Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Informatik
Computer Science

# Tutorial 6_Part2

## Project2: Classification

November 5-7, 2014

# Outline

- Review of classification concepts
- Data normalization
- Project 2

# Supervised learning big picture so far

Representation/ features

Linear hypotheses; nonlinear hypotheses using kernels

Model/ objective:

Loss-function + Regularization

Squared loss, 0/1 loss, Perceptron loss, Hinge loss, Multi-class hinge loss, Regret, Bayesian expected loss, …

$L^2$ norm, $L^1$ norm, …

Method:

Exact solution, Gradient Descent, SGD, Convex Programming, Sampling, Dynamic programming,…

Model selection:

Cross-Validation, Bayes factor, Minimum description length …

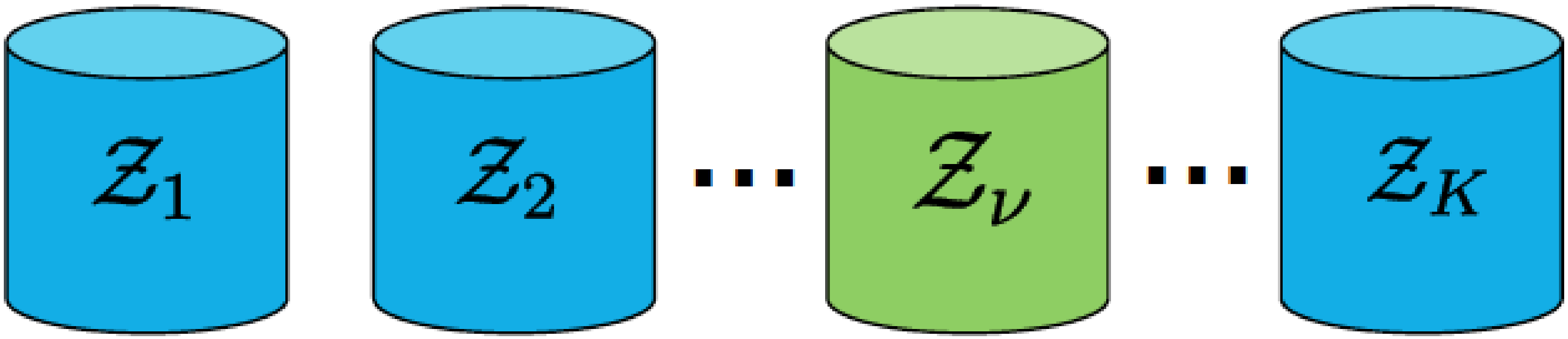
# Best practice for evaluating supervised learning

- Split data set into training and test set
- Optimize model on training set (e.g., by splitting it further using cross-validation)
- Report final accuracy on test set (but never optimize on test set)!
- Check for overfitting !

- Caveat: This only works if the data is i.i.d.
- Be careful, for example, if there are temporal trends or other dependencies

# K-fold cross-validation

Split data in $K$ approximately equally sized subsets, i.e.,
$\mathcal{Z} = \mathcal{Z}_1 \bigcup \mathcal{Z}_2 \bigcup \cdots \bigcup \mathcal{Z}_\nu \bigcup \cdots \bigcup \mathcal{Z}_K$;

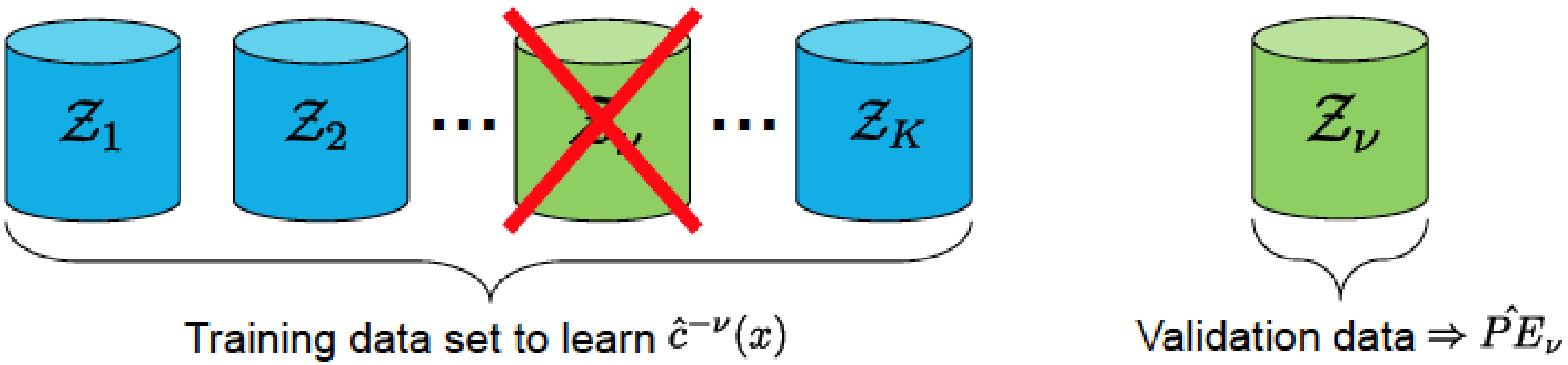



For every partition of a data set in $K$ subsets, we can define $K$ training data sets with approximately $n\frac{K-1}{K}$ data samples.

# K-fold cross-validation

## 2) $\nu$-th step

Adapt a model to the $K-1$ data subsets (learning step); validate the resulting model with the not yet used subset $\mathcal{Z}_\nu$




## 3) Estimation of the prediction error

# Cross-validation

- How large should we pick K?
- Too small
  - → Risk of overfitting to test set
  - → Using too little data for training → risk of underfitting to training set
- Too large
  - In general, better performance! K=n is perfectly fine (called leave-one-out cross-validation, LOOCV)
  - Higher computational complexity
- In practice, K=5 or K=10 is often used
- CV does not necessarily mean no overfitting

# Prediction error

- Prediction error is not the same as loss function!
- Prediction error is defined by the nature of the problem you tackle not by the model or method you use.
- Regression:
  - Mean squared sum of residuals (MSE – mean squared error)
  - Root mean squared sum of residuals (RMSE)
  - …
- Classification:
  - Accuracy
  - Asymmetrical classification error
  - Precision and recall
  - Mutual information
  - …

# Cross-validation for model selection

- Run cross-validation for every value of the hyperparameter ($\lambda$, $C$,…).
- Choose the value corresponding to the lowest prediction error.
- However (!) you can not report this error as the generalization error of your algorithm. Why?
- You have used all your data already for choosing your hyperparameter.
- So you need a separate dataset to evaluate the generalization error.

# MATLAB Demo

- Run 10 fold cross validation
- Compare with random classifier !

# Data Normalization

- Why normalize data ?
- Data comes from different sources – same type of data can have different range.
- E.g. magnetic resonance (MR) images from two different scanners

# Data Normalization

- MR images

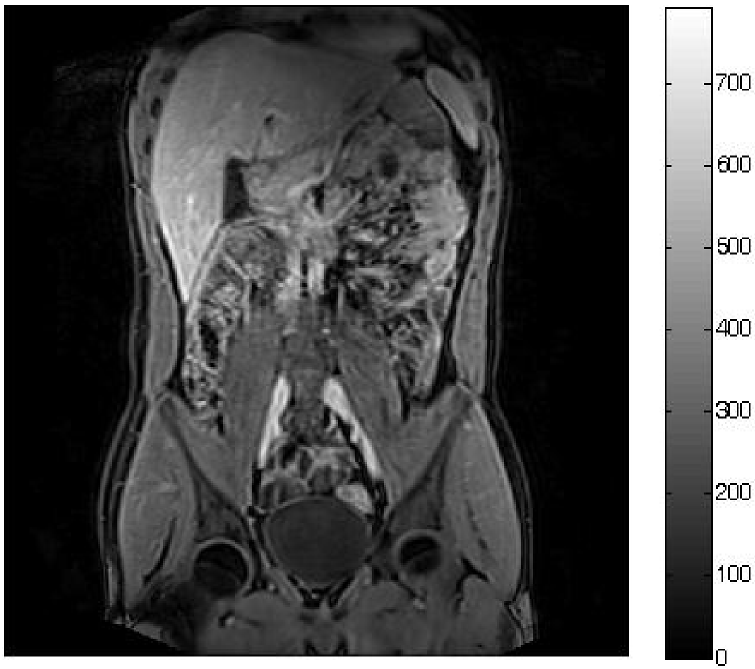



Image 1

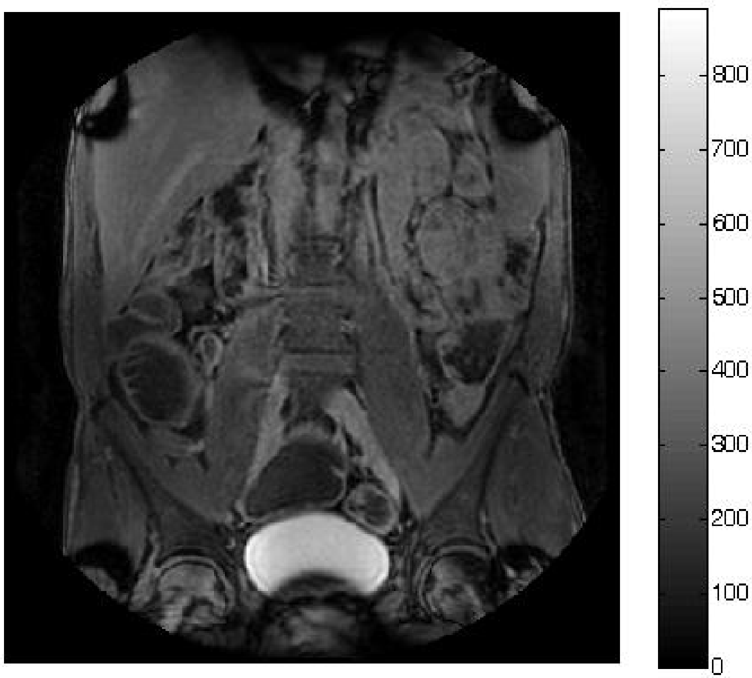


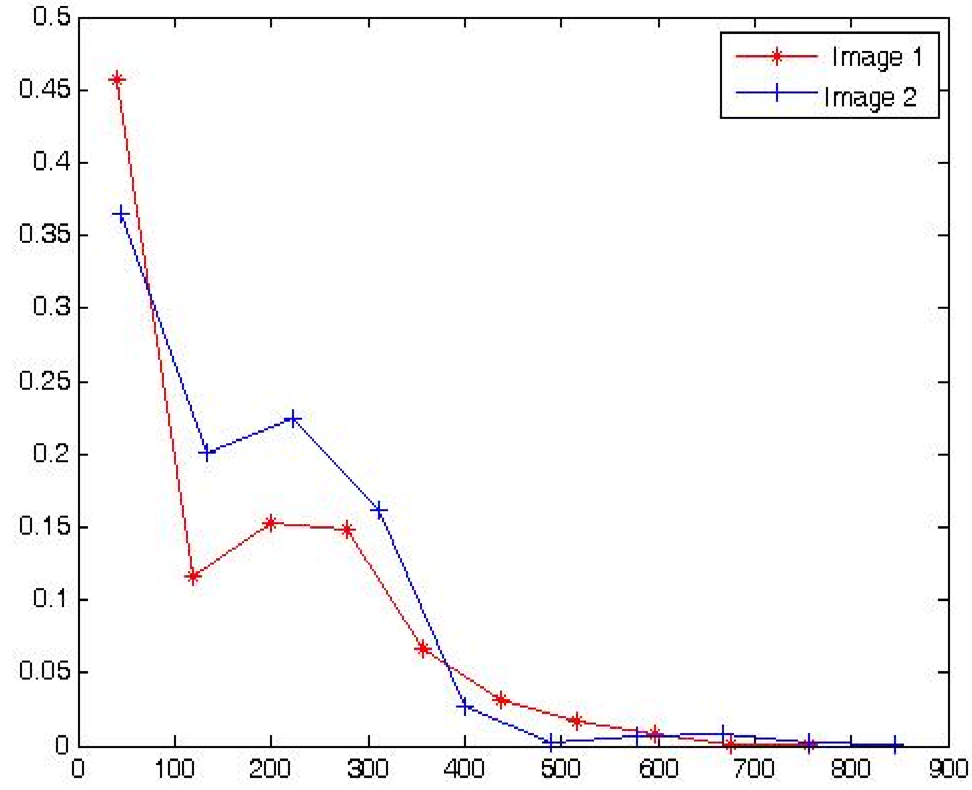

Image 2

# Data Normalization

- Intensities

| | Max | Min | Mean |
|---|---|---|---|
| Image 1 | 795 | 0 | 144 |
| Image 2 | 890 | 0 | 158 |

- But histograms are similar !

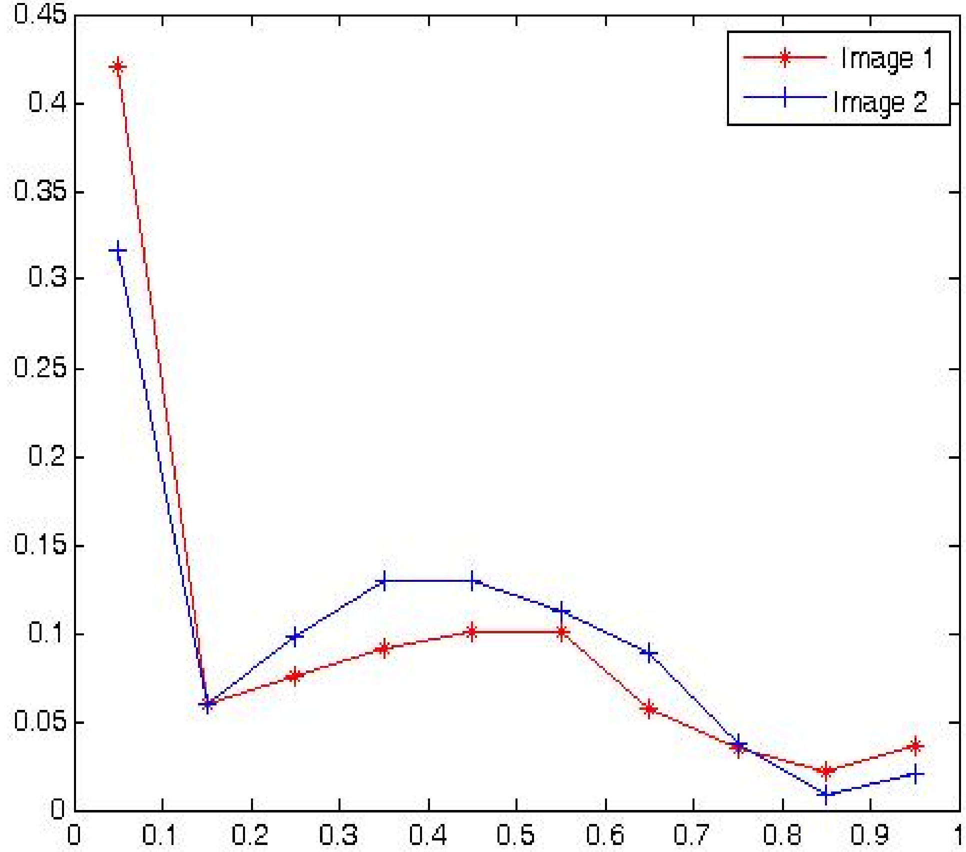



Normalized histogram of unnormalized images

# Data Normalization

- After normalization – max intensity – 1, min-0;

| | Max | Min | Mean |
|---|---|---|---|
| Image 1 | 1 | 0 | 0.28 |
| Image 2 | 1 | 0 | 0.31 |




Normalized histogram of normalized images

# Data normalization

- Without normalization classification goes wrong – same class may have different range of values

$$\tilde{x}_{i,j} = (x_{i,j} - \hat{\mu}_j)/\hat{\sigma}_j$$

- Normalization depends on type of data – medical images have isolated high signal intensities.
- Example.
- In most cases above formula will work.

# Project 2

- Classification of medical data for Crohns disease detection
- Feature vectors consist of 27 values – mean intensity, mean and variance of gradient over multiple neighborhoods.
- Images were normalized before feature extraction.
- Subsequent feature normalization

# Project 2

- Disease detection – screen patients for further tests
- False negatives (FN) –  normal patient is indicated diseased
- False positives (FP) – diseased patient goes undetected → undesirable.
- Penalize FPs more than FN→ asymmetrical error function

$$\text{CE} = \frac{5 \cdot |FP| + |FN|}{m}$$

# Project 2

- How to handle asymmetrical cost?
- Optimize training cost with asymmetric slack penalty

$$\min_{w} \mathbf{w}^t\mathbf{w} + C_+ \sum_{i\in +} \xi_+ + C_- \sum_{i\in -} \xi_- \quad s.t. y_i \mathbf{w}^t \mathbf{x}_i \geq 1 - \xi_i$$

- May use other classifiers