

Classificare bioimmagini con le reti neurali

Vincenzo Della Mea

Dept. of Maths, Computer Science and Physics, University of Udine, Italy

<http://users.dimi.uniud.it/~vincenzo.dellamea/>



UNIVERSITÀ
DEGLI STUDI
DI UDINE
hic sunt futura

Obiettivo

- In questa sessione impareremo come usare le reti neurali per riconoscere il cancro in immagini da microscopio
- Faremo tutti i passi necessari, utilizzando un sistema in cloud gratuito
- L'esercizio è riusabile su altre immagini con pochissime modifiche
- ... ma è solo uno dei tanti modi per sperimentare con il deep learning

Il processo completo

- Abbiamo un insieme di immagini **annotate**,
 - Cioè **etichettate** come rappresentative di cancro oppure no
- L'insieme è suddiviso in due sottoinsiemi:
 - **Training set**, ~80%, per il training vero e proprio;
 - **Validation set**, ~20%, for controllare il risultato dell'addestramento ad ogni passo
 - Controlliamo su un set diverso da quello di training per evitare di addestrare il sistema a riconoscere solo le immagini cui è stato addestrato!
 - Quando siamo soddisfatti del nostro modello, possiamo usarlo su immagini non etichettate (il **test set**)

Cosa ci serve

- Dalla comunità:
 - Python 3
 - Jupyter
 - Google Colab
 - Fastai
- Da voi:
 - Un account gmail
 - Un browser recente
- Pochissima conoscenza di programmazione
- Da me:
 - Un Notebook di Jupyter
 - Un insieme di immagini

Python

- L'addestramento di reti neurali avviene prevalentemente scrivendo ed eseguendo programmi che le implementano e realizzano l'addestramento
 - Però esistono librerie pronte che fanno quasi tutto
 - Torch, Tensorflow, Keras
 - Rimane solo da impostare i parametri di configurazione delle reti
 - Quindi di solito si programma "poco"
- In quest'epoca **Python** è il linguaggio più usato per queste operazioni

Jupyter

- Jupyter è **un'applicazione web open-source** che consente di creare, eseguire e condividere documenti che contengono programmi, commenti, visualizzazioni ecc
 - <https://jupyter.org>
- Noi non lo installeremo, perché...

Google Colab

- In generale, per fare addestramento su grandi dataset abbiamo bisogno di tanta potenza di calcolo, tipicamente fornita da GPU
 - Es. nVidia 3090, 16GB RAM, 2000€ (+ il computer!)
 - Nel nostro esercizio: GPU: 9s/epoca, CPU: 3 minuti/epoca
- Google Colab è un sistema collaborativo che permette di eseguire gratis **Jupyter notebooks** su Google Cloud, anche con GPU, per meno di 12 ore/giorno, senza garanzie
 - Utile per esercizi ed esperimenti
- <http://colab.research.google.com>

Fastai

- Fastai prima di tutto è un corso online su deep learning
 - <https://www.fast.ai>
- Ma è anche una libreria di alto livello che permette di realizzare numerosi compiti di deep learning, non solo su immagini
 - <https://docs.fast.ai/vision.html>
 - Si basa a sua volta su un'altra libreria: pyTorch
- Il nostro esercizio si basa su Fastai

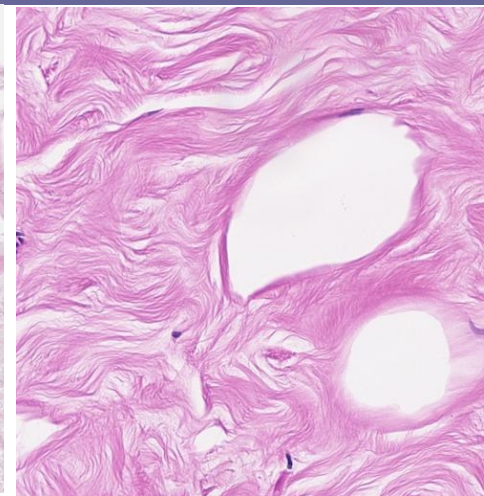
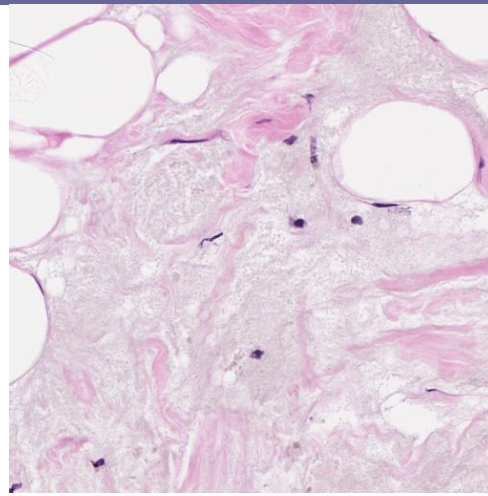
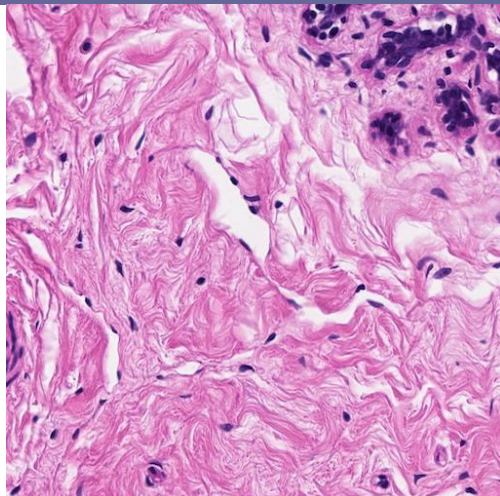
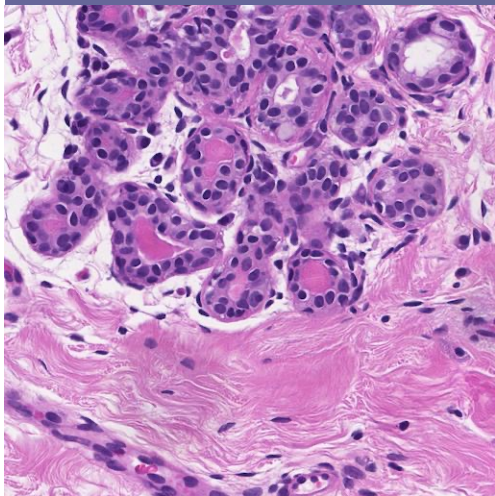
Le immagini

- Fastai vuole una specifica organizzazione delle immagini:
 - Per distinguere training, validation e test data sets
 - Per distinguere le classi che ci interessano
- Immagini in cartelle organizzate come segue:
- **Mydata**
 - **train**
 - **Class1** -> il nome della cartella è il nome della classe!
 - Immagini della classe1
 - **Class2** -> il nome della cartella è il nome della classe!
 - Immagini della classe1 2
 - ...
 - **valid**
 - **Class1** -> il nome della cartella è il nome della classe!
 - Immagini della classe1
 - **Class2** -> il nome della cartella è il nome della classe!
 - Immagini della classe1
 - ...
 - **test**

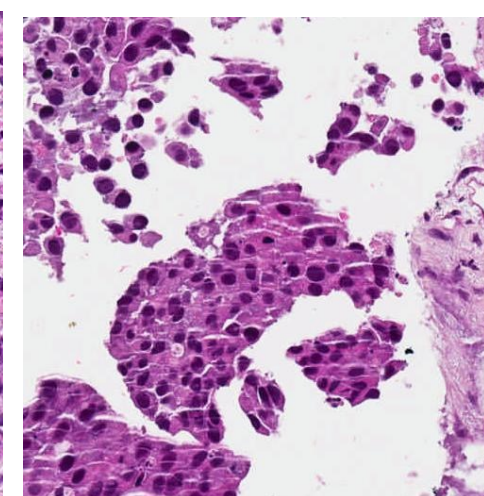
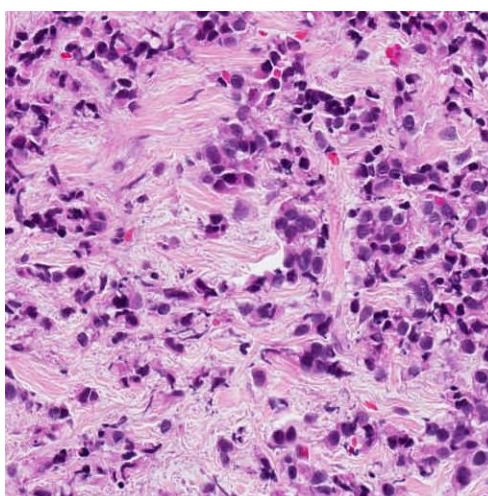
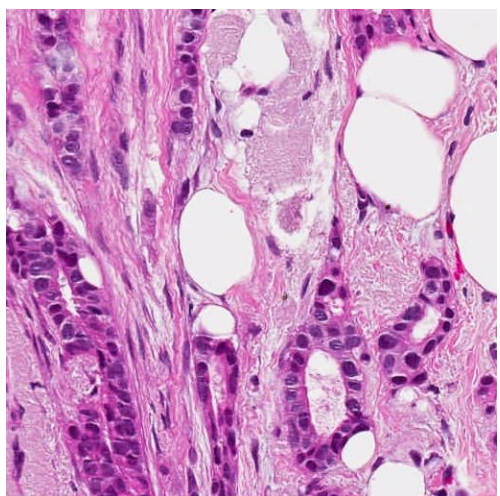
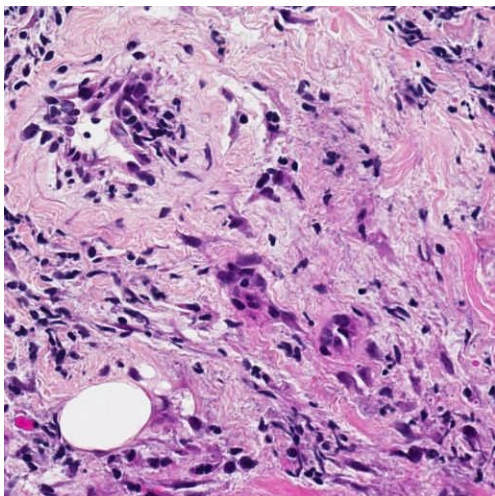
Il nostro set di immagini

- ~ 700 immagini di biopsie o noduli della mammella, 512x512 pixels, acquisite al microscopio con ingrandimento 20x
 - (prese da 30 casi, quindi immagini non indipendenti)
- Cartelle:
- IATS2020
 - **train**
 - **0-NonCancer**
 - Immagini senza cancro
 - **1-Cancer**
 - Immagini senza cancro
 - **valid**
 - **0-NonCancer**
 - Immagini senza cancro
 - **1-Cancer**
 - Immagini senza cancro
 - **test**

Le immagini



**Non cancro
cancro**



Da dove vengono queste immagini?

Vetrino istologico: il tessuto viene fissato in formalina e poi incorporato in paraffina per renderlo sezionabile,

- Poi dal blocco vengono tagliate sezioni da circa 3-4micron tramite il microtomo
- Provenienza: biopsia, prelievo operatorio

Al naturale, il campione è trasparente

- A causa del ridotto spessore

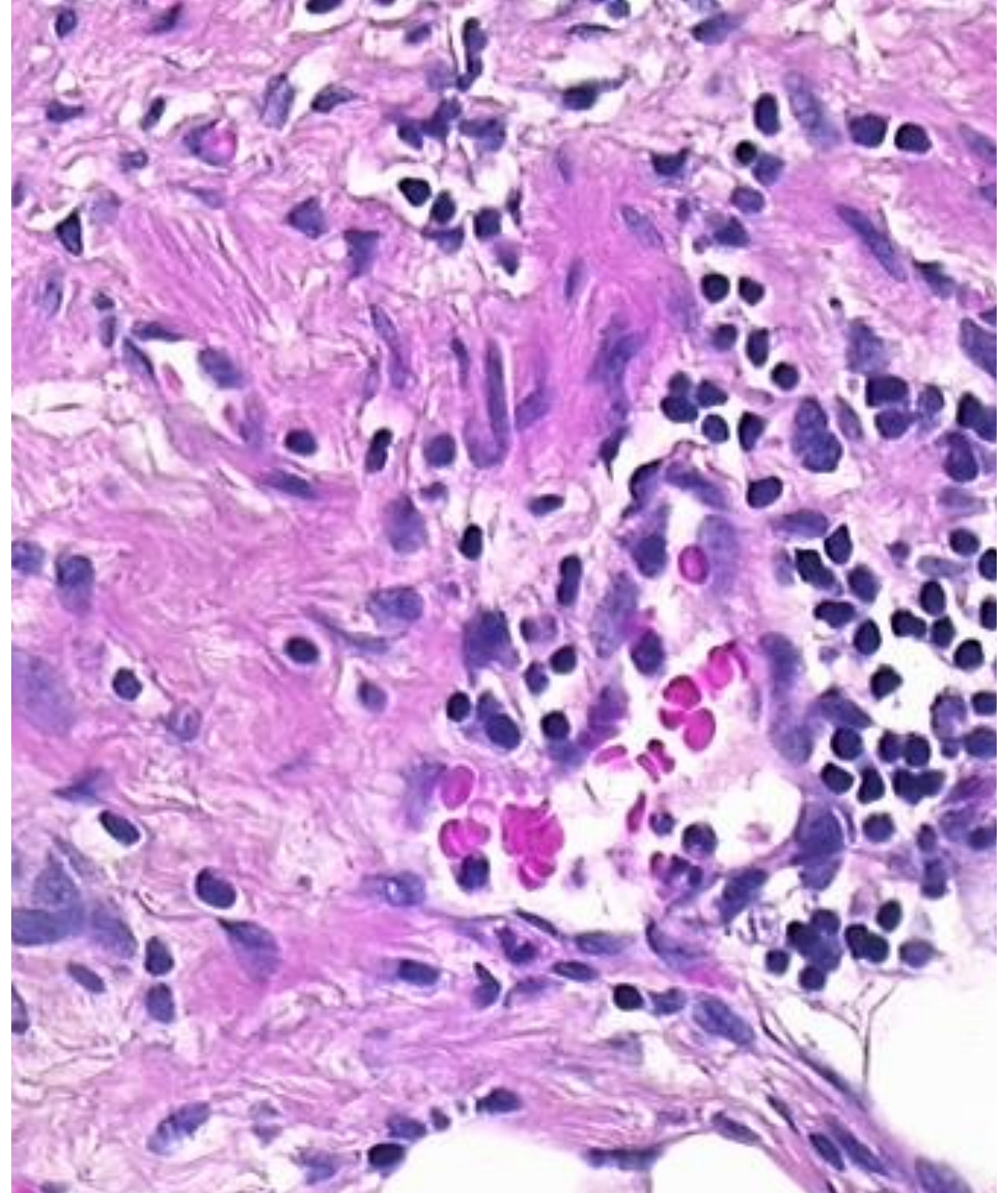
Viene quindi colorato

- Con coloranti che si legano a strutture o sostanze presenti nelle cellule (per affinità chimica, permeabilità, concentrazione)
- Evidenziando di volta in volta ciò che interessa
- Anche più sezioni per campione



Istologia: ematossilina/eosina

- Colorazione più usata
 - Ematossilina: colorante basico **blu** che si lega ai nuclei
 - Eosina: colorante acido **rosa** che si lega al citoplasma
- Il patologo osserva gli aspetti *morfologici* delle cellule e dei tessuti, e in base a ciò che vede, fa una diagnosi



Il nostro esercizio

- Andate a:
- <https://colab.research.google.com/>
- E caricate il notebook **CampusUNIUD.ipynb**
- Poi seguitemi... le istruzioni sono autocontenute nel notebook.

Altre cose da fare

- Buona parte del lavoro riguarda l'ottimizzazione degli **iperparametri** per raffinare la performance
 - Modello, epoche, *dropout*, batch size, learning rate,...
 - **Grid search**
 - Si prova tutto
 - (tanti addestramenti, tanto tempo, tanta elettricità)
- Ottimizzare il training set
 - Per esempio, bilanciare il numero di immagini per classe

Volete addestrare per altre immagini?

- Create la struttura di cartelle:
 - `mieidati`
 - `train`
 - `valid`
 - `test`
- Elencate le classi che vi interessano
 - cane, gatto, compagnodiclasse, ...
- Dentro train e valid, create cartelle con i nomi delle classi
- Mettete le immagini in train e valida (80%-20% circa)
- Caricate su colab
- Cambiate i due parametri relativi a dataset e eventuale test
 - `path='data/mieidati'`
 - `testpath='data/mieidati/test/'` (NB solo se avete anche un test set)
- Mandate in esecuzione il notebook (eccetto la parte che carica le immagini del nostro esercizio)

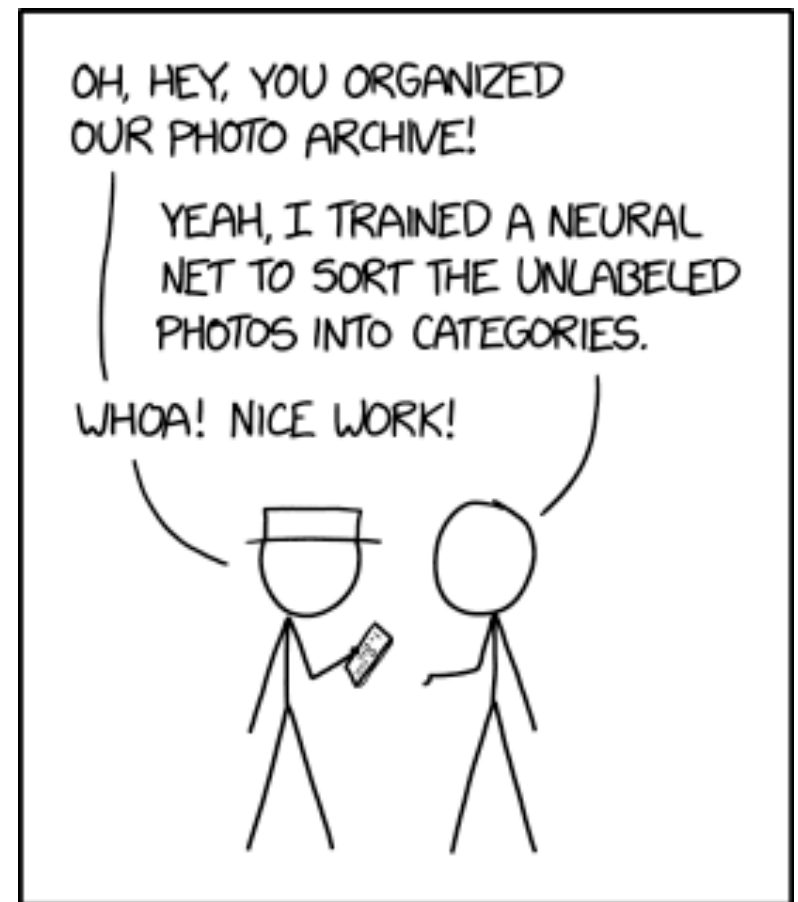


UNIVERSITÀ
DEGLI STUDI
DI UDINE

hic sunt futura

Domande?

GRAZIE



ENGINEERING TIP:
WHEN YOU DO A TASK BY HAND,
YOU CAN TECHNICALLY SAY YOU
TRAINED A NEURAL NET TO DO IT.

xkcd.com