Supervised Learning

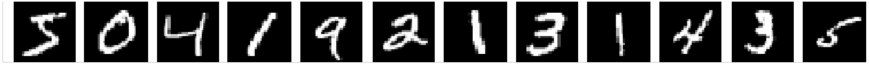
Johanni Brea

Introduction à l'apprentissage automatique

GYMINF 2021

Table of Contents

Handwritten Digit Classification (MNIST)



our goal: assign the correct digit class to images

5 0 4 1 9 2 1 3 1 4 3 5

input X: 28x28 = 784 pixels with values between 0 (black) and 1 (white)

output Y: digit class 0, 1, . . ., 9

Spam Detection with the Enron Dataset

spam

Subject: follow up
here ’ s a question i ’ ve been wanting to ask you , are you feeling down but too embarrassed to go to the doc to get your m / ed ’ s ?
here ’ s the answer , forget about your local p harm . acy and the long waits , visits and embarassments . . do it all in the privacy of your own home , right now . http : / / chopin . manilamana . com / p / test / duet it ’ s simply the best and most private way to obtain the stuff you need without all the red tape .

ham

Subject: darrin presto
amy :
please follow up as soon as possible with darrin presto regarding a real time interview . i forwarded his resume to you last week . he can be reached at 509 - 946 - 7879
thanks
greg

Our goal: classify new emails as spam or “ham“ (not spam).

input X: sequences of characters (emails), output Y: label spam or ham

Wind Speed Prediction

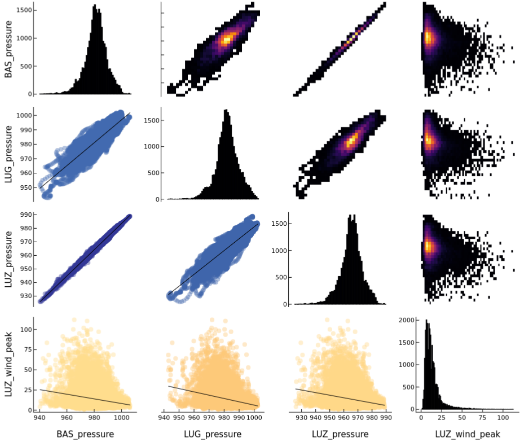
- SwissMeteo data: hourly measurements for 5 years from different stations (Bern, Basel, Luzern, Lugano, etc.).
- Our goal: given measurements at different stations, predict wind speed in Luzern 5 hours later.

Wind Speed Prediction

time	BAS_pressure	LUG_pressure	...	LUZ_pressure	LUZ_wind_peak
$x_{11} = 2015010100$	$x_{12} = 997.1$	$x_{13} = 998.6$	$\dots$	$x_{1p} = 980.0$	$y_1 = 13.0$
$x_{21} = 2015010101$	$x_{22} = 997.3$	$x_{23} = 998.8$	$\dots$	$x_{2p} = 979.9$	$y_2 = 6.8$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	
$x_{n1} = 2017123123$	$x_{n2} = 972.7$	$x_{n3} = 981.5$	$\dots$	$x_{np} = 957.5$	$y_n = 11.9$

- p **input** variables $X = (X_1, X_2, \dots, X_p)$
  e.g. X_1 `time`, X_2 `BAS_pressure`, X_3 `LUG_pressure`
  also called: **predictors, independent variables, features**
- **output** variable Y e.g. `LUZ_wind_peak`
  also called: **response, dependent variable**
- n **measurements** or **data points**

Always Look at Raw Data!

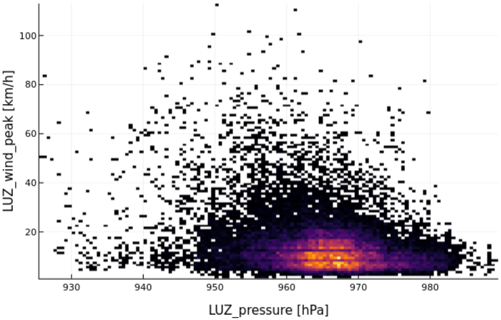


- **on diagonal**: 1D histogram
- **lower triangle**: scatter plot & trend line
- **upper triangle**: 2D histogram

Observations

1. LUZ_wind_peak has a long tail.
2. For low pressures there are outliers of strong wind.
3. Pressure in Basel and Luzern is highly correlated.
4. ...

Wind Speed Prediction



- The higher the pressure in Luzern, the less probable it is to have strong winds.
- There is no function `LUZ_wind_peak` = f(`LUZ_pressure`) that can describe this data; instead we use conditional probability densities p(`LUZ_wind_peak` | `LUZ_pressure`).

Table of Contents

Learning Objectives for this Lesson

- For a given data generating process you can define a supervised learning problem as a loss minimizing machine or a log-likelihood maximizing machine.
- For a given function f you can compute training and test losses.

Data Generating Processes

It is useful to think of our datasets as samples from **data generating processes** for the input X and the conditional output Y|X.

- **MNIST**
  X: people write digits $\rightarrow$ people take standardized photos thereof.
  Y|X: different people label the same photo X.
- **Spam**
  X: people write emails.
  Y|X: different people classify the same email X as spam or not.
- **Weather**
  X: the weather acts on sensors in weather stations.
  Y|X: the weather evolves from X and is measured again 5 hours later.

Using samples from these data generating processes, supervised learning aims at learning something about the conditional processes, i.e how Y depends on X.

Where Does Noise Come From?

For most data generating processes we **cannot measure all factors** that determine the outcome.

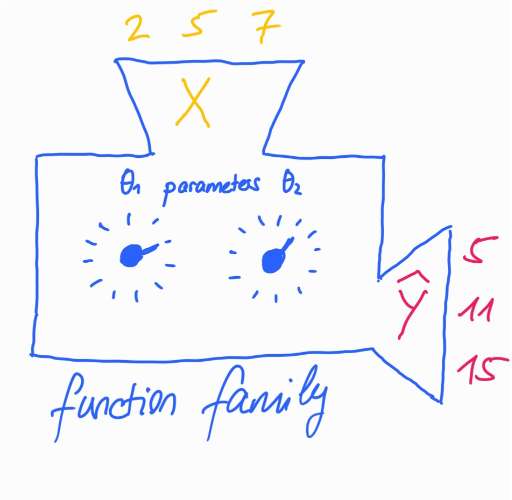
$\Rightarrow$ **same values of the measured factors can cause different outcomes.**

- **MNIST** Different persons may label the same handwritten digit differently.
- **Spam** What is spam for somebody, may not be spam for someone else.
- **Weather** Even when all considered weather stations measure exactly the same values at time t_1 and t_2, the full state of the weather at t_1 differs most likely from the one at t_2.

In machine learning we treat the effect of unmeasured factors as noise with certain probability distributions.

Table of Contents

How Does Supervised Learning Work?



Function Family

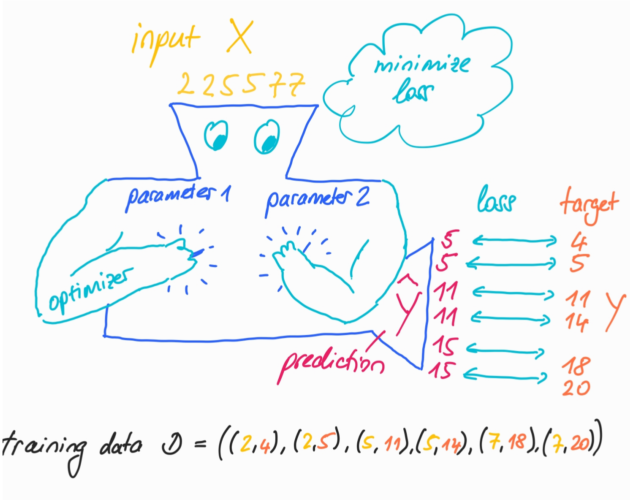
- We change the parameters.
- The machine computes $\hat{y}$ given parameters θ and x.

For example

$$\hat{y} = f_\theta(x) = \theta_0 + \theta_1 x$$

When we change the parameters θ_0 and θ_1, we change the way $\hat{y}$ depends on x.

How Does Supervised Learning Work?



Loss Minimizing Machine

- We specify
  1. the training data
  2. the function family (model)
  3. the loss function $L(y, \hat{y})$
  4. the optimizer
- The machine changes the parameters with the help of the optimizer until the loss is minimal.

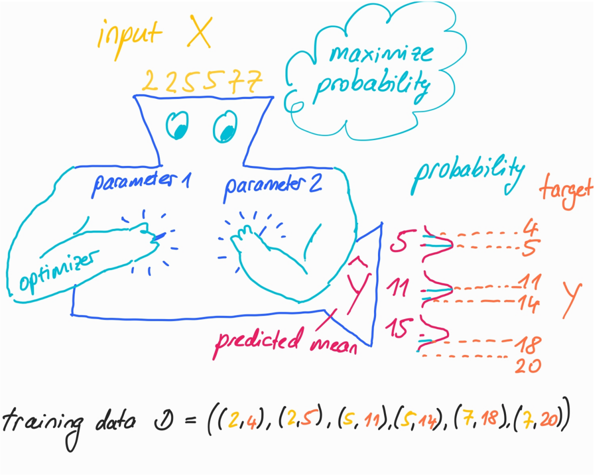
For example: linear regression

Training Loss and Test Loss

- **Training Set** $\mathcal{D}$: Data used by the machine to tune the parameters.
- **Training Loss of Function** f: $\mathcal{L}(f, \mathcal{D}) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i))$
- **Test Loss of Function** f **at x for a Conditional Data Generating Process**: $\mathrm{E}_{Y|x}\left[L(Y, f(x))\right]$ = expected loss under the conditional generating process.
- **Test Loss of Function** f **for a Joint Data Generating Process**: $\mathrm{E}_{X,Y|X}\left[L(Y, f(X))\right]$ = expected loss under the joint generating process.
- **Test Set** $\mathcal{D}_{\text{test}}$: Data from the same generating process as the training set, not used for parameter tuning.
- **Test Loss of Function** f **for a Test Set** $\mathcal{D}_{\text{test}}$: $\mathcal{L}(f, \mathcal{D}_{\text{test}})$ = same computation as for the training loss but for a test set.

Blackboard: Linear Regression as a Loss Minimizing Machine

How Does Supervised Learning Work?



Likelihood Maximizing Machine

- We specify
  1. the training data
  2. the family of probability distributions (model)
  3. the optimizer
- The machine changes the parameters with the help of the optimizer until the likelihood of the parameters is maximal.

For example: linear regression

The Likelihood Function

For a family of conditional probability distributions $P(y|x,\theta)$ and training data $\mathcal{D} = ((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n))$ the **likelihood function** is defined as

$$\ell(\theta) = \prod_{i=1}^{n} P(y_i|x_i, \theta)\,.$$

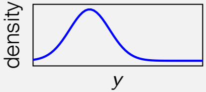
This is the probability of all the responses y_i given all the inputs x_i for a given value of the parameters θ.

In practice it is usually more convenient to work with the **log-likelihood function**

$$\log \ell(\theta) = \sum_{i=1}^{n} \log P(y_i|x_i, \theta)$$

The Normal, Bernoulli and Categorical Distribution

Normal

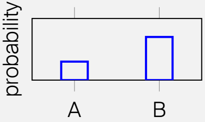


$$p(y|x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y-f(x))^2}{2\sigma^2}}$$

$f(x)$: a number

mean: $f(x)$
variance: σ^2
mode: $f(x)$

Bernoulli



$$p(\mathrm{A}|x) = p_\mathrm{A} = \sigma(f(x))$$

$f(x)$: a number

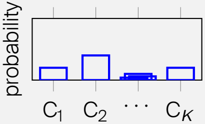
sigmoid/logistic function

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

$$p(\mathrm{B}|x) = 1 - p_\mathrm{A} = \sigma(-f(x))$$

rate of A: $\sigma(f(x))$
mode: A if $p_\mathrm{A} > p_\mathrm{B}$

Categorical



$$p(\mathrm{C}_i|x) = p_{\mathrm{C}_i} = s(f(x))_i$$

$f(x)$: a vector of K numbers

softmax function

$$s(x)_i = \frac{e^{x_i}}{\sum_{j=1}^{K} e^{x_j}}$$

mode: X with largest p_X.

Blackboard: Maximum Likelihood Estimation