Master in Computer Vision Barcelona

UAB UOC UPC upf.

Module: M2. Optimization and inference techniques for Computer Vision **Final exam**
Date: December 4th, 2018
Teachers: Juan F. Garamendi, Coloma Ballester, Oriol Ramos, Joan Serrat **Time: 2h30min**

- Books, lecture notes, calculators, phones, etc. are not allowed.
- All sheets of paper should have your name.
- Answer each problem in a separate sheet of paper.
- All results should be demonstrated or justified.

Problem 1 *Juan F. Garamendi, 2 Points*

Let

$$\begin{aligned} J: \quad \mathcal{V} \quad &\to \mathbb{R}, \\ u \quad &\mapsto J(u) = \int_\Omega \mathcal{F}\left(\mathbf{x}, u(\mathbf{x}), \nabla u(\mathbf{x})\right) d\mathbf{x} \end{aligned}$$

be a convex energy functional over functions u, where

- $\mathcal{V}$ is a suitable space of functions.
- $\Omega \in \mathbb{R}^d$ is a bounded open domain of the d dimensional euclidean space $\mathbb{R}^d$.
- $u \in \mathcal{V}$, $u : \Omega \to \mathbb{R}$ is a scalar function defined on Ω.
- $\mathbf{x} \in \Omega$ such that $\mathbf{x} = (x_1, .., x_d)$ is the spatial variable and ∇ is the gradient operator such that $\nabla u(\mathbf{x}) = (u_{x_1}, .., u_{x_d})$

(a) (0.5 points) Say in a few words which is the fundamental problem in calculus of variations.

(b) (1 points) Let $J(u)$ be (strictly) convex. Explain in few words the difference between minimizing $J(u)$ using 1) calculus of variations and 2) the backpropagation strategy

(c) (0.5 points) What if $J(u)$ is not convex ?

Problem 2 *Juan F. Garamendi, $min(1, 2.a + 2.b)$ Point*

You have decided to use the TensorFlow (open source) library for minimizing $J(u)$ using calculus of variations. At the end, you have an algebraic system of equations $A\bar{x} = b$ where A is a well-conditioned mxm square matrix and $\bar{x}$ and $\bar{b}$ are vectors of size m and $\bar{x}$ is the unknown. In figure 1 you can find screenshots from tensorflow documentation.

tf.linalg.lstsq

Aliases:

- `tf.linalg.lstsq`
- `tf.matrix_solve_ls`

```
tf.linalg.lstsq(
    matrix,
    rhs,
    l2_regularizer=0.0,
    fast=True,
    name=None
)
```

Defined in `tensorflow/python/ops/linalg_ops.py`.

Solves one or more linear least-squares problems.

tf.linalg.solve

Aliases:

- `tf.linalg.solve`
- `tf.matrix_solve`

```
tf.linalg.solve(
    matrix,
    rhs,
    adjoint=False,
    name=None
)
```

Defined in generated file: `tensorflow/python/ops/gen_linalg_ops.py`.

Solves systems of linear equations.

tf.linalg.svd

Aliases:

- `tf.linalg.svd`
- `tf.svd`

```
tf.linalg.svd(
    tensor,
    full_matrices=False,
    compute_uv=True,
    name=None
)
```

Defined in `tensorflow/python/ops/linalg_ops.py`.

Computes the singular value decompositions of one or more matrices

Figura 1: `Matrix` and `tensor` words are both alias for A matrix. `rhs` word is an alias for b vector.

(a) (1 points) Which function from figure 1 you should try first? Explain in a few words why.

(b) (1 points) Bonus track for curious students: Explain the difference between solving the inpainting problem as you solve in the project, and the backprogpagation tensorflow implementation given in class. What are the advantages and disadvantages of each one?

Problem 3 — Coloma Ballester *0.75 Points*

Consider the constrained minimization problem

$$\begin{array}{ll} \min\limits_{x_1,x_2} & -x_1+3x_2 \\ \text{subject to} & \frac{1}{2}x_1 - x_2 + 2 \geq 0 \\ & x_1 - 3 \leq 0, \\ & x_1 + x_2 + 3 \geq 0. \end{array}$$

(a) Sketch the set of constraints of the problem. Is it a convex set? (0.2 *points*)

(b) Write the KKT optimality conditions for the problem. (0.4 *points*)

(c) Check if any of the points $(x_1, x_2) = (0, -3)$ and $(x_1, x_2) = (3, -6)$ could be the solution of the problem using the KKT conditions. (0.15 *points*)

Problem 4 — Coloma Ballester *0.75 Points*

Let us consider the vectors $\mathbf{x}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^n$, the $n \times n$ real matrix A, $\alpha > 0$, and the following minimization problem:

$$\min_{\mathbf{x}} \|A\mathbf{x}\| + \frac{\alpha}{2}\|\mathbf{x} - \mathbf{b}\|^2 + \langle \mathbf{x}, \mathbf{c} \rangle$$

(a) Is this a convex function? Why? (0.15 *points*)

(b) Write an equivalent min-max problem and the resulting iterations of a primal-dual algorithm to solve it. (0.4 *points*)

(c) Outline the dual algorithm to solve this problem? (0.2 *points*)

Problem 5 J.Serrat *0.5 Points*

We saw that binary image denoising could be modeled as a problem of maximum a posteriori inference over the graphical model of figure 2. The goal then was

$$\arg\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) = \arg\max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x})\, p(\mathbf{x})$$

(a) What are $\mathbf{x}$ and $\mathbf{y}$?

(b) What are the names for the $p(\mathbf{y}|\mathbf{x})$ and $p(\mathbf{x})$ terms ?

(c) Which is the "order" of the model ? (can be a number or a word)

(d) The term $p(\mathbf{y})$ does not appear, how comes we can get rid of it ?

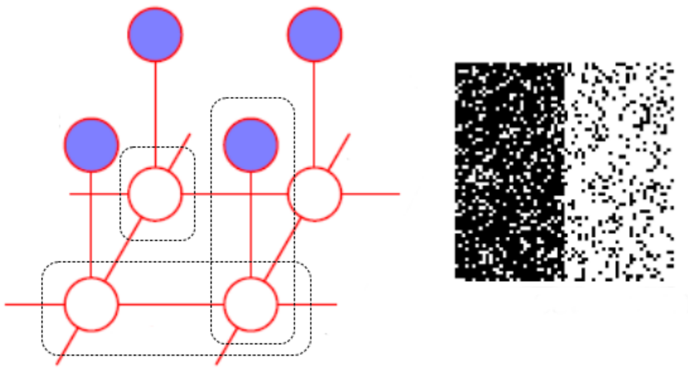


Figura 2: graphical model for binary image denoising

Problem 6 J.Serrat *0.5 Points*

Again with respect to the binary denoising example, we saw that the final equation was

$$\arg\min_{\mathbf{x}} \sum_i \alpha x_i + \sum_{j \in \mathrm{Ne}_i} \beta x_i x_j + \sum_i \gamma x_i y_i$$

where $x_i, y_i \in \{-1, +1\}$ and Ne_i means the neighbors of pixel i. What's false then ? (can be none, one or more choices)

(a) α is related to the mean intensity we expect in a solution

(b) large and positive β makes the solution more smooth

(c) with this formulation we can express our preference for a less smooth solution around image edges

(d) γ is a parameter of the prior

(e) α, β and γ can be learned from training samples

Problem 7 J.Serrat *0.5 Points*

Consider the graphical model of figure 3 where observations are binary images $16 \times 8 = 128$ pixels, that is, $x_i \in \{0,1\}^{128}$, $y_i \in Y = \{\mathrm{a}, \mathrm{b} \ldots \mathrm{z}\}$ (26 lowercase letters), $x = (x_1 \ldots x_9), y = (y_1 \ldots y_9)$.

We want to learn w to later infer a word from a series of binary images of letters as

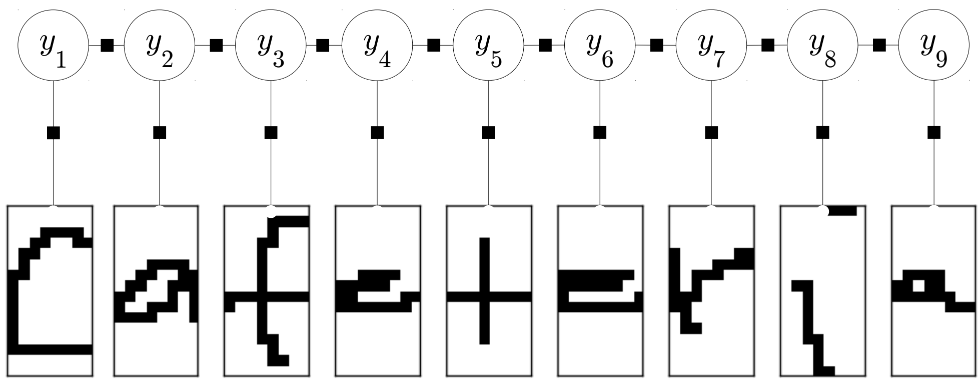


Figura 3

$$
\begin{aligned}
y^\star &= \arg\max_{y \in \mathcal{Y}} \; \langle\, w, \psi(x,y)\,\rangle \\
&= \arg\max_{y \in Y^9} \sum_{i=1}^{9} \sum_{p=\text{a}}^{\text{z}} \sum_{j=1}^{16} \sum_{k=1}^{8} w_{pjk}\, x_{ijk} \;+\; \sum_{i=1}^{8} \sum_{p=\text{a}}^{\text{z}} \sum_{q=\text{a}}^{\text{z}} w_{pq}\, \mathbf{1}_{y_i=\text{p},\, y_{i+1}=\text{q}}
\end{aligned}
$$

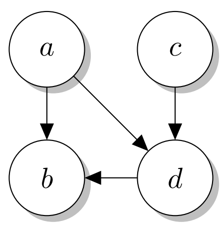
where $\mathbf{1}_{y_i=\text{p},\, y_{i+1}=\text{q}}$ evaluates to 1 if $y_i = \text{p}$ and $y_{i+1} = \text{q}$. In this context,

(a) does it make sense to apply the technique of two stage learning ? why and how ?

(b) what does $w_{p=\text{a},q=\text{b}}$ mean or represent ?

Problem 8 *Oriol Ramos Terrades, 0.5 Points*

Given the following Bayesian network (BN):



a) Write the joint distribution given by the conditional probabilities inferred from the BN.

b) Draw a factor graph derived from the BN. Also, write the definition of its factor functions in terms of the conditional probabilities used in the joint distribution.

Problem 9 *Oriol Ramos Terrades, 1 Point*

Say whether the next statements are true (**T**) or false (**F**) [Correct: +0.25, Incorrect: -0.25, unanswered: 0 points].

a) Belief propagation infers exact marginals in acyclic directed graphs.

b) The complexity of the message passing algorithm on a chain model is $O(N^2K)$, where N is the number of identically distributed random variables, X_i, and K is the number of states of X_i.

c) Samples that are consecutively generated by the Gibbs sampling method are independently distributed.

d) The Normalized importance sampling method generates unbiased estimators.