

MACHINE LEARNING: THEORETICAL CONCEPTS UE

Assignment 1: Maximum likelihood, Generalization Error



Institute for Machine Learning

Contact

LVA Head: Johannes Brandstetter, Johannes Kofler

Staff: M. Holzleitner, J. Arjona-Medina, A. Mayr, T. Adler,
H. Ramsauer

Institute for Machine Learning
Johannes Kepler University
Altenberger Str. 69
A-4040 Linz

E-Mail: `theoretical@ml.jku.at`

Only mails to this list are answered!

Institute Homepage

Generalization Error

Supervised learning:

- some real world process produces data $\mathbf{x} \in \mathbb{R}^d$
- to every data point we want to infer a $y \in \mathbb{R}$ that is either a category (classification) or a value (regression)
- for a set of data points $X = \{\mathbf{x}^1, \dots, \mathbf{x}^l\}$ we know the associated $\{y^1, \dots, y^l\}$
- we call $\{\mathbf{z}^1, \dots, \mathbf{z}^l\}$ the training data, where $\mathbf{z}^i = (\mathbf{x}^i, y^i)$

What does it mean to learn from data?

- learning is model selection
- supervised learning: select a model that minimizes the prediction error on future data
- i.e. we want our model to generalize from the training data to future data

Generalization Error

What does it mean to learn from data? More formal:

- selecting a model, i.e. a function $g(\mathbf{x})$ that associates y to input \mathbf{x}
- if the model is parametrized with an vector \mathbf{w} we write $g(\mathbf{x}; \mathbf{w})$
- we want to select a "good" model (i.e. good parameters)
- we measure the performance of our model with a loss function $L(y, g(\mathbf{x}; \mathbf{w}))$

Typical loss functions

- zero-one-loss

$$L(y, g(\mathbf{x}; \omega)) = \begin{cases} 0 & \text{for } y = g(\mathbf{x}; \omega) \\ 1 & \text{for } y \neq g(\mathbf{x}; \omega) \end{cases}$$

- quadratic loss

$$L(y, g(\mathbf{x}; \omega)) = (y - g(\mathbf{x}; \omega))^2$$

Generalization Error

What does it mean to generalize?

- the **generalization error** which is the **expected loss on future data** should be as low as possible
- the generalization error, also called the risk R is the functional:

$$R(g(\cdot; \mathbf{w})) = E_{\mathbf{z}} (L(y, g(\mathbf{x}; \mathbf{w}))) = \int_Z L(y, g(\mathbf{x}; \mathbf{w})) p(\mathbf{z}) d\mathbf{z}$$

where $p(\mathbf{z})$ denotes the probability of \mathbf{z} and Z is the set of all future \mathbf{z} .

Since we don't have all future data, we need to approximate the risk

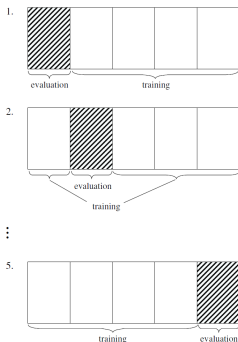
- we choose m samples from $\{\mathbf{z}^1, \dots, \mathbf{z}^l\}$
- this is the so called "test set" $\{\mathbf{z}^1, \dots, \mathbf{z}^m\}$, $m < l$
- assuming the \mathbf{z} are iid and m is large enough we can approximate the risk

$$R(g(\cdot; \mathbf{w})) \approx \frac{1}{m} \sum_{i=1}^m (L(y^i, g(\mathbf{x}^i; \mathbf{w})))$$

Empirical Estimation of Risk

If we don't have much data:

- Cross Validation: using folds of the data



CV risk is an almost unbiased estimator for the risk

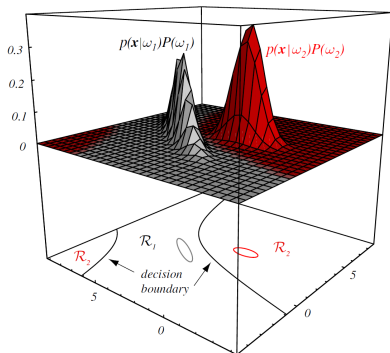
Minimal Risk for Gaussian Classification

Density function of multivariate Gaussian:

$$\mathcal{N}(\mathbf{x}; \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} e^{-\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu)}$$

Classification task where the data for each class is drawn from a Gaussian

- $p(\mathbf{x}|y = 1) \propto \mathcal{N}(\mu_1, \Sigma_1)$
- $p(\mathbf{x}|y = -1) \propto \mathcal{N}(\mu_{-1}, \Sigma_{-1})$



A two-dimensional classification task where the data for each class are drawn from a Gaussian (black: class 1, red: class -1). The optimal decision boundaries are two hyperbolas. Here $\omega_1 \equiv y = 1$ and $\omega_2 \equiv y = -1$. In the gray regions $p(y = 1 | \mathbf{x}) > p(y = -1 | \mathbf{x})$ holds and in the red regions the opposite holds. Copyright © 2001 John Wiley & Sons, Inc.

Minimal Risk for Gaussian Classification

We define the regions

- of class 1 as $X_1 = \{\mathbf{x} \mid g(\mathbf{x}) > 0\}$
- of class -1 as $X_{-1} = \{\mathbf{x} \mid g(\mathbf{x}) < 0\}$

and the loss function as

$$L(y, g(\mathbf{x}; \omega)) = \begin{cases} 0 & \text{for } y \cdot g(\mathbf{x}; \omega) > 0 \\ 1 & \text{for } y \cdot g(\mathbf{x}; \omega) < 0 \end{cases}$$

Using the zero-one-loss we obtain for the risk

$$\begin{aligned} R(g(\cdot; \omega)) &= \int_{X_1} p(y = -1 \mid \mathbf{x}) p(\mathbf{x}) d\mathbf{x} + \int_{X_{-1}} p(y = 1 \mid \mathbf{x}) p(\mathbf{x}) d\mathbf{x} \\ &= \int_X \left\{ \begin{array}{ll} p(y = -1 \mid \mathbf{x}) & \text{for } g(\mathbf{x}) > 0 \\ p(y = 1 \mid \mathbf{x}) & \text{for } g(\mathbf{x}) < 0 \end{array} \right\} p(\mathbf{x}) d\mathbf{x} . \end{aligned}$$

Minimal Risk for Gaussian Classification

Risk can be minimized by

- choosing the smaller value of $p(y = -1 | \mathbf{x})$ and $p(y = 1 | \mathbf{x})$.

Therefore, risk is minimal if

$$g(\mathbf{x}; \omega) \begin{cases} > 0 & \text{for } p(y = 1 | \mathbf{x}) > p(y = -1 | \mathbf{x}) \\ < 0 & \text{for } p(y = -1 | \mathbf{x}) > p(y = 1 | \mathbf{x}) \end{cases}$$

The minimal risk is

$$R_{\min} = \int_X \min\{p(y = -1 | \mathbf{x}), p(y = 1 | \mathbf{x})\} p(\mathbf{x}) d\mathbf{x}$$

Discriminant Function

A discriminant function which minimizes the future risk is

$$\begin{aligned} g(\mathbf{x}) &= p(y = 1 \mid \mathbf{x}) - p(y = -1 \mid \mathbf{x}) \\ &= \frac{1}{p(\mathbf{x})} (p(\mathbf{x} \mid y = 1) p(y = 1) - p(\mathbf{x} \mid y = -1) p(y = -1)) , \end{aligned}$$

- only the difference in the last brackets matters because $p(\mathbf{x}) > 0$
- optimal discriminant function is not unique since difference of strict monotone mappings of $p(y = 1 \mid \mathbf{x})$ and $p(y = -1 \mid \mathbf{x})$ keep the sign

Take the logarithm → more convenient discriminant function which also minimizes the future risk:

$$\begin{aligned} g(\mathbf{x}) &= \ln p(y = 1 \mid \mathbf{x}) - \ln p(y = -1 \mid \mathbf{x}) \\ &= \ln \frac{p(\mathbf{x} \mid y = 1)}{p(\mathbf{x} \mid y = -1)} + \ln \frac{p(y = 1)}{p(y = -1)} . \end{aligned}$$

Discriminant Function for Gaussian Classific.

$$\begin{aligned}g(\mathbf{x}) &= -\frac{1}{2} (\mathbf{x} - \mu_1)^T \Sigma_1^{-1} (\mathbf{x} - \mu_1) - \frac{d}{2} \ln 2\pi - \frac{1}{2} \ln |\Sigma_1| + \ln p(y = 1) \\&\quad + \frac{1}{2} (\mathbf{x} - \mu_{-1})^T \Sigma_{-1}^{-1} (\mathbf{x} - \mu_{-1}) + \frac{d}{2} \ln 2\pi + \frac{1}{2} \ln |\Sigma_{-1}| - \ln p(y = -1) \\&= -\frac{1}{2} (\mathbf{x} - \mu_1)^T \Sigma_1^{-1} (\mathbf{x} - \mu_1) - \frac{1}{2} \ln |\Sigma_1| + \ln p(y = 1) \\&\quad + \frac{1}{2} (\mathbf{x} - \mu_{-1})^T \Sigma_{-1}^{-1} (\mathbf{x} - \mu_{-1}) + \frac{1}{2} \ln |\Sigma_{-1}| - \ln p(y = -1) \\&= -\frac{1}{2} \mathbf{x}^T (\Sigma_1^{-1} - \Sigma_{-1}^{-1}) \mathbf{x} + \mathbf{x}^T (\Sigma_1^{-1} \mu_1 - \Sigma_{-1}^{-1} \mu_{-1}) - \frac{1}{2} \mu_1^T \Sigma_1^{-1} \mu_1 \\&\quad + \frac{1}{2} \mu_{-1}^T \Sigma_{-1}^{-1} \mu_{-1} - \frac{1}{2} \ln |\Sigma_1| + \frac{1}{2} \ln |\Sigma_{-1}| + \ln p(y = 1) - \ln p(y = -1) \\&= -\frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{w}^T \mathbf{x} + b.\end{aligned}$$

Maximum Likelihood

Quality criterion for our model:

- in case of supervised learning: generalization error
- in case of unsupervised learning: Maximum Likelihood

Unsupervised setting:

■ **Given:**

- data samples $\{\mathbf{x}\} = \{\mathbf{x}^1, \dots, \mathbf{x}^L\}$ (note: here $\mathbf{z}^i = \mathbf{x}^i$, i.e. no labels!)
- a parametrized model distribution $p(\mathbf{x}; \hat{\mathbf{w}})$ where $\hat{\mathbf{w}}$ are the parameters

■ **Task:** find the parameter \mathbf{w} that was most likely to produce this data.

■ **Idea:** How likely was a given $\hat{\mathbf{w}}$ to produce the dataset? Assuming that the \mathbf{x} are iid.:

$$\mathcal{L}(\{\mathbf{x}\}; \hat{\mathbf{w}}) = p(\{\mathbf{x}\}; \hat{\mathbf{w}}) = \prod_{i=1}^n p(\mathbf{x}^i; \hat{\mathbf{w}})$$

■ **Solution:** Find the \mathbf{w}^* that maximizes $\mathcal{L}(\{\mathbf{x}\}; \hat{\mathbf{w}})$

Maximum Likelihood

Find the \mathbf{w}^* that maximizes $\mathcal{L}(\{\mathbf{x}\}; \hat{\mathbf{w}})$:

$$\mathbf{w}^* = \arg \max_{\hat{\mathbf{w}}} \mathcal{L}(\{\mathbf{x}\}; \hat{\mathbf{w}}) = \arg \max_{\hat{\mathbf{w}}} \prod_{i=1}^n p(\mathbf{x}^i; \hat{\mathbf{w}})$$

It is better to optimize a sum instead of a product: use log!

$$\mathbf{w}^* = \arg \max_{\hat{\mathbf{w}}} \log \mathcal{L}(\{\mathbf{x}\}; \hat{\mathbf{w}}) = \arg \max_{\hat{\mathbf{w}}} \sum_{i=1}^n \log p(\mathbf{x}^i; \hat{\mathbf{w}})$$