theta^*)$ lies within our model $\mathbb{F}$.*

Theorem (Cramér-Rao Lower Bound [2])

At a fixed (large) number N of examples, no consistent estimator of Θ^* has a lower expected MSE (mean square error) than the ML estimator Θ_{ML}.

- That is, Θ_{ML} has a low sample complexity (or is statistic efficient)
- ML estimation is popular due to its consistency and efficiency 適合夠大的data set
- When N is small that yields overfitting behavior, we can use MAP estimation to ***introduce bias*** and ***reduce variance***

data set太小的話

Remarks on ML and MAP Estimation

Theorem (Consistency)

*The ML estimator Θ_{ML} is **consistent**, i.e., $\lim_{N\to\infty} \Theta_{ML} \xrightarrow{\text{Pr}} \Theta^*$ as long as the "true" $\mathrm{P}(y\,|\,\boldsymbol{x};\Theta^*)$ lies within our model $\mathbb{F}$.*

Theorem (Cramér-Rao Lower Bound [2])

At a fixed (large) number N of examples, no consistent estimator of Θ^* has a lower expected MSE (mean square error) than the ML estimator Θ_{ML}.

- That is, Θ_{ML} has a low sample complexity (or is statistic efficient)
- ML estimation is popular due to its consistency and efficiency
- When N is small that yields overfitting behavior, we can use MAP estimation to ***introduce bias*** and ***reduce variance***

Outline

Bayesian Estimation

- In ML/MAP estimation, we solve Θ first, then uses it as a constant to make prediction:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y} \,|\, \boldsymbol{x}; \Theta)$$

Bayesian Estimation

- In ML/MAP estimation, we solve Θ first, then uses it as a constant to make prediction:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y} \,|\, \boldsymbol{x}; \Theta)$$

- ***Bayesian estimation*** threats Θ as a random variable:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y} \,|\, \boldsymbol{x}, \mathbb{X}) = \arg\max_{\boldsymbol{y}} \int \mathrm{P}(\boldsymbol{y}, \Theta \,|\, \boldsymbol{x}, \mathbb{X}) d\Theta$$

  - Makes prediction by considering ***all*** Θ's (weighted by their chances)

Bayesian Estimation

- In ML/MAP estimation, we solve Θ first, then uses it as a constant to make prediction:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y} \,|\, \boldsymbol{x}; \Theta)$$

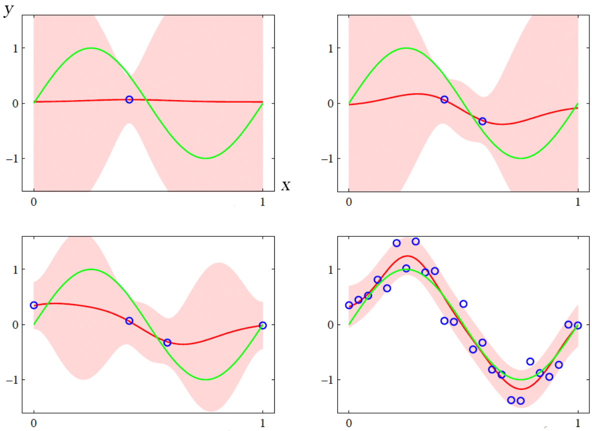
- ***Bayesian estimation*** threats Θ as a random variable:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y} \,|\, \boldsymbol{x}, \mathbb{X}) = \arg\max_{\boldsymbol{y}} \int \mathrm{P}(\boldsymbol{y}, \Theta \,|\, \boldsymbol{x}, \mathbb{X}) d\Theta$$

  - Makes prediction by considering ***all*** Θ's (weighted by their chances)
- Bayesian estimation usually generalizes much better when the size N of training set is small

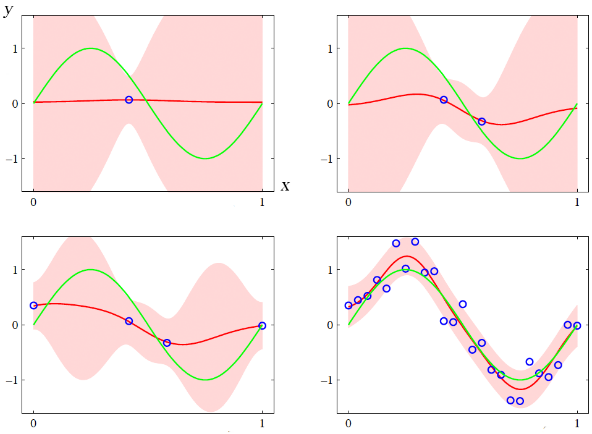
Bayesian vs. ML Estimation

- Example: polynomial regression



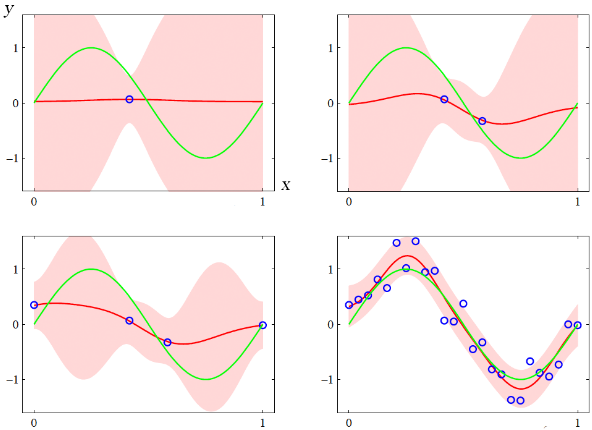
Bayesian vs. ML Estimation

- Example: polynomial regression
- Red line: predictions by Bayesian estimation regressor



Bayesian vs. ML Estimation

- Example: polynomial regression
- Red line: predictions by Bayesian estimation regressor
- Shaded area: predictions by ML/MAP estimation regressors

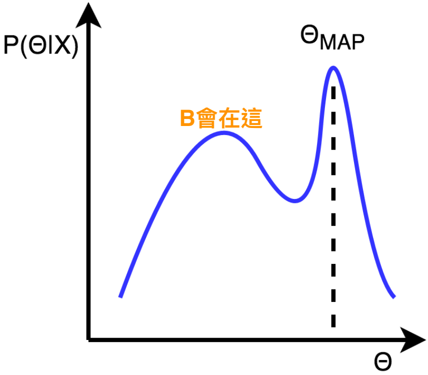


Bayesian vs. MAP Estimation

- MAP gains some benefit of Bayesian approach by incorporating prior as $\mathrm{bias}(\Theta_{\mathsf{MAP}})$
  - Reduces $\mathrm{Var}_{\mathbb{X}}(\Theta_{\mathsf{MAP}})$ when training set is small

Bayesian vs. MAP Estimation

- MAP gains some benefit of Bayesian approach by incorporating prior as $\mathrm{bias}(\Theta_{\mathrm{MAP}})$
  - Reduces $\mathrm{Var}_{\mathbb{X}}(\Theta_{\mathrm{MAP}})$ when training set is small
- However, does ***not*** work if Θ_{MAP} is unrepresentative of the majority Θ in $\int \mathrm{P}(\boldsymbol{y}, \Theta \,|\, \boldsymbol{x}, \mathbb{X}) d\Theta$
- E.g. when $\mathrm{P}(\Theta \,|\, \mathbb{X})$ is a mixture of Gaussian



Remarks

- Bayesian estimation:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y} \,|\, \boldsymbol{x}, \mathbb{X}) = \arg\max_{\boldsymbol{y}} \int \mathrm{P}(\boldsymbol{y}, \Theta \,|\, \boldsymbol{x}, \mathbb{X}) d\Theta$$

- Usually generalizes much better given a small training set

Remarks

- Bayesian estimation:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y} \,|\, \boldsymbol{x}, \mathbb{X}) = \arg\max_{\boldsymbol{y}} \int \mathrm{P}(\boldsymbol{y}, \Theta \,|\, \boldsymbol{x}, \mathbb{X}) d\Theta$$

- Usually generalizes much better given a small training set
- Unfortunately, solution may not be tractable in many applications

Remarks

- Bayesian estimation:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \mathrm{P}(\boldsymbol{y}\,|\,\boldsymbol{x},\mathbb{X}) = \arg\max_{\boldsymbol{y}} \int \mathrm{P}(\boldsymbol{y},\Theta\,|\,\boldsymbol{x},\mathbb{X})d\Theta$$

- Usually generalizes much better given a small training set
- Unfortunately, solution may not be tractable in many applications
- Even tractable, incurs high computation cost
  - Not suitable for large-scale learning tasks

再夠大的data set下 效果可能比ML/MAP好一點點
但computation cost會高很多

Reference I

[1] Deepak Roy Chittajallu.
Why is the error function minimized in logistic regression convex?
http://mathgotchas.blogspot.tw/2011/10/why-is-error-function-minimized-in.html, 2011.

[2] Harald Cramér.
Mathematical Methods of Statistics.
Princeton university press, 1946.