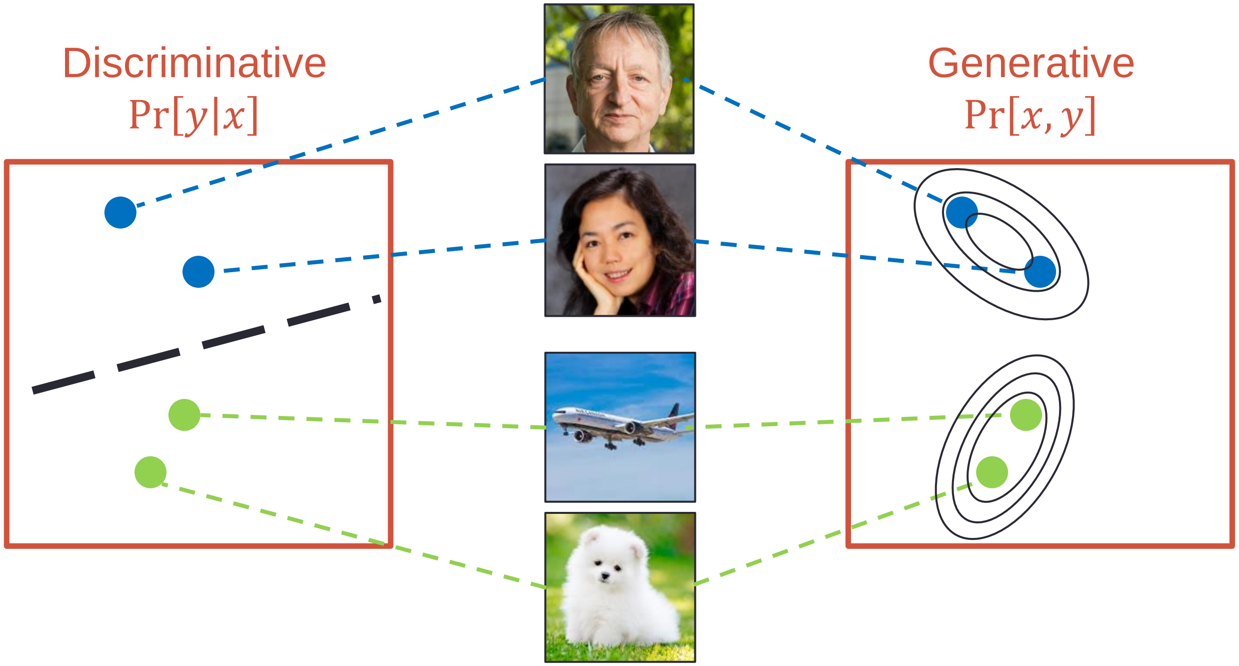
GENERATIVE MODELS

Ronen Nir & Itay Evron

Generative models vs Discriminative models

- Generative models solve a harder task
- But sometimes it's easy to learn their parameters
- Can gain a better "understanding" of structures in data



Sources: Google's ML crash course

Tutorial outline

- Maximum Likelihood Estimation (MLE)
- Naïve Bayes
- Maximum a Posteriori estimation (MAP)

MAXIMUM-LIKELIHOOD ESTIMATION

Maximum Likelihood Estimation (MLE)

- Setting: Independent draws $x_1, \dots, x_m \sim \mathcal{D}_\Theta$ from a parametric distribution
- Likelihood: Probability of the observed data given parameters

$$L(x_1, \dots, x_m | \Theta) = \Pr(S|\Theta) = \Pr(x_1, \dots, x_m | \Theta)$$

- MLE maximizes the likelihood

$$\widehat{\Theta}_{\mathrm{MLE}} = \mathrm{argmax}_\Theta \, L(x_1, \dots, x_m | \Theta)$$

- Equivalently,

$$\widehat{\Theta}_{\mathrm{MLE}} = \mathrm{argmax}_\Theta \ln L(x_1, \dots, x_m | \Theta)$$

Recall: MLE for Gaussian variables

- Given i.i.d Gaussian variables $x_1, \ldots, x_m \sim \mathcal{N}(\mu, \sigma^2)$ we wish to estimate μ, σ^2
- Intuition: given the observations $2, 9, 4$ how would you "guess" μ?

- In the lecture, we saw that:
  - The likelihood is:

$$L(S|\mu, \sigma^2) = \Pr(x_1, \ldots, x_m|\mu, \sigma^2) = \prod_i \Pr(x_i|\mu, \sigma^2) = \prod_i \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(x_i-\mu)^2}{2\sigma^2}\right\}$$

  - The log-likelihood is: $\ln L(x_1, \ldots, x_m|\mu, \sigma^2) = -m \cdot \ln \sigma\sqrt{2\pi} - \sum_i \frac{(x_i-\mu)^2}{2\sigma^2}$

  - By simple differentiation, one can find the MLEs:

$$\hat{\mu}_{\text{MLE}} = \frac{1}{m}\sum_i x_i \quad , \qquad \hat{\sigma}^2_{\text{MLE}} = \frac{1}{m}\sum_i (x_i - \hat{\mu}_{\text{MLE}})^2$$

NAÏVE BAYES

Naïve Bayes

- Wish to find the most probable label by maximizing the posterior probability

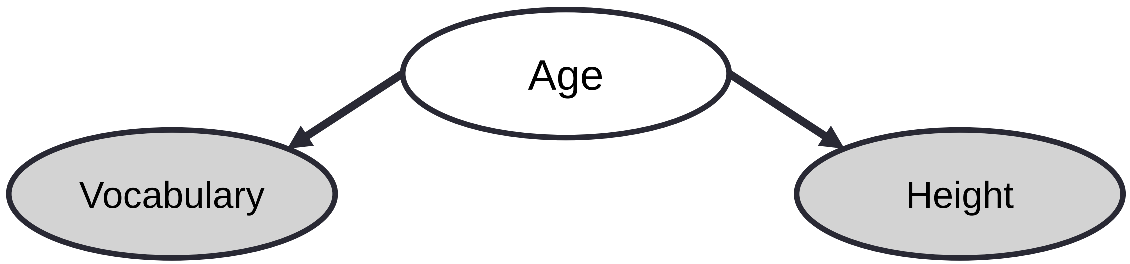
$$\hat{y} = h(\boldsymbol{x}) = \operatorname{argmax}_y \Pr(y|\boldsymbol{x}) \overset{\text{Bayes}}{=} \operatorname{argmax}_y \frac{\Pr(\boldsymbol{x}|y)\Pr(y)}{\Pr(\boldsymbol{x})} = \operatorname{argmax}_y \Pr(\boldsymbol{x}|y)\Pr(y)$$

- We make a naïve assumption – the coordinates are conditionally independent

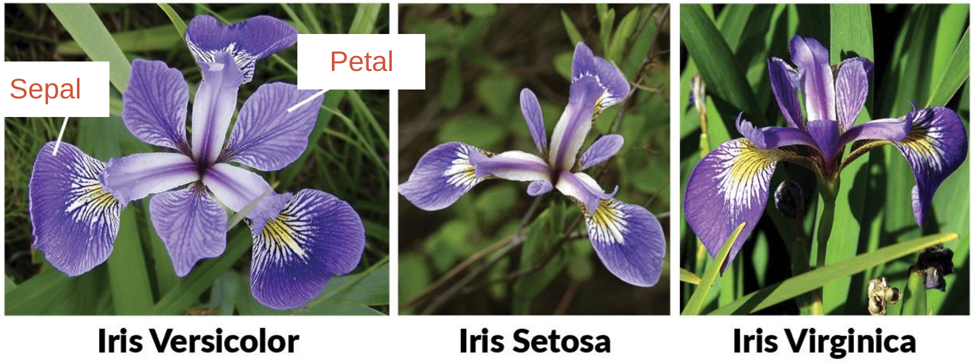
$$\overset{\text{Naïve}}{=} \operatorname{argmax}_y \Pr(y) \prod_{k=1}^{d} \Pr(X[k] = x[k] \mid y)$$

Estimate these probabilities
(with MLE for instance)

- Often works well in practice, despite being a naïve assumption.
- Sometimes realistic! $\Pr(\text{height}, \text{vocabulary} \mid \text{age}) \approx \Pr(\text{height} \mid \text{age}) \cdot \Pr(\text{vocabulary} \mid \text{age})$



Demo: Iris dataset

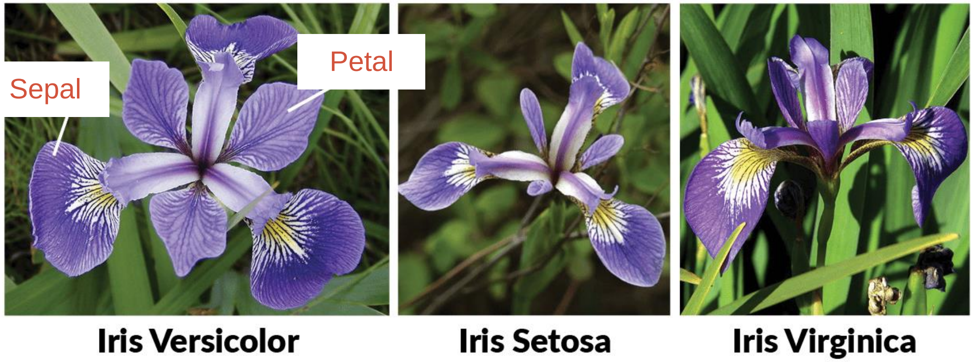


Source: ML in R

- Three classes
  - Setosa, Versicolor, and Virginica
- Four features
  - Sepal length, sepal width, petal length, petal width (in cm)
  - Sepal is עלה גביע,
  - Petal is עלה כותרת

Demo: Iris dataset

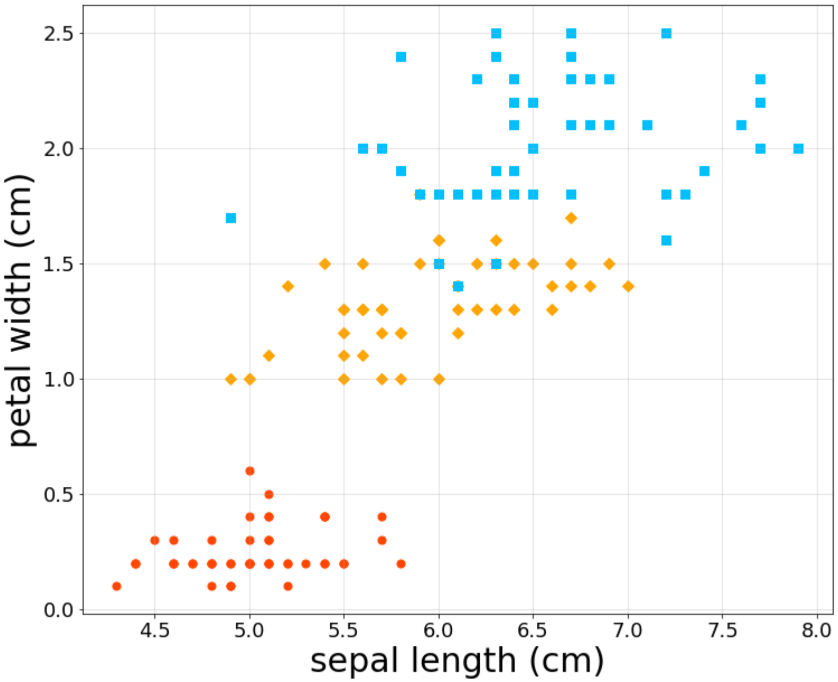
- Three classes
  - Setosa, Versicolor, and Virginica



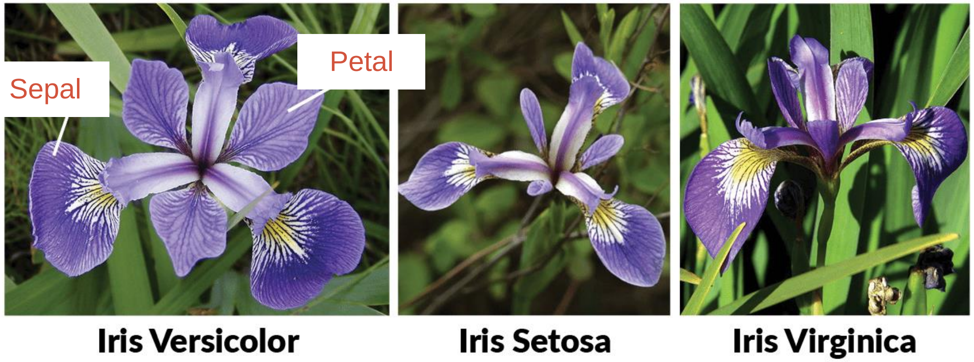
Source: ML in R

- Today we focus on two features
  - Sepal length ($x[1]$) and Petal width ($x[2]$)

Visualization of the two features



Demo: Iris dataset

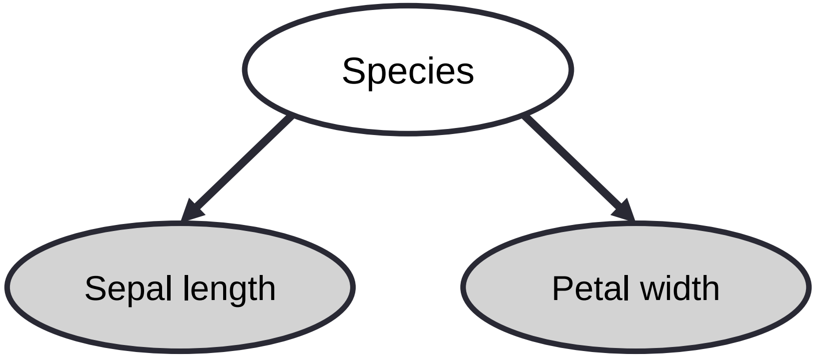


Source: ML in R

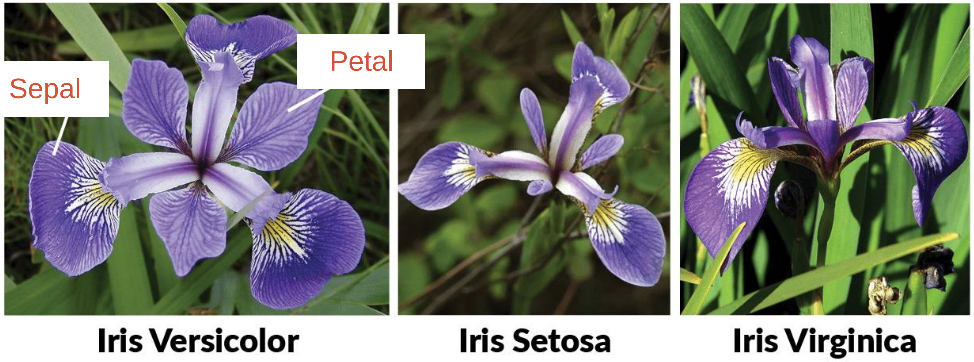
- Three classes
  - Setosa, Versicolor, and Virginica
- Today we focus on two features
  - Sepal length ($x[1]$) and Petal width ($x[2]$)
- Naïve assumption:
  - Given the species (y),

    features are independent of each other

Graphical model:



Demo: Iris dataset

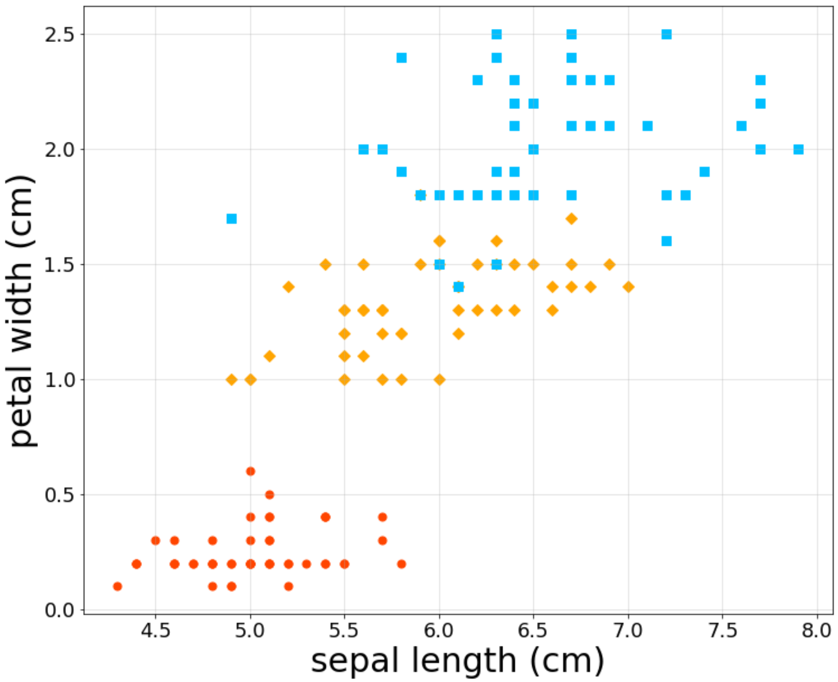


Source: ML in R

- Three classes
  - Setosa, Versicolor, and Virginica

- Today we focus on two features
  - Sepal length ($x[1]$) and Petal width ($x[2]$)

Visualization of the two features



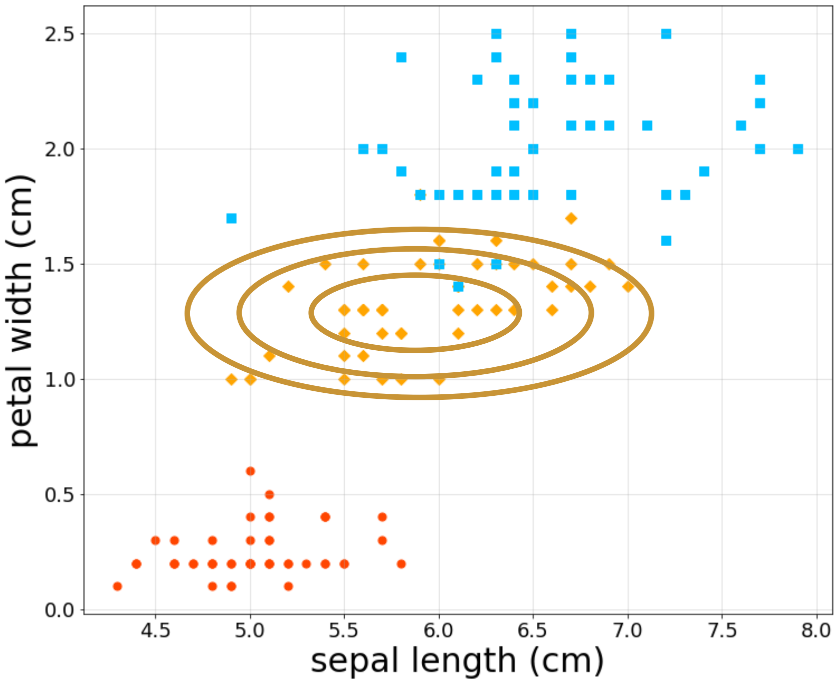
- Naïve assumption:
  - Given the species (y),

    features are independent of each other

  - Q: in the plot, do they look independent?

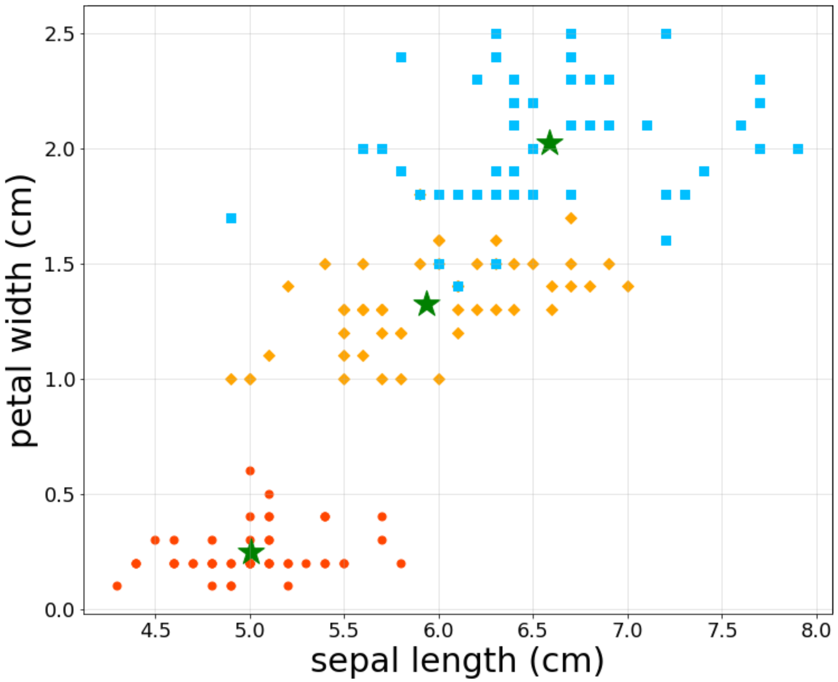
Demo: Gaussian Naïve Bayes

- Naïve assumption: given the species (y), features are independent
  - Use a Naïve Bayes classifier $\hat{y} = \text{argmax}_y \Pr(y) \prod_{k=1}^{d} \Pr(X[k] = x[k] \mid y)$
- Modeling: assume probabilities $\Pr(X[k] = x[k] \mid y)$ are distributed $\mathcal{N}\left(\mu_y[k], \sigma_k^2\right)$
- Goal: fit multivariate Gaussians to the data
  - Estimate a different mean $\boldsymbol{\mu}_y \in \mathbb{R}^d$ per class
  - Estimate a different variance $\sigma_k^2 \in \mathbb{R}_+$ per feature



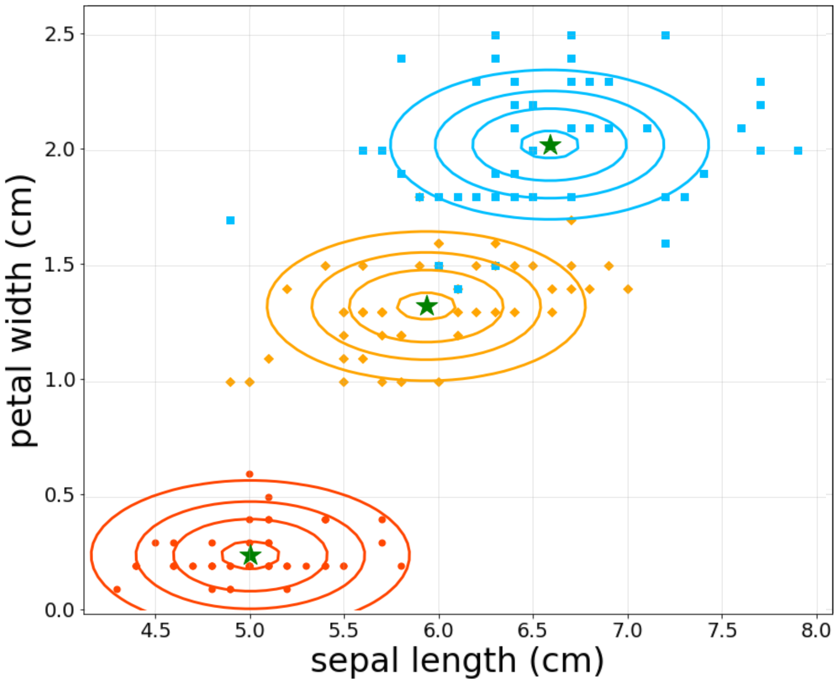
Demo: Gaussian Naïve Bayes

- Naïve assumption: given the species (y), features are independent
  - Use a Naïve Bayes classifier $\hat{y} = \operatorname{argmax}_y \Pr(y) \prod_{k=1}^{d} \Pr(X[k] = x[k] \mid y)$
- Modeling: assume probabilities $\Pr(X[k] = x[k] \mid y)$ are distributed $\mathcal{N}\left(\mu_y[k], \sigma_k^2\right)$
- Goal: fit multivariate Gaussians to the data
  - Estimate a different mean $\boldsymbol{\mu}_y \in \mathbb{R}^d$ per class
  - Estimate a different variance $\sigma_k^2 \in \mathbb{R}_+$ per feature
- Estimation: maximize likelihood (MLE)
  - Means: $\widehat{\mu_y}[k] = \frac{1}{\#\{y_i = y\}} \sum_{i:\, y_i = y} x_i[k]$



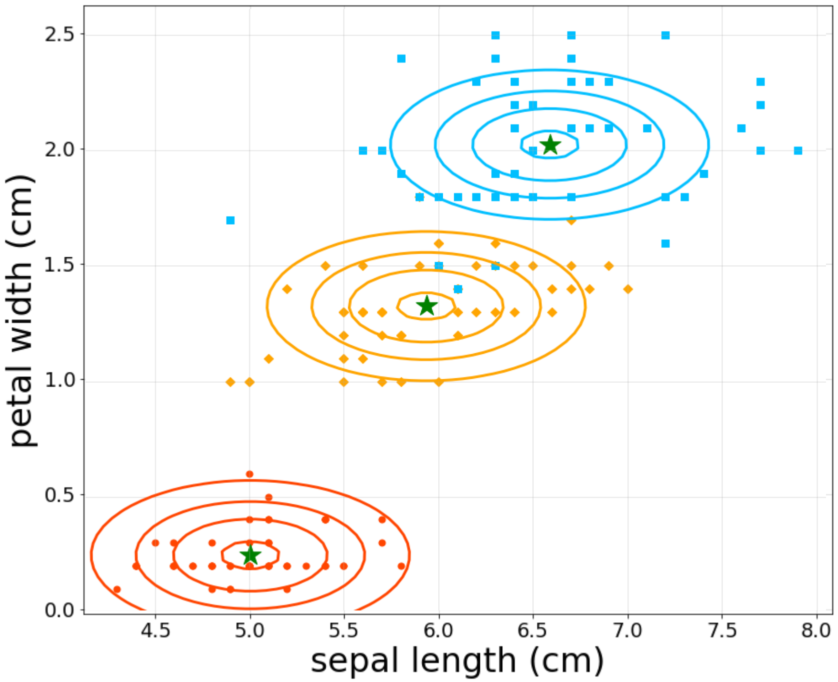
Demo: Gaussian Naïve Bayes

- Naïve assumption: given the species (y), features are independent
  - Use a Naïve Bayes classifier $\hat{y} = \operatorname{argmax}_y \Pr(y) \prod_{k=1}^{d} \Pr(X[k] = x[k] \mid y)$
- Modeling: assume probabilities $\Pr(X[k] = x[k] \mid y)$ are distributed $\mathcal{N}\left(\mu_y[k], \sigma_k^2\right)$
- Goal: fit multivariate Gaussians to the data
  - Estimate a different mean $\boldsymbol{\mu}_y \in \mathbb{R}^d$ per <u>class</u>
  - Estimate a different variance $\sigma_k^2 \in \mathbb{R}_+$ per <u>feature</u>
- Estimation: maximize likelihood (MLE)
  - Means: $\widehat{\mu_y}[k] = \frac{1}{\#\{y_i = y\}} \sum_{i:\, y_i = y} x_i[k]$
  - Variances: $\widehat{\sigma_k}^2 = \frac{1}{m} \sum_i \left(x_i[k] - \widehat{\mu_{y_i}}[k]\right)^2$ (extra)
- Q: why are the Gaussians “axis-aligned”?



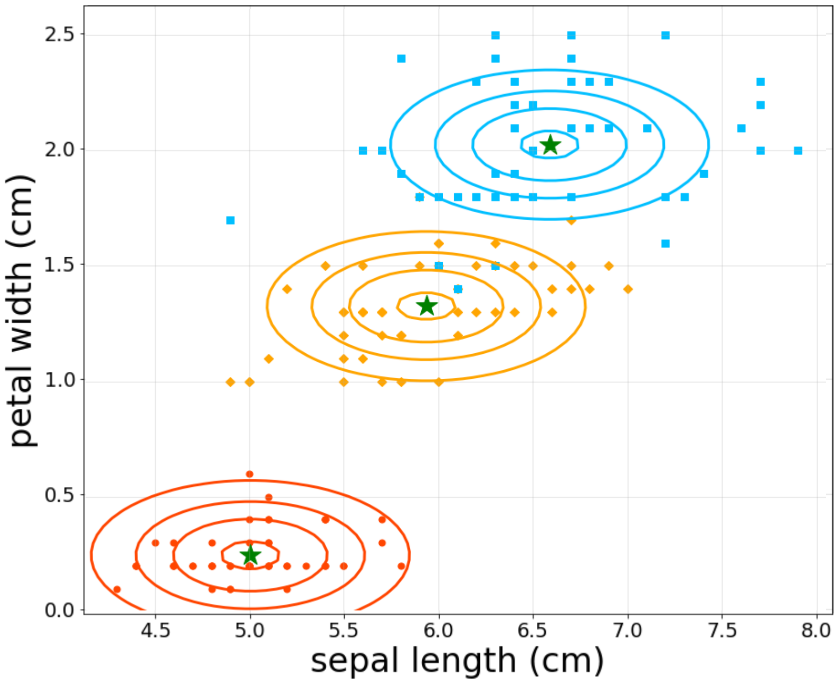
Demo: Gaussian Naïve Bayes

- Naïve assumption: given the species (y), features are independent
  - Use a Naïve Bayes classifier $\hat{y} = \operatorname{argmax}_y \Pr(y) \prod_{k=1}^{d} \Pr(X[k] = x[k] \mid y)$
- Modeling: assume probabilities $\Pr(X[k] = x[k] \mid y)$ are distributed $\mathcal{N}\left(\mu_y[k], \sigma_k^2\right)$
- Goal: fit multivariate Gaussians to the data
  - Estimate a different mean $\boldsymbol{\mu}_y \in \mathbb{R}^d$ per <u>class</u>
  - Estimate a different variance $\sigma_k^2 \in \mathbb{R}_+$ per <u>feature</u>
- Estimation: maximize likelihood (MLE)
  - Means: $\widehat{\mu_y}[k] = \frac{1}{\#\{y_i = y\}} \sum_{i:\, y_i = y} x_i[k]$
  - Variances: $\widehat{\sigma_k}^2 = \frac{1}{m} \sum_i \left(x_i[k] - \widehat{\mu_{y_i}}[k]\right)^2$ (extra)
- Q: why all Gaussians "look" the same?



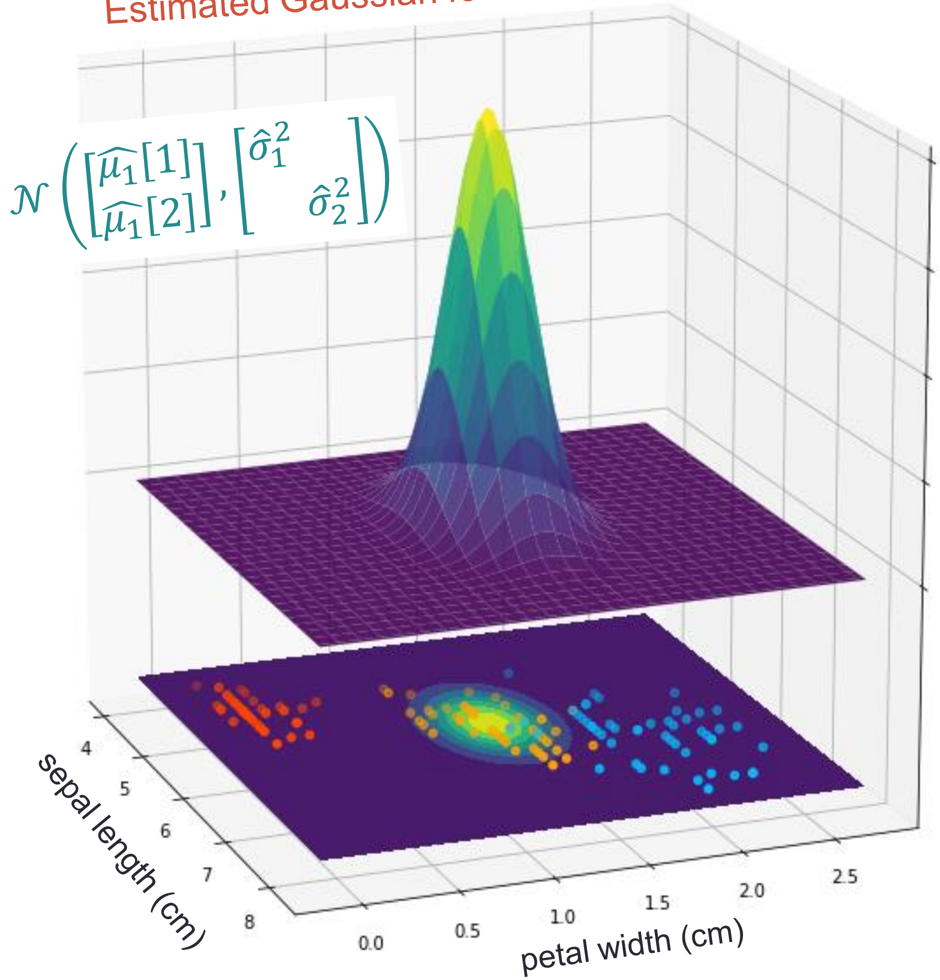
Demo: Gaussian Naïve Bayes

- Naïve assumption: given the species (y), features are independent
  - Use a Naïve Bayes classifier $\hat{y} = \operatorname{argmax}_y \Pr(y) \prod_{k=1}^{d} \Pr(X[k] = x[k] \mid y)$
- Modeling: assume probabilities $\Pr(X[k] = x[k] \mid y)$ are distributed $\mathcal{N}\left(\mu_y[k], \sigma_k^2\right)$
- Goal: fit multivariate Gaussians to the data
  - Estimate a different mean $\boldsymbol{\mu}_y \in \mathbb{R}^d$ per class
  - Estimate a different variance $\sigma_k^2 \in \mathbb{R}_+$ per feature
- Estimation: maximize likelihood (MLE)
  - Means: $\widehat{\mu_y}[k] = \frac{1}{\#\{y_i = y\}} \sum_{i:\, y_i = y} x_i[k]$
  - Variances: $\widehat{\sigma_k}^2 = \frac{1}{m} \sum_i \left(x_i[k] - \widehat{\mu_{y_i}}[k]\right)^2$ (extra)
  - Marginal: $\widehat{\Pr}(y) = \frac{1}{m} \#\{y_i = y\}$

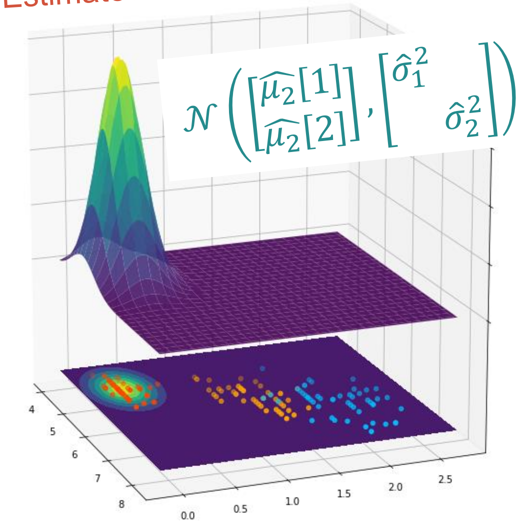


Demo: Estimated Gaussians

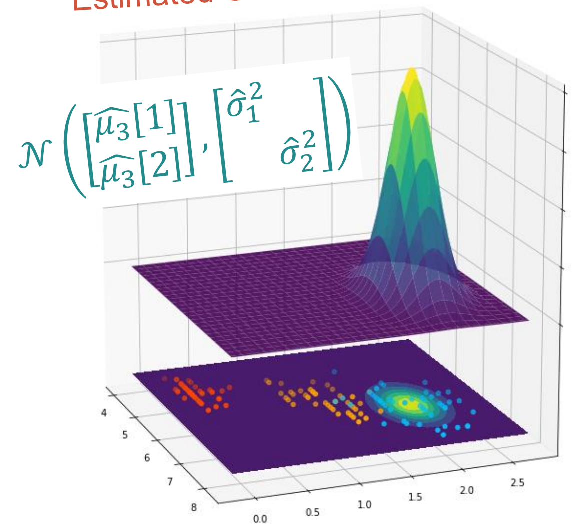
Estimated Gaussian for 1st class



Estimated Gaussian for 2nd class



Estimated Gaussian for 3rd class



Demo: Making a prediction

- Naïve Bayes rule: $\hat{y} = \text{argmax}_y \Pr(y|\boldsymbol{x}) = \text{argmax}_y \Pr(y) \prod_{k=1}^{d} \Pr(X[k] = x[k] \mid y)$
- Prediction using our estimators:

$$h(x) = \text{argmax}_y \widehat{\Pr}(y) \prod_{k=1}^{d} \frac{1}{\widehat{\sigma_k}\sqrt{2\pi}} \exp\left\{-\frac{\left(x - \widehat{\mu_y}\right)^2}{2\widehat{\sigma_k}^2}\right\}$$

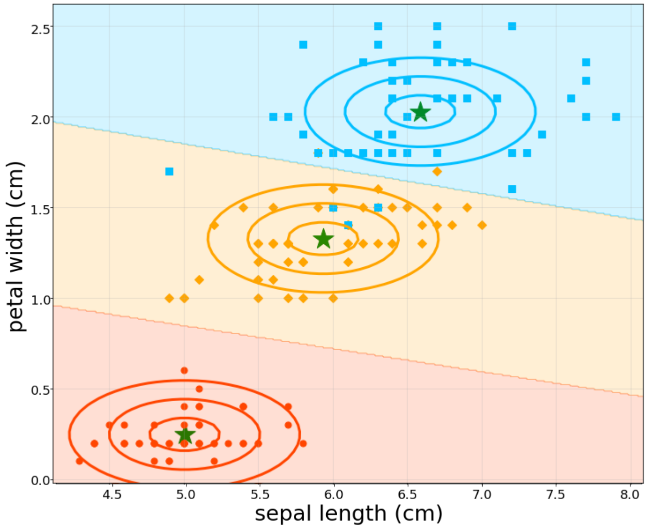
  - The predictor asks: which Gaussian gives the maximal probability to seeing x?

    (normalized by the “prior”/marginal probability)

- Assuming same covariance for all classes,

  decision boundaries are linear (proof in lecture)

- Q: What is the training error here?



MAXIMUM A POSTERIORI ESTIMATION

Recall: Least squares as MLE

- We saw the following theorem (lecture 09):
  - Assuming a noisy linear model: $y_i = \boldsymbol{x}_i^\top \boldsymbol{w} + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1)$ (i.i.d)
    - Note: a sample $\boldsymbol{x}_i$ is not random but the noise ε_i is.
  - Solving Least Squares (LS) is equivalent to Maximum-Likelihood Estimation (MLE):

$$\widehat{\boldsymbol{w}}_{\mathrm{LS}} \triangleq \operatorname{argmin}_{\boldsymbol{w}} \frac{1}{m}\sum_i (\boldsymbol{w}^\top \boldsymbol{x}_i - y_i)^2 = \operatorname{argmax}_{\boldsymbol{w}} L(\boldsymbol{w}; S) \triangleq \widehat{\boldsymbol{w}}_{\mathrm{MLE}}$$

- Exercise (like in lecture 09):

1. Prove that the likelihood is as follows, by justifying the equalities below:

$$L(\boldsymbol{w}; S) \triangleq P(\{(\boldsymbol{x}_i, y_i)\}_{i=1}^m \,|\boldsymbol{w}\,) = \prod_{i=1}^m P(y_i, |\boldsymbol{x}_i, \boldsymbol{w}) = \prod_{i=1}^m \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right\}$$

$$= (2\pi)^{-\frac{m}{2}} \cdot \exp\left\{-\frac{1}{2}\sum_{i=1}^m \left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right\}$$

Recall: Least squares as MLE

- We saw the following theorem (lecture 09):
  - Assuming a noisy linear model: $y_i = \boldsymbol{x}_i^\top \boldsymbol{w} + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1)$ (i.i.d)
    - Note: a sample $\boldsymbol{x}_i$ is not random but the noise ε_i is.
  - Solving Least Squares (LS) is equivalent to Maximum-Likelihood Estimation (MLE):

$$\widehat{\boldsymbol{w}}_{\mathrm{LS}} \triangleq \mathrm{argmin}_{\boldsymbol{w}} \frac{1}{m} \sum_i (\boldsymbol{w}^\top \boldsymbol{x}_i - y_i)^2 = \mathrm{argmax}_{\boldsymbol{w}}\, L(\boldsymbol{w}; S) \triangleq \widehat{\boldsymbol{w}}_{\mathrm{MLE}}$$

- Exercise (like in lecture 09):

1. The likelihood is: $L(\boldsymbol{w}; S) = (2\pi)^{-\frac{m}{2}} \cdot \exp\left\{-\frac{1}{2}\sum_{i=1}^m \left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right\}$
2. Prove the theorem above.

$$\widehat{\boldsymbol{w}}_{\mathrm{MLE}} \triangleq \mathrm{argmax}_{\boldsymbol{w}}\, L(\boldsymbol{w}; S) =$$

$$= \mathrm{argmin}_{\boldsymbol{w}} \frac{1}{m} \sum_i (\boldsymbol{w}^\top \boldsymbol{x}_i - y_i)^2 \triangleq \widehat{\boldsymbol{w}}_{\mathrm{LS}}$$

Recall: Least squares as MLE

- We saw the following theorem (lecture 09):
  - Assuming a noisy linear model: $y_i = \boldsymbol{x}_i^\top \boldsymbol{w} + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1)$ (i.i.d)
    - Note: a sample $\boldsymbol{x}_i$ is not random but the noise ε_i is.
  - Solving Least Squares (LS) is equivalent to Maximum-Likelihood Estimation (MLE):

$$\widehat{\boldsymbol{w}}_{\mathrm{LS}} \triangleq \operatorname{argmin}_{\boldsymbol{w}} \frac{1}{m}\sum_i (\boldsymbol{w}^\top \boldsymbol{x}_i - y_i)^2 = \operatorname{argmax}_{\boldsymbol{w}} L(\boldsymbol{w}; S) \triangleq \widehat{\boldsymbol{w}}_{\mathrm{MLE}}$$

- Exercise (like in lecture 09):

1. The likelihood is: $L(\boldsymbol{w}; S) = (2\pi)^{-\frac{m}{2}} \cdot \exp\left\{-\frac{1}{2}\sum_{i=1}^m \left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right\}$
2. Prove the theorem above.

$$\widehat{\boldsymbol{w}}_{\mathrm{MLE}} \triangleq \operatorname{argmax}_{\boldsymbol{w}} L(\boldsymbol{w}; S) = \operatorname{argmax}_{\boldsymbol{w}} \ln\left[(2\pi)^{-\frac{m}{2}} \cdot \exp\left\{-\frac{1}{2}\sum_{i=1}^m \left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right\}\right]$$

$$= \operatorname{argmax}_{\boldsymbol{w}} \left(-\frac{m}{2}\ln(2\pi) - \frac{1}{2}\sum_{i=1}^m \left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right) = \operatorname{argmin}_{\boldsymbol{w}} \frac{1}{m}\sum_i (\boldsymbol{w}^\top \boldsymbol{x}_i - y_i)^2 \triangleq \widehat{\boldsymbol{w}}_{\mathrm{LS}}$$

Maximum a Posteriori (MAP) Estimation

- Maximizes the posterior probability of the parameters given the observations

$$\widehat{\Theta}_{\mathrm{MAP}} = \operatorname{argmax}_{\Theta} \Pr[\Theta|S] = \operatorname{argmax}_{\Theta} \frac{\Pr[S|\Theta] \Pr[\Theta]}{\Pr[S]} = \operatorname{argmax}_{\Theta} \underbrace{\Pr[S|\Theta]}_{\triangleq L(\Theta;\, S)} \Pr[\Theta]$$

- Assumes a prior on the parameters themselves!
- Notice the difference from MLE

$$\widehat{\Theta}_{\mathrm{MLE}} = \operatorname{argmax}_{\Theta} \Pr[S|\Theta]$$

  - Q: When are they equivalent?

Exercise: Ridge regression as MAP

- Like before, we assume a noisy linear model: $y_i = \boldsymbol{x}_i^\top \boldsymbol{w} + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1)$ (i.i.d)
  - Recall, The likelihood is: $L(\boldsymbol{w}; S) = (2\pi)^{-\frac{m}{2}} \cdot \exp\left\{-\frac{1}{2}\sum_{i=1}^{m}\left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right\}$
- We further assume a prior on the weights: $w[k] \sim \mathcal{N}(0, {}^{1}/{}_{\lambda m}), \quad \lambda > 0$.

1. Express the prior PDF $p(\boldsymbol{w})$

Exercise: Ridge regression as MAP

- Like before, we assume a noisy linear model: $y_i = \boldsymbol{x}_i^\top \boldsymbol{w} + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1)$ (i.i.d)
  - Recall, The likelihood is: $L(\boldsymbol{w}; S) = (2\pi)^{-\frac{m}{2}} \cdot \exp\left\{-\frac{1}{2}\sum_{i=1}^{m}\left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right\}$
- We further assume a prior on the weights: $w[k] \sim \mathcal{N}(0, {}^{1}/_{\lambda m}), \quad \lambda > 0.$

1. Express the prior PDF $p(\boldsymbol{w}) = \prod_{k=1}^{d} p(w[k])$

$$= \prod_{k=1}^{d} \sqrt{\frac{\lambda m}{2\pi}} \exp\left\{-\frac{\lambda}{2} w[k]^2\right\}$$

$$= \left(\frac{\lambda m}{2\pi}\right)^{\frac{d}{2}} \exp\left\{-\frac{\lambda m}{2}\sum_{k=1}^{d} w[k]^2\right\}$$

Exercise: Ridge regression as MAP

- Like before, we assume a noisy linear model: $y_i = \boldsymbol{x}_i^\top \boldsymbol{w} + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1)$ (i.i.d)
  - Recall, The likelihood is: $L(\boldsymbol{w}; S) = (2\pi)^{-\frac{m}{2}} \cdot \exp\left\{-\frac{1}{2}\sum_{i=1}^{m}\left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right\}$
- We further assume a prior on the weights: $w[k] \sim \mathcal{N}(0, {}^1/_{\lambda m}), \quad \lambda > 0.$

1. The prior PDF is $p(\boldsymbol{w}) = \left(\frac{\lambda m}{2\pi}\right)^{\frac{d}{2}} \exp\left\{-\frac{\lambda m}{2}\sum_{k=1}^{d} w[k]^2\right\}$

2. Prove: under the assumptions above,

   solving Ridge regression is equivalent to Maximum a posteriori Estimation (MAP):

$$\widehat{\boldsymbol{w}}_{\text{Ridge}} \triangleq \operatorname{argmin}_{\boldsymbol{w}} \frac{1}{m}\sum_i (\boldsymbol{w}^\top \boldsymbol{x}_i - y_i)^2 + \lambda \|\boldsymbol{w}\|_2^2 = \operatorname{argmax}_{\boldsymbol{w}} L(\boldsymbol{w}; S)\, p(\boldsymbol{w}) \triangleq \widehat{\boldsymbol{w}}_{\text{MAP}}$$

Exercise: Ridge regression as MAP

- Like before, we assume a noisy linear model: $y_i = \boldsymbol{x}_i^\top \boldsymbol{w} + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1)$ (i.i.d)
  - Recall, The likelihood is: $L(\boldsymbol{w}; S) = (2\pi)^{-\frac{m}{2}} \cdot \exp\left\{-\frac{1}{2}\sum_{i=1}^{m}\left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2\right\}$
- We further assume a prior on the weights: $w[k] \sim \mathcal{N}(0, {}^1/_{\lambda m}), \quad \lambda > 0.$

1. The prior PDF is $p(\boldsymbol{w}) = \left(\frac{\lambda m}{2\pi}\right)^{\frac{d}{2}} \exp\left\{-\frac{\lambda m}{2}\sum_{k=1}^{d} w[k]^2\right\}$
2. Proof: $\widehat{\boldsymbol{w}}_{\mathrm{MAP}} \triangleq \mathrm{argmax}_{\boldsymbol{w}}\, L(\boldsymbol{w}; S)p(\boldsymbol{w})$

$$= \mathrm{argmax}_{\boldsymbol{w}}(\ln L(\boldsymbol{w}; S) + \ln p(\boldsymbol{w}))$$

$$= \mathrm{argmax}_{\boldsymbol{w}}\left(-\frac{m}{2}\ln(2\pi) - \frac{1}{2}\sum_{i=1}^{m}\left(\boldsymbol{x}_i^\top \boldsymbol{w} - y_i\right)^2 + \frac{d}{2}\ln\left(\frac{\lambda m}{2\pi}\right) - \frac{\lambda m}{2}\sum_{k=1}^{d} w[k]^2\right)$$

$$= \mathrm{argmin}_{\boldsymbol{w}} \frac{1}{m}\sum_{i}(\boldsymbol{w}^\top \boldsymbol{x}_i - y_i)^2 + \lambda\|\boldsymbol{w}\|_2^2 \triangleq \widehat{\boldsymbol{w}}_{\mathrm{Ridge}}$$

Summary

- MLE finds the most likely model to have generated the data

$$\widehat{\Theta}_{\mathrm{MLE}} = \mathrm{argmax}_{\Theta} \Pr[S|\Theta]$$

- MAP finds the most probable model based on the data

$$\widehat{\Theta}_{\mathrm{MAP}} = \mathrm{argmax}_{\Theta} \Pr[\Theta|S] = \mathrm{argmax}_{\Theta} \Pr[S|\Theta] \Pr[\Theta]$$

- Naïve Bayes makes a naïve assumption that the features are conditionally independent given the label

$$\Pr[\boldsymbol{x}|y] = \prod_{k=1}^{d} \Pr[X[k] = x_k | Y = y]$$