Medical image analysis: Refresher

Just pointers and reminders

Ender Konukoglu

ETH Zürich

February 18, 2020

Outline

- Basic notation
- Probabilistic modeling
- Optimization, cost function and regularization
- Linear basis models and function parameterizations
- Spatial transformations
- Derivative approximations

Basic notation

- **Continuous version**
  A volumetric grayscale image is a function
  $I : \Omega \rightarrow \mathbb{R}$, $\Omega \subset \mathbb{R}^3$ is the image domain
  $I(x)$ is the intensity at point $x \in \Omega$, $x = [x_1, x_2, x_3]$

Basic notation

- **Continuous version**
  A volumetric grayscale image is a function
  $I : \Omega \to \mathbb{R}$, $\Omega \subset \mathbb{R}^3$ is the image domain
  $I(x)$ is the intensity at point $x \in \Omega$, $x = [x_1, x_2, x_3]$
- **Discrete version**
  $\Omega \subset \mathbb{Z}^3$ is the discrete image domain, i.e. Cartesian grid
  $I(x)$ is the intensity at $x \in \Omega$, $x = [i, j, k]$

Basic notation

- **Continuous version**
  A volumetric grayscale image is a function
  $I : \Omega \to \mathbb{R}$, $\Omega \subset \mathbb{R}^3$ is the image domain
  $I(x)$ is the intensity at point $x \in \Omega$, $x = [x_1, x_2, x_3]$
- **Discrete version**
  $\Omega \subset \mathbb{Z}^3$ is the discrete image domain, i.e. Cartesian grid
  $I(x)$ is the intensity at $x \in \Omega$, $x = [i, j, k]$
- For multiparametric images, such as diffusion-weighted MRI, spectroscopy or adding multiple modalities together
  $I : \Omega \to \mathbb{R}^N$
  $I(x)$ is the vector of intensities at x

Basic notation

- **Continuous version**
  A volumetric grayscale image is a function
  $I : \Omega \to \mathbb{R}$, $\Omega \subset \mathbb{R}^3$ is the image domain
  $I(x)$ is the intensity at point $x \in \Omega$, $x = [x_1, x_2, x_3]$
- **Discrete version**
  $\Omega \subset \mathbb{Z}^3$ is the discrete image domain, i.e. Cartesian grid
  $I(x)$ is the intensity at $x \in \Omega$, $x = [i, j, k]$
- For multiparametric images, such as diffusion-weighted MRI, spectroscopy or adding multiple modalities together
  $I : \Omega \to \mathbb{R}^N$
  $I(x)$ is the vector of intensities at x
- For dynamic images with temporal information
  $I : \Omega x \mathbb{R}^+ \to \mathbb{R}^N$
  $I(x, t)$ is the intensity at point x and time t

Outline

- Basic notation
- Probabilistic modeling
  - PDF, CDF, PMF
  - Histogram of an image
  - Conditionals and the Bayes' Rule
  - Posterior distribution, MAP and MLE
- Optimization, cost function and regularization
- Linear basis models and function parameterizations
- Spatial transformations
- Derivative approximations

PDF, CDF, PMF

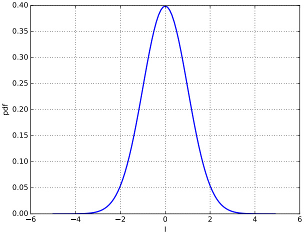
- Intensity at each point $I(x) \in \mathbb{R}$ is seen as a random variable
  (Other functions can also be seen as random variables, e.g. transformation $T(x)$ or discrete labels $L(x)$)

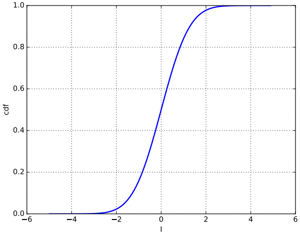
PDF, CDF, PMF

- Intensity at each point $I(x) \in \mathbb{R}$ is seen as a random variable
  (Other functions can also be seen as random variables, e.g. transformation $T(x)$ or discrete labels $L(x)$)
- For continuous random variables we define (dropping (x) for simplicity)
  $p(i)$ as its *probability density function (PDF)*
  $P(i) = Pr[I \leq i] = \int_{-\infty}^{i} p(j)dj$ as its *cumulative distribution function (CDF)*

PDF, CDF, PMF

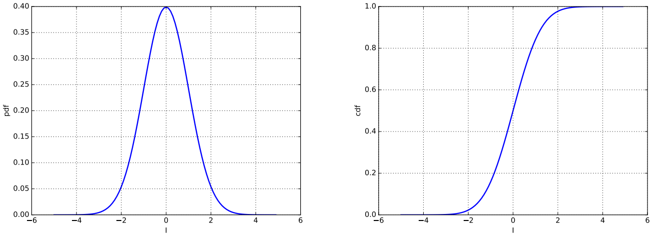
- Intensity at each point $I(x) \in \mathbb{R}$ is seen as a random variable
  (Other functions can also be seen as random variables, e.g. transformation $T(x)$ or discrete labels $L(x)$)
- For continuous random variables we define (dropping (x) for simplicity)
  $p(i)$ as its *probability density function (PDF)*
  $P(i) = Pr[I \leq i] = \int_{-\infty}^{i} p(j)dj$ as its *cumulative distribution function (CDF)*





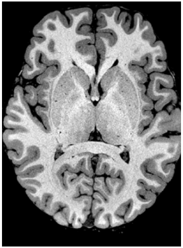
PDF, CDF, PMF

- Intensity at each point $I(x) \in \mathbb{R}$ is seen as a random variable
(Other functions can also be seen as random variables, e.g. transformation $T(x)$ or discrete labels $L(x)$)
- For continuous random variables we define (dropping (x) for simplicity)
$p(i)$ as its *probability density function (PDF)*
$P(i) = Pr[I \leq i] = \int_{-\infty}^{i} p(j)dj$ as its *cumulative distribution function (CDF)*

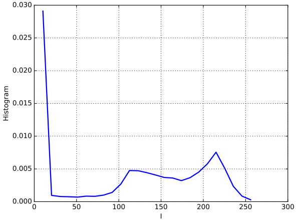


- For discrete random variables, L, we define
$p(l) = p(L = l)$ as its *probability mass function (PMF)*
PMF can be seen as a PDF is often named as PDF in scientific articles

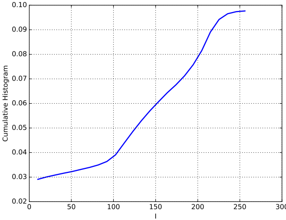
Histogram of an image



Image



Histogram



Cumulative histogram

If we consider each pixel intensity as an independent realization of the random variable I then the histogram is an approximation to the PDF and cumulative histogram is an approximation to the CDF

Conditionals and the Bayes' Rule

- When we have two variables such as I and L
  $p(i,l)$: joint distribution
  $p(i) = \sum_{l=0}^{N} p(i,l)$ and $p(l) = \int_{-\infty}^{\infty} p(i,l)di$: marginal distributions
  $p(i|l)$ and $p(l|i)$: conditional distributions
  $p(i,l) = p(i|l)p(l) = p(l|i)p(i)$

Conditionals and the Bayes' Rule

- When we have two variables such as I and L
  $p(i, l)$: joint distribution
  $p(i) = \sum_{l=0}^{N} p(i, l)$ and $p(l) = \int_{-\infty}^{\infty} p(i, l) di$: marginal distributions
  $p(i|l)$ and $p(l|i)$: conditional distributions
  $p(i, l) = p(i|l)p(l) = p(l|i)p(i)$
- The conditionals are linked with the Bayes' rule

$$p(i|l) = \frac{p(l|i)p(i)}{p(l)} \text{ and } p(l|i) = \frac{p(i|l)p(l)}{p(i)}$$

Conditionals and the Bayes' Rule

- When we have two variables such as I and L
  $p(i,l)$: joint distribution
  $p(i) = \sum_{l=0}^{N} p(i,l)$ and $p(l) = \int_{-\infty}^{\infty} p(i,l)di$: marginal distributions
  $p(i|l)$ and $p(l|i)$: conditional distributions
  $p(i,l) = p(i|l)p(l) = p(l|i)p(i)$
- The conditionals are linked with the Bayes' rule

$$p(i|l) = \frac{p(l|i)p(i)}{p(l)} \text{ and } p(l|i) = \frac{p(i|l)p(l)}{p(i)}$$

- In a large variety of problems one of them is observed and the other not. Assume i is observed in that case
  $p(i|l)$: likelihood
  $p(l)$: prior distribution
  $p(l|i)$: posterior distribution

Posterior distribution, MAP and MLE

- A large range of problems can be formulated as given an observation i estimate I.
- Example problems: image enhancement, segmentation and even registration.

Posterior distribution, MAP and MLE

- A large range of problems can be formulated as given an observation i estimate I.
- Example problems: image enhancement, segmentation and even registration.
- A generic solution to such problems is to determine the posterior distribution of I, i.e. *Bayesian Inference*

$$p(I|i) = \frac{p(i|I)p(I)}{p(i)}$$

Posterior distribution, MAP and MLE

- A large range of problems can be formulated as given an observation i estimate I.
- Example problems: image enhancement, segmentation and even registration.
- A generic solution to such problems is to determine the posterior distribution of I, i.e. *Bayesian Inference*

$$p(I|i) = \frac{p(i|I)p(I)}{p(i)}$$

- $p(i)$ requires summing over all I (or integration over all I if continuous), which can be infeasible. The alternative is to determine the I that maximizes the posterior, i.e. *Maximum-A-Posteriori (MAP)* estimate

$$\arg_I \max p(I|i) = \arg_I \max p(i|I)p(I) = \arg_I \max \log p(i|I) + \log p(I)$$

Posterior distribution, MAP and MLE

- A large range of problems can be formulated as given an observation i estimate I.
- Example problems: image enhancement, segmentation and even registration.
- A generic solution to such problems is to determine the posterior distribution of I, i.e. *Bayesian Inference*

$$p(I|i) = \frac{p(i|I)p(I)}{p(i)}$$

- $p(i)$ requires summing over all I (or integration over all I if continuous), which can be infeasible. The alternative is to determine the I that maximizes the posterior, i.e. *Maximum-A-Posteriori (MAP)* estimate

$$\arg_I \max p(I|i) = \arg_I \max p(i|I)p(I) = \arg_I \max \log p(i|I) + \log p(I)$$

- Posterior distribution and MAP estimate requires the prior. When such a prior does not exist the alternative is to determine *Maximum Likelihood Estimate (MLE)*

$$\arg_I \max p(i|I)$$

Posterior distribution, MAP and MLE

- A large range of problems can be formulated as given an observation i estimate I.
- Example problems: image enhancement, segmentation and even registration.
- A generic solution to such problems is to determine the posterior distribution of I, i.e. *Bayesian Inference*

$$p(I|i) = \frac{p(i|I)p(I)}{p(i)}$$

- $p(i)$ requires summing over all I (or integration over all I if continuous), which can be infeasible. The alternative is to determine the I that maximizes the posterior, i.e. *Maximum-A-Posteriori (MAP)* estimate

$$\arg_I \max p(I|i) = \arg_I \max p(i|I)p(I) = \arg_I \max \log p(i|I) + \log p(I)$$

- Posterior distribution and MAP estimate requires the prior. When such a prior does not exist the alternative is to determine *Maximum Likelihood Estimate (MLE)*

$$\arg_I \max p(i|I)$$

- MLE is the same as MAP when the prior is uniform, i.e. $p(I) = c, \ \forall I$

Outline

- Basic notation
- Probabilistic modeling
- Optimization, cost function and regularization
  - Data term and regularization
  - Optimization
  - Calculus of variation
- Linear basis models and function parameterizations
- Spatial transformations
- Derivative approximations

Data term and regularization

- Most problems in medical image analysis are formulated as optimization problems

$$\arg_{\theta}\min \underbrace{\mathcal{L}(\theta)}_{\text{Cost Function}} = \underbrace{\mathcal{D}(I;\theta)}_{\text{Data Term}} + \lambda \underbrace{\mathcal{R}(\theta)}_{\text{Regularization}}$$

Data term and regularization

- Most problems in medical image analysis are formulated as optimization problems

$$\arg_\theta \min \underbrace{\mathcal{L}(\theta)}_{\text{Cost Function}} = \underbrace{\mathcal{D}(I;\theta)}_{\text{Data Term}} + \lambda \underbrace{\mathcal{R}(\theta)}_{\text{Regularization}}$$

- Or the related form with explicit constraints

$$\arg_\theta \min \mathcal{D}(I;\theta), \text{ such that, } \mathcal{R}(\theta) = 0$$

Data term and regularization

- Most problems in medical image analysis are formulated as optimization problems

$$\arg_\theta \min \underbrace{\mathcal{L}(\theta)}_{\text{Cost Function}} = \underbrace{\mathcal{D}(I;\theta)}_{\text{Data Term}} + \lambda \underbrace{\mathcal{R}(\theta)}_{\text{Regularization}}$$

- Or the related form with explicit constraints

$$\arg_\theta \min \mathcal{D}(I;\theta), \text{ such that, } \mathcal{R}(\theta) = 0$$

- Examples:

Data term and regularization

- Most problems in medical image analysis are formulated as optimization problems

$$\arg_\theta \min \underbrace{\mathcal{L}(\theta)}_{\text{Cost Function}} = \underbrace{\mathcal{D}(I;\theta)}_{\text{Data Term}} + \lambda \underbrace{\mathcal{R}(\theta)}_{\text{Regularization}}$$

- Or the related form with explicit constraints

$$\arg_\theta \min \mathcal{D}(I;\theta), \text{ such that, } \mathcal{R}(\theta) = 0$$

- Examples:
  - Image denoising: Assume I is a noisy image and we want to retrieve a denoised J

$$\arg_J \min \|I - J\|_2^2 + \lambda \|\nabla J(x)\|_1 = \arg_J \min \int (I(x) - J(x))^2 \, dx + \lambda \int |\nabla J(x)| dx$$

Data term and regularization

- Most problems in medical image analysis are formulated as optimization problems

$$\arg_\theta \min \underbrace{\mathcal{L}(\theta)}_{\text{Cost Function}} = \underbrace{\mathcal{D}(I;\theta)}_{\text{Data Term}} + \lambda \underbrace{\mathcal{R}(\theta)}_{\text{Regularization}}$$

- Or the related form with explicit constraints

$$\arg_\theta \min \mathcal{D}(I;\theta), \text{ such that, } \mathcal{R}(\theta) = 0$$

- Examples:
  - Image denoising: Assume I is a noisy image and we want to retrieve a denoised J

$$\arg_J \min \|I - J\|_2^2 + \lambda \|\nabla J(x)\|_1 = \arg_J \min \int (I(x) - J(x))^2 \, dx + \lambda \int |\nabla J(x)| dx$$

  - Image registration: Determine T between images I and J

$$\arg_T \min \int (I(x) - J(T(x)))^2 dx + \int \|\alpha \Delta T(x) + \beta \nabla(\nabla \cdot T(x)) - \gamma T(x)\|_2^2 dx$$

Data term and regularization

- Most problems in medical image analysis are formulated as optimization problems

$$\arg_\theta \min \underbrace{\mathcal{L}(\theta)}_{\text{Cost Function}} = \underbrace{\mathcal{D}(I;\theta)}_{\text{Data Term}} + \lambda \underbrace{\mathcal{R}(\theta)}_{\text{Regularization}}$$

- Or the related form with explicit constraints

$$\arg_\theta \min \mathcal{D}(I;\theta), \text{ such that, } \mathcal{R}(\theta) = 0$$

- Examples:
  - Image denoising: Assume I is a noisy image and we want to retrieve a denoised J

$$\arg_J \min \|I - J\|_2^2 + \lambda \|\nabla J(x)\|_1 = \arg_J \min \int (I(x) - J(x))^2 \, dx + \lambda \int |\nabla J(x)| dx$$

  - Image registration: Determine T between images I and J

$$\arg_T \min \int (I(x) - J(T(x)))^2 dx + \int \|\alpha \Delta T(x) + \beta \nabla(\nabla \cdot T(x)) - \gamma T(x)\|_2^2 dx$$

- The MAP estimate is also in the same form

$$\arg_\theta \max \log p(i|\theta) + \log p(\theta) = \arg_\theta \min - \log p(i|\theta) - \log p(\theta)$$

Regularizers can be thought of as priors with $-\log p(\theta) \propto \mathcal{R}(\theta)$

Optimization

- Basic quantities for discrete θ:

Optimization

- Basic quantities for discrete θ:
  - Gradient

$$\nabla\mathcal{L}(\theta) = \left[\frac{\partial\mathcal{L}(\theta)}{\partial\theta_1}, \ldots, \frac{\partial\mathcal{L}(\theta)}{\partial\theta_d}\right]^T$$

Optimization

- Basic quantities for discrete θ:
  - Gradient

$$\nabla\mathcal{L}(\theta) = \left[\frac{\partial\mathcal{L}(\theta)}{\partial\theta_1}, \ldots, \frac{\partial\mathcal{L}(\theta)}{\partial\theta_d}\right]^T$$

  - Hessian

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_1^2} & \cdots & \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_1\partial\theta_d} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_1\partial\theta_d} & \cdots & \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_d^2} \end{bmatrix}$$

Optimization

- Basic quantities for discrete θ:
  - Gradient

$$\nabla\mathcal{L}(\theta) = \left[\frac{\partial\mathcal{L}(\theta)}{\partial\theta_1}, \ldots, \frac{\partial\mathcal{L}(\theta)}{\partial\theta_d}\right]^T$$

  - Hessian

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_1^2} & \cdots & \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_1\partial\theta_d} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_1\partial\theta_d} & \cdots & \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_d^2} \end{bmatrix}$$

- Gradient-based algorithms
  - Gradient descent / ascent
  - Newton's method
  - Limited-memory BFGS, ...

Optimization

- Basic quantities for discrete θ:
  - Gradient

$$\nabla\mathcal{L}(\theta) = \left[\frac{\partial\mathcal{L}(\theta)}{\partial\theta_1}, \dots, \frac{\partial\mathcal{L}(\theta)}{\partial\theta_d}\right]^T$$

  - Hessian

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_1^2} & \dots & \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_1\partial\theta_d} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_1\partial\theta_d} & \dots & \frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_d^2} \end{bmatrix}$$

- Gradient-based algorithms
  - Gradient descent / ascent
  - Newton's method
  - Limited-memory BFGS, ...
- Gradient-free algorithms - because sometimes gradient is difficult to evaluate
  - Nelder-Mead Simplex
  - Simulated Annealing
  - Powell's method, BOBYQA, ...

Optimization

- Basic quantities for discrete θ:
  - Gradient

$$\nabla \mathcal{L}(\theta) = \left[\frac{\partial \mathcal{L}(\theta)}{\partial \theta_1}, \ldots, \frac{\partial \mathcal{L}(\theta)}{\partial \theta_d} \right]^T$$

  - Hessian

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 \mathcal{L}(\theta)}{\partial \theta_1^2} & \cdots & \frac{\partial^2 \mathcal{L}(\theta)}{\partial \theta_1 \partial \theta_d} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 \mathcal{L}(\theta)}{\partial \theta_1 \partial \theta_d} & \cdots & \frac{\partial^2 \mathcal{L}(\theta)}{\partial \theta_d^2} \end{bmatrix}$$

- Gradient-based algorithms
  - Gradient descent / ascent
  - Newton's method
  - Limited-memory BFGS, ...
- Gradient-free algorithms - because sometimes gradient is difficult to evaluate
  - Nelder-Mead Simplex
  - Simulated Annealing
  - Powell's method, BOBYQA, ...
- Some references
  - Convex Optimization, Boyd and Vandenberghe, Cambridge University Press, 2004.
  - Nonlinear Programming, Bertsekas, Athena Scientific, 2016.

Calculus of Variation

- When θ is continuous, i.e. deformation field, denoised version of the image,...

Calculus of Variation

- When θ is continuous, i.e. deformation field, denoised version of the image,...
- Cost function is called the functional (this looks like some of the examples)

$$\mathcal{L}(\theta) = \int_{x_a}^{x_b} L(x, \theta, \nabla\theta) dx$$

Calculus of Variation

- When θ is continuous, i.e. deformation field, denoised version of the image,...
- Cost function is called the functional (this looks like some of the examples)

$$\mathcal{L}(\theta) = \int_{x_a}^{x_b} L(x, \theta, \nabla\theta) dx$$

- The identity similar to gradient is the first variation

$$\delta\mathcal{L}(\theta) \triangleq \lim_{\epsilon \to 0} \frac{L(\theta + \epsilon\eta) - L(\theta)}{\epsilon}$$

for arbitrary η whose value vanishes at the boundaries.

Calculus of Variation

- When θ is continuous, i.e. deformation field, denoised version of the image,...
- Cost function is called the functional (this looks like some of the examples)

$$\mathcal{L}(\theta) = \int_{x_a}^{x_b} L(x, \theta, \nabla\theta) dx$$

- The identity similar to gradient is the first variation

$$\delta\mathcal{L}(\theta) \triangleq \lim_{\epsilon \to 0} \frac{L(\theta + \epsilon\eta) - L(\theta)}{\epsilon}$$

  for arbitrary η whose value vanishes at the boundaries.
- Setting first variation to 0 for all η gives the Euler-Lagrange equation

$$\frac{\partial L}{\partial \theta} - \sum_{k=1}^{d} \frac{d}{dx_k} \frac{\partial L}{\partial \theta^{(k)}} = 0, \;\; \theta^{(k)} = \frac{\partial \theta}{\partial x_k}$$

Calculus of Variation

- When θ is continuous, i.e. deformation field, denoised version of the image,...
- Cost function is called the functional (this looks like some of the examples)

$$\mathcal{L}(\theta) = \int_{x_a}^{x_b} L(x, \theta, \nabla\theta) dx$$

- The identity similar to gradient is the first variation

$$\delta\mathcal{L}(\theta) \triangleq \lim_{\epsilon \to 0} \frac{L(\theta + \epsilon\eta) - L(\theta)}{\epsilon}$$

for arbitrary η whose value vanishes at the boundaries.

- Setting first variation to 0 for all η gives the Euler-Lagrange equation

$$\frac{\partial L}{\partial \theta} - \sum_{k=1}^{d} \frac{d}{dx_k} \frac{\partial L}{\partial \theta^{(k)}} = 0, \;\; \theta^{(k)} = \frac{\partial \theta}{\partial x_k}$$

- Left handside is called functional derivative $\delta\mathcal{L}/\delta\theta$. Often used in gradient ascent.

Calculus of Variation

- When θ is continuous, i.e. deformation field, denoised version of the image,...
- Cost function is called the functional (this looks like some of the examples)

$$\mathcal{L}(\theta) = \int_{x_a}^{x_b} L(x, \theta, \nabla\theta) dx$$

- The identity similar to gradient is the first variation

$$\delta\mathcal{L}(\theta) \triangleq \lim_{\epsilon \to 0} \frac{L(\theta + \epsilon\eta) - L(\theta)}{\epsilon}$$

  for arbitrary η whose value vanishes at the boundaries.
- Setting first variation to 0 for all η gives the Euler-Lagrange equation

$$\frac{\partial L}{\partial \theta} - \sum_{k=1}^{d} \frac{d}{dx_k} \frac{\partial L}{\partial \theta^{(k)}} = 0, \ \theta^{(k)} = \frac{\partial \theta}{\partial x_k}$$

- Left handside is called functional derivative $\delta\mathcal{L}/\delta\theta$. Often used in gradient ascent.
- Reference: The Calculus of Variations, Bruce van Brunt, Springer, 2004.

Outline

- Basic notation
- Probabilistic modeling
- Optimization, cost function and regularization
- Linear basis models and function parameterizations
    - Basics
    - Function parameterizations with linear basis models
    - Kernel-based parameterization
- Spatial transformations
- Derivative approximations

Basics for parameterization

- Many problems need optimizing over a function, e.g. non-linear registration - transformations, bias correction - bias field

Basics for parameterization

- Many problems need optimizing over a function, e.g. non-linear registration - transformations, bias correction - bias field
- Discretizing Euler-Lagrange at every point is an option but something with less parameters would be very useful

Basics for parameterization

- Many problems need optimizing over a function, e.g. non-linear registration - transformations, bias correction - bias field
- Discretizing Euler-Lagrange at every point is an option but something with less parameters would be very useful
- Function parameterization for optimization, few parameters to describe the function

Basics for parameterization

- Many problems need optimizing over a function, e.g. non-linear registration - transformations, bias correction - bias field
- Discretizing Euler-Lagrange at every point is an option but something with less parameters would be very useful
- Function parameterization for optimization, few parameters to describe the function
- In finite vector spaces

$$\vec{v} = a_1 \vec{b}_1 + a_2 \vec{b}_2 + \cdots + a_d \vec{b}_d = \mathbf{B}\vec{a}$$

$\vec{b}_i$ are the basis functions and a_i coefficients.
if $\vec{b}_i$ are orthogonal, i.e. $\vec{b}_i^T \vec{b}_j = 0, \ \forall i \neq j$ then $a_i = \vec{v}^T \vec{b}_i / \|\vec{b}_i\|_2$.
if not then Ordinary Least Square (OLS) regression must be performed

$$\arg_{\vec{a}} \min \|\vec{v} - \mathbf{B}\vec{a}\|_2^2 = \left(\mathbf{B}^T \mathbf{B}\right)^{-1} \mathbf{B}^T \vec{v}$$

For known $\vec{b}_i$, $\vec{a}$ can be a parameterization for $\vec{v}$

Function parameterizations with linear basis models

Global parameterizations

$$f(x) = \sum_{i}^{d} a_i b_i(x)$$

where $b_i(x)$ are smooth basis functions.

- The parameterization is $\vec{a}$
- To determine $\vec{a}$, the space is discretized and a **B** matrix is formed.
- Note that this parameterization cannot represent all functions.
- Examples:
  - polynomials: bias-field correction in MRI
  - splines: non-linear registration

Kernel-based parameterization

$$f(x) = \sum_{i=1}^{d} a_i K(x, x_i)$$

where x_i are the control points and $K(x, x_i)$ is a kernel function

- Radial basis functions are often used: $K(x, x_i) = K(\|x - x_i\|_2)$
- Popular radial basis functions:
  Gaussian
  $$K(\|x - x_i\|_2) = \exp - \frac{\|x - x_i\|_2^2}{\sigma^2}$$
  Thin-plate spline
  $$K(\|x - x_i\|_2) = \|x - x_i\|_2^2 \ln(\|x - x_i\|_2)$$
- Example: landmark based registration, linear/non-linear registration, kernel density estimation

Outline

- Basic notation
- Probabilistic modeling
- Optimization, cost function and regularization
- Linear basis models and function parameterizations
- Spatial transformations
  - Linear transformations
  - Non-linear transformations
  - Transformation related identities
- Derivative approximations

Linear transformations

$$T(\vec{x}) = \mathbf{T}\vec{x},$$

Linear transformations

$$T(\vec{x}) = \mathbf{T}\vec{x},$$

- Used in linear image registration

Linear transformations

$$T(\vec{x}) = \mathbf{T}\vec{x},$$

- Used in linear image registration
- Rigid
  - 3 dof in 2D - 2 translation and 1 rotation
  - 6 dof in 3D - 3 translation and 3 rotation
  - Intra-subject registration, multi-modal intra-subject registration

Linear transformations

$$T(\vec{x}) = \mathbf{T}\vec{x},$$

- Used in linear image registration
- Rigid
  - 3 dof in 2D - 2 translation and 1 rotation
  - 6 dof in 3D - 3 translation and 3 rotation
  - Intra-subject registration, multi-modal intra-subject registration
- Similarity transformation
  - 4 dof in 2D - Rigid + scale
  - 7 dof in 3D - Rigid + scale
  - Used for coarse alignment in inter-subject. Initialization for non-linear

Linear transformations

$$T(\vec{x}) = \mathbf{T}\vec{x},$$

- Used in linear image registration
- Rigid
  - 3 dof in 2D - 2 translation and 1 rotation
  - 6 dof in 3D - 3 translation and 3 rotation
  - Intra-subject registration, multi-modal intra-subject registration
- Similarity transformation
  - 4 dof in 2D - Rigid + scale
  - 7 dof in 3D - Rigid + scale
  - Used for coarse alignment in inter-subject. Initialization for non-linear
- Affine transformation
  - 6 dof in 2D
  - 12 dof in 3D
  - Often not used in full dof
  - Instead, 9 dof transformation (3 translation + 3 rotation + 3 scale) may be preferred.

$$\mathbf{T} = \underbrace{\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix}}_{\text{scale}} \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\theta) & -sin(\theta) \\ 0 & sin(\theta) & cos(\theta) \end{bmatrix}}_{\text{rotation around x-axis}} \underbrace{\begin{bmatrix} cos(\phi) & 0 & sin(\phi) \\ 0 & 1 & 0 \\ -sin(\phi) & 0 & cos(\theta) \end{bmatrix}}_{\text{rotation around y-axis}} \underbrace{\begin{bmatrix} cos(\gamma) & sin(\gamma) & 0 \\ -sin(\gamma) & cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{rotation around z-axis}} + \underbrace{\begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}}_{\text{translation}}$$

Non-linear transformations

$$T(\vec{x}) = \vec{x} + \underbrace{\vec{u}(\vec{x})}_{\text{displacement field}}$$

- Used in non-linear registration.
- Different models are used.
- Linear basis function models - additive
- More advanced techniques based on composition of transformations

$$T(\vec{x}) = T_n \circ T_{n-1} \circ \cdots \circ T_1(\vec{x})$$

where $\circ$ is basic function composition
Allows modeling diffeomorphisms
see Computational Anatomy

Transformation related identities

$$T(\vec{x}) = [T_1(\vec{x}), T_2(\vec{x}), T_3(\vec{x})]^T$$

- Jacobian

$$\mathbf{J} = \begin{bmatrix} \frac{\partial T_1}{\partial x_1} & \frac{\partial T_1}{\partial x_2} & \frac{\partial T_1}{\partial x_3} \\ \frac{\partial T_2}{\partial x_1} & \frac{\partial T_2}{\partial x_2} & \frac{\partial T_2}{\partial x_3} \\ \frac{\partial T_3}{\partial x_1} & \frac{\partial T_3}{\partial x_2} & \frac{\partial T_3}{\partial x_3} \end{bmatrix}$$

  Jacobian determinant quantifies local volumetric change.
  $\det(\mathbf{J}) = 1$: no change, $\det(\mathbf{J}) < 1$: compression, $\det(\mathbf{J}) > 1$: expansion
- Divergence: $\nabla \cdot T$ - trace of the Jacobian.
  Gives information about the amount of compression and expansion as well.
- Curl - $\nabla \times T$ - information on the amount of infinitesimal rotation

Outline

- Basic notation
- Probabilistic modeling
- Optimization, cost function and regularization
- Linear basis models and function parameterizations
- Spatial transformations
- Derivative approximations

Derivative approximations

- Finite difference approximations is used the most often
- First order derivative at a grid point x_0 with grid spacing Δx

$$\begin{aligned}\frac{df}{dx}\Big|_{x_0} &\approx \frac{f(x_0+\Delta x)-f(x_0)}{\Delta x}\\ &\approx \frac{f(x_0)-f(x_0-\Delta x)}{\Delta x}\\ &\approx \frac{f(x_0+\Delta x)-f(x_0-\Delta x)}{2\Delta x}\end{aligned}$$

- Second order derivative at a grid point x_0 with grid spacing Δx

$$\frac{d^2 f}{dx^2}\Big|_{x_0} \approx \frac{f(x_0+\Delta x)-2f(x_0)+f(x_0-\Delta x)}{\Delta x^2}$$

- Many different approximations exist