# Melanoma Skin Cancer Detector

Design of a computer-based system to automatically classify histopathological images of skin tissue

Glòria Macià Muñoz
ETH Studentin D-ITET, Biomedical Engineering MSc

# Abstract

Malignant melanoma is the most dangerous form of skin cancer, but early diagnosis plays an essential role in the control and cure of the disease. In this presentation, a computer-based system to classify histopathological images of skin tissue using artificial neural networks is implemented. Performance measures of the proposed system are encouraging and there is no evidence of overfitting. Therefore, an extended version of this system could be used to assist hospital pathologists and increase the efficiency of our healthcare system.

# Disclaimer

The dataset used in this project was provided by the Pathology Unit – Hospital del Mar – Parc de Salut Mar (Barcelona, Spain) and must be used solely for the purpose of research and education. The same applies to the Neural Network Classifier which has not undergone the CE marking regulatory process for medical and in vitro diagnostic devices in the European Unit.

# Introduction - Melanoma Skin Cancer

- Skin cancer is by far the most common of all cancers. Melanoma accounts for only about 1% of skin cancers but causes a large majority of skin cancer deaths.
- According to The American Cancer Society's estimates about 87,110 new melanomas will be diagnosed in 2017 and about 9,730 people are expected to die of it.

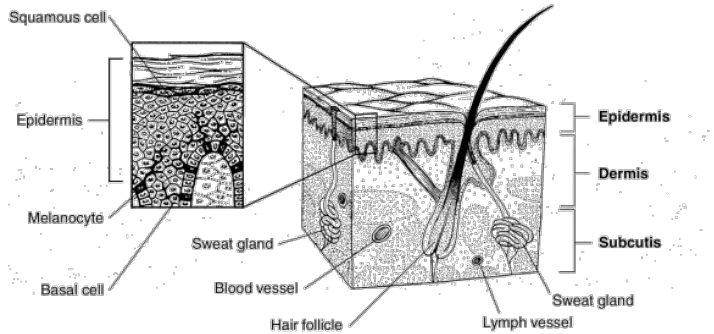
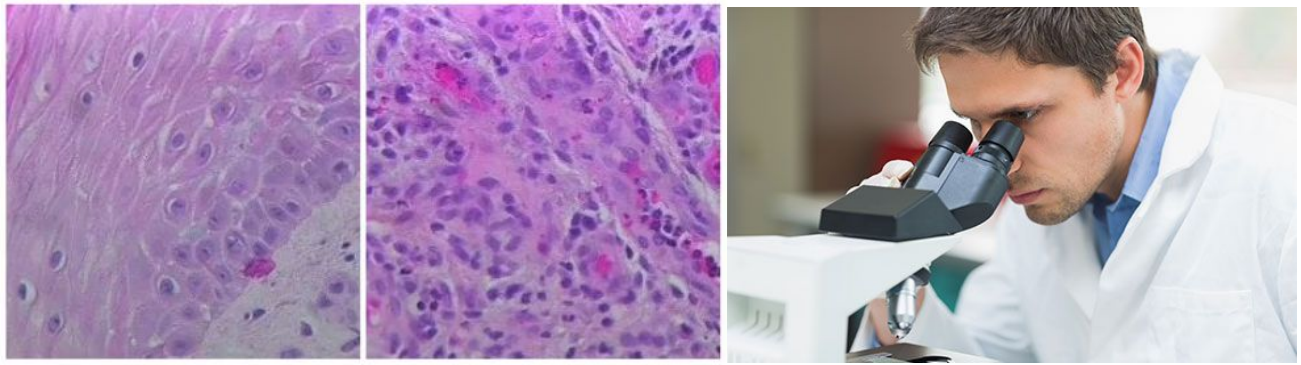



Figure and data from The American Cancer Society's website

- It usually starts in a certain type of skin cell called melanocyte.
- In a context of rising healthcare costs, solutions to increase efficiency are needed.

# Objective - Melanoma Skin Cancer Detector

- Design a computer-based system to classify histopathological images of skin tissue using artificial neural networks.



Histological images of skin tissue obtained using an optical microscope with an 400x lense. Samples were hematoxylin-eosin stained during Mohs surgery. On the left, a portion of normal skin tissue. On the right, a malignant melanoma cancerous sample.

# Methods - DataSet Generation

- Histological Image Data → Image Processing → Feature Extraction

Hospital del Mar

Parc de Salut MAR Barcelona

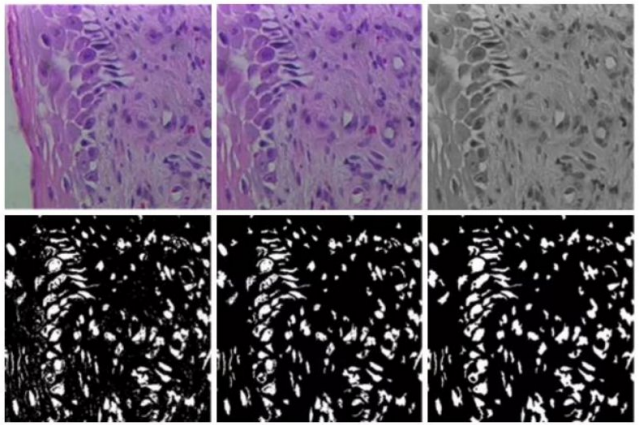


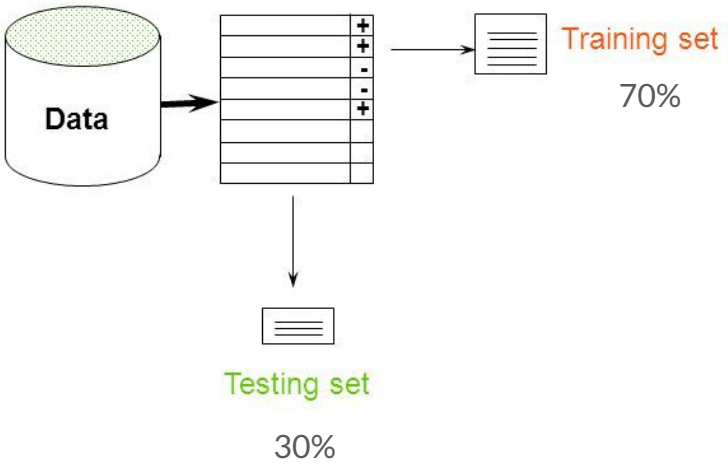
**Figure 3:** Histological image data processing. From upper left to bottom right: original histological sample, cropped sample, grayscale image, thresholded image, connectivity analysis performed and resulting image.

| NCR | Nuclei Count | Size variance | Label |
|---|---|---|---|
| 0,1231 | 69 | 1.03E+09 | 0 |
| 0,1117 | 60 | 1.77E+08 | 0 |
| 0,0986 | 57 | 1.54E+09 | 0 |
| ... | ... | ... | ... |
| 0,3125 | 128 | 5.61E+09 | 1 |
| 0,2950 | 156 | 1.65E+09 | 1 |
| 0,2041 | 109 | 1.97E+09 | 1 |

**Table 1:** Excerpt from the .xlsx file containing the three input features for the classifier plus the ground truth label. In the ground truth label column, 0 corresponds to a normal image sample whereas 1 corresponds to a cancerous one.

# Methods - Artificial Neural Network Classifier

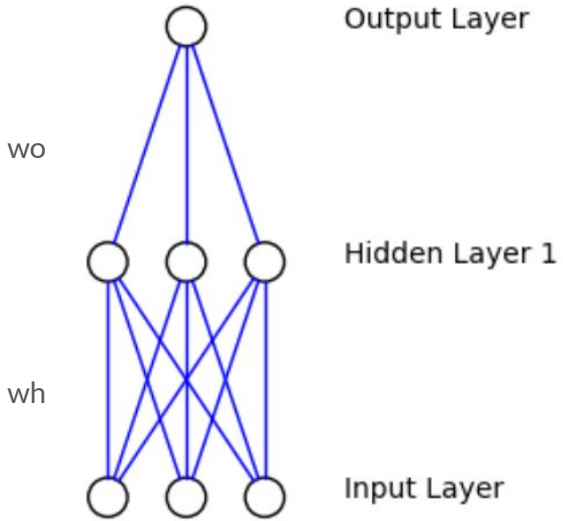
- Importing Data (100 samples) - Training & Testing Set




Importing Data – Training & Testing Set

```
In[1]:= nImages = 100; (* number of sample images *)
nTrainset = 70; (* number of images used for training the ANN *)
data = Import["/home/gloria/Documents/ETH/Semester2/NN/ImageDatav2.xlsx",
   {"Data", 1, Range[2, nImages + 1], {1, 2, 3}}];
(* data file has a header in row 1 *)
target = Import["/home/gloria/Documents/ETH/Semester2/NN/ImageDatav2.xlsx",
   {"Data", 1, Range[2, nImages + 1], 4}];
(* 4th column corresponds to the ground truth *)
trainindex = RandomSample[Range[2, nImages], nTrainset];
(* Randomly split images betweeen training and testing set *)
testindex = Complement[Range[2, nImages], trainindex];
```

# Methods - Artificial Neural Network Classifier




Defining the ANN with 1 hidden layer

```
In[7]:= inNum = 3; hidNum = 3; outNum = 1; (* 3 input units,
        3 hidden units and 1 output unit *)
        wh = Table[Table[Random[Real, {-2.5, 2.5}], {inNum}], {hidNum}];
        (* hidden weights *)
        wo = Table[Table[Random[Real, {-2.5, 2.5}], {hidNum}], {outNum}];
        (* output weights *)
        eta = 0.5;(* learning rate *)

In[11]:=
        sigmoid[x_] := 1 / (1 + Exp[-x]);
        relu[x_] := Map[Max[0, #] &, x]; (* activation functions *)
        l = nTrainset; (* number images in the training set *)
```

**Figure 5:** Artificial Neural Network (ANN) architecture.

# Methods - Artificial Neural Network Classifier

- Training the ANN

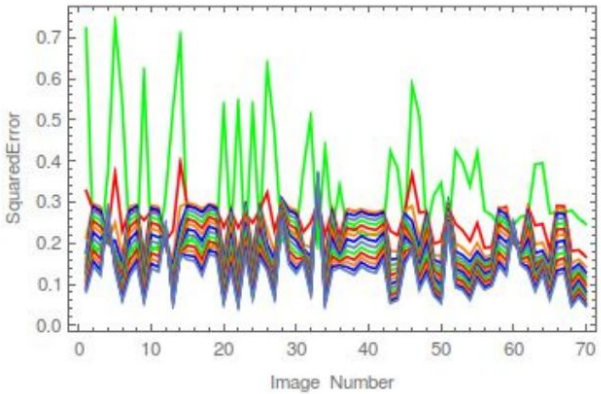



**Figure 6:** Squared Error for each image in the training set (training set size = 70) decreases epoch after epoch. A total of 15 iterations are done for every image.

```
In[13]:= (* funtion to train the ANN *)
TrainEpochs[n_] := Module[{iter = nTrainset}, Table[
    If[k ≤ l, r = trainindex[[k]], r = trainindex[[Mod[k, l] + 1]]];
    in = data[[r]];
    y = target[[r]];
    outhid = sigmoid[wh.in];
    out = sigmoid[wo.outhid];
    e = y - out;
    outdelta = e out (1 - out);
    hiddelta = outhid (1 - outhid) Transpose[wo].outdelta;
    wo += eta e.e Outer[Times, outdelta, outhid];
    wh += eta e.e Outer[Times, hiddelta, in];
    {e.e, wo, wh}, {k, 1, iter}]];
```

* For details about the mathematics behind the algorithm, see the supplementary materials of the presentation

# Methods - Artificial Neural Network Classifier

- Performance Evaluation

| Performance Measure | Value |
|---|---|
| Sensitivity | 1 |
| Precision | 0.933 |
| Specificity | 0.933 |
| Accuracy | 0.966 |

Is there overfitting?

In[22]:=
```
TP = 0; FP = 0; TN = 0;
FN = 0; (* true positives, false positives, true negatives and false negatives *)
For[i = 1, i ≤ Length[testindex], i++,
  in = data[[testindex[[i]]]];
  y = target[[testindex[[i]]]];
  outhid = sigmoid[wh.in];
  out = Round[sigmoid[wo.outhid]];
  Which[out[[1]] == y == 1, TP += 1, out[[1]] == y == 0, TN += 1,
   out[[1]] ≠ y && out[[1]] == 0, FN += 1, out[[1]] ≠ y && out[[1]] == 1, FP += 1]
 ];
```

In[24]:=
```
(* Compute True Positive Rate (TPR), also called recall or sensitivity*)
NSensitivity = TP / (TP + FN);
(* Compute Precision, also called Positive Predictive Value (PPV)*)
NPrecision = TP / (TP + FP);
(*Compute True Negative Rate (TNR), also called specificity*)
NSpecificity = TN / (TN + FP);
(* Compute Accuracy*)
NAccuracy = (TP + TN) / (TP + TN + FP + FN);
```

# Methods - Artificial Neural Network Classifier

- Assessing overfitting

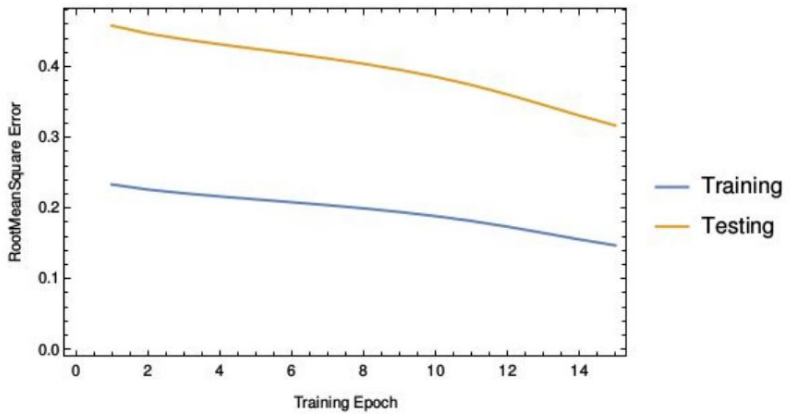
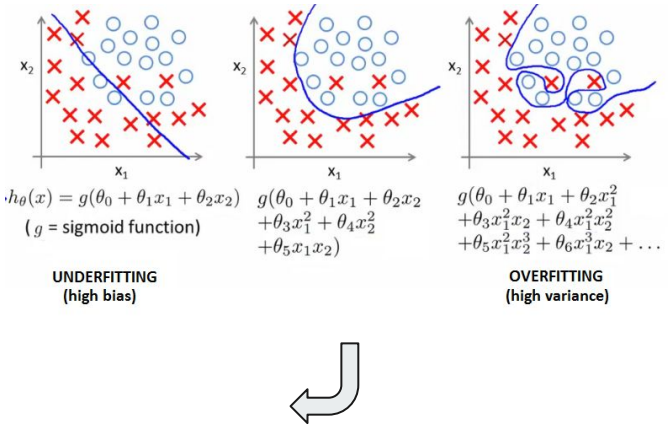



**Figure 7:** Root Mean Squared (RMS) Error of the ANN classifier plotted at every epoch for both the training and testing set.



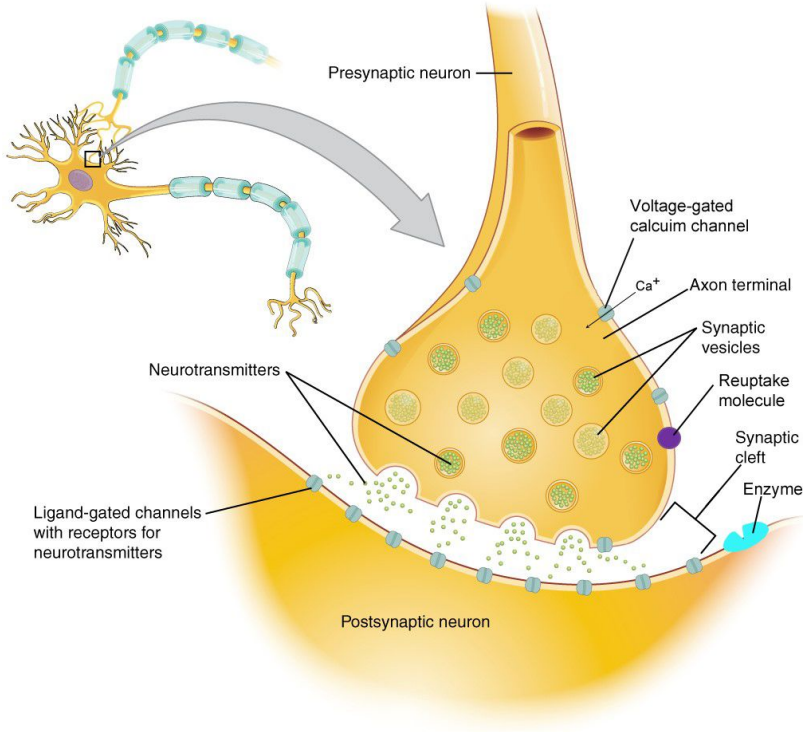
# Conclusions

- It is indeed possible to design a computer-based system to classify histopathological images of skin tissue using ANN.
- Even though the proposed system has not undergone the European CE marking regulatory process for medical devices, performance measures are very encouraging and there seems to be no evidence of overfitting.
- An improved version of such a system could be used to assist pathologists in their day-to-day hospital tasks so that they can spend their limited time better and contribute to increase the efficiency of our healthcare system.

# Thanks a lot for your attention

A special thank to Prof. Dr. Ruedi Stoop for teaching the course *227-1040-00L Theory, Programming and Simulation of Neuronal Networks* at ETH Zürich and to the TAs of the course for answering my doubts during the implementation of this project.

# Supplementary Materials



# Methods - Feature Extraction

In this research, 3 main tissue features were extracted for the subsequent classification task:

- Nuclear-cytoplasmic ratio (NCR)
- Nuclei number
- Pleomorphism (nuclei size variance)

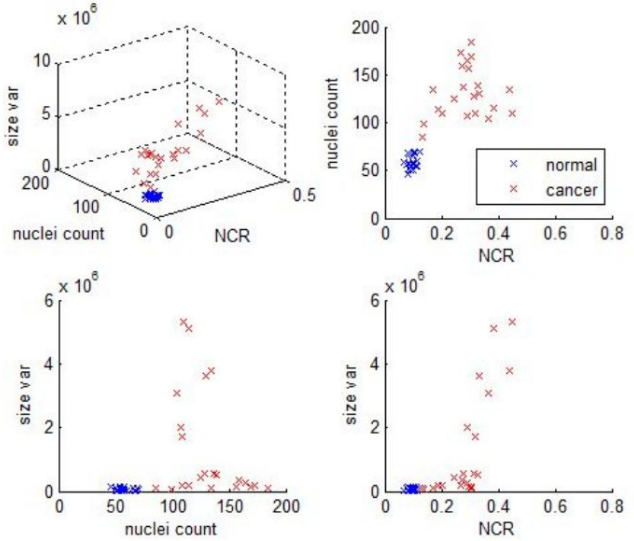



**Figure 4:** Feature space in the three features' dimensions and plane projections. Instances correspondent to normal images are represented in blue, while instances of cancer images are represented in red.

# Methods - Training the ANN

- Most ANNs contain some form of 'learning rule' which modifies the weights of the connections according to the input patterns that it is presented with.
- Here, the delta rule was used. This rule is frequently utilized by the most common class of ANNs called 'backpropagational neural networks' (BPNNs).
- In this class of ANNs, learning is a supervised process that occurs with each cycle or 'epoch' (i.e. each time the network is presented with a new input pattern) through a forward activation flow of outputs, and the backwards error propagation of weight adjustments.

# Methonds - Training the ANN

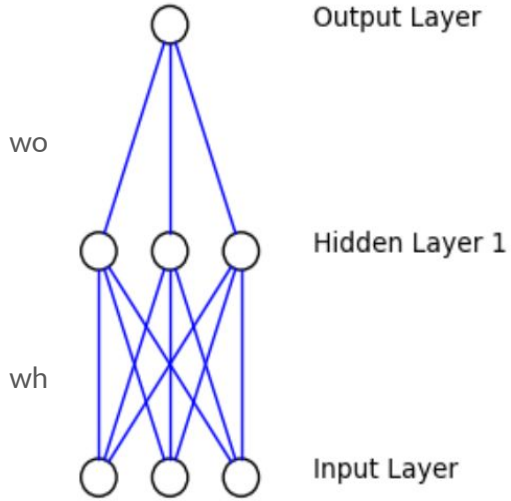



**Figure 5:** Artificial Neural Network (ANN) architecture.

$$E = \frac{1}{2}(t - out)^2$$

$$out = s(w_o \cdot outhid) \; ; \; outhid = s(wh \cdot in)$$

input

$$\frac{dE}{dw_o} = (t - out)\frac{dout}{dw_o} = \underbrace{(t - out) \cdot out \cdot (1 - out)}_{outdelta} \cdot outhid$$

$$\frac{dE}{dwh} = (t - out)\frac{dout}{dwh} = \overbrace{\underbrace{(t - out)\, out\, (1 - out)}_{outdelta}\, w_o \cdot outhid\, (1 - outhid)}^{hiddelta} \cdot in$$

$$\begin{cases} w_o = w_o + \eta \cdot \underbrace{outdelta \cdot outhid}_{\frac{dE}{dw_o}} \\ wh = wh + \eta \cdot \underbrace{hiddelta \cdot in}_{\frac{dE}{dwh}} \end{cases}$$

# Bibliography


1. Patton M. U.S. Health Care Costs Rise Faster Than Inflation. Forbes. https://www.forbes.com/sites/mikepatton/2015/06/29/u-s-health-care-costs-rise-faster-than-inflation/#1d9013c96fa1. Published June 29, 2015. Accessed June 1, 2017.
2. 4 Ways Hospitals Can Solve the Problem of Rising Healthcare Costs. Health Catalyst. https://www.healthcatalyst.com/hospitals-solve-rising-healthcare-costs. Published June 3, 2017. Accessed June 1, 2017.
3. Key Statistics for Melanoma Skin Cancer. American Cancer Society. https://www.cancer.org/cancer/melanoma-skin-cancer/about/key-statistics.html. Accessed June 1, 2017.
4. Hospital del Mar. Hospital del Mar - Parc de Salut Mar. https://www.parcdesalutmar.cat/hospitals/hospital-del-mar/en_presentacio.html. Accessed June 7, 2017.
5. WM-. Guidelines for Classification of Medical Devices - CE Marking (CE Mark) for Medical Devices - EU Council Directive 93/42/EEC. Guidelines for Classification of Medical Devices - CE Marking (CE Mark) for Medical Devices - EU Council Directive 93/42/EEC. http://www.ce-marking.org/Guidelines-for-Classification-of-Medical-Devices.html. Accessed June 7, 2017.
6. 227-1040-00L Theory, Programming and Simulation of Neuronal Networks. ETH Zurich - Course Catalogue. http://www.vvz.ethz.ch/Vorlesungsverzeichnis/lerneinheitPre.do?lerneinheitId=112257&semkez=2017S&lang=en. Accessed June 7, 2017.
7. 文章. http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html# - Open Access Library. http://www.oalib.com/references/7680507. Accessed June 12, 2017.
8. A Basic Introduction To Neural Networks. A Basic Introduction To Neural Networks. http://pages.cs.wisc.edu/~bolo/shipyard/neural/local.html. Accessed June 12, 2017.
9. Overfitting. Wikipedia. https://en.wikipedia.org/wiki/Overfitting. Published June 9, 2017. Accessed June 12, 2017.
10. S.J. Keenan, J. Diamond, W.G. McCluggage, H. Bharucha, D. Thompson, P.H. Bartels, P.W. Hamilton, An automated machine vision system for the histological grading of cervical intraepithelial neoplasia (CIN), Journal of Pathology 192 (3) (2000) 351–362.
11. M. Guillaud, K. Adler-Storthz, A. Malpica, G. Staerkel, J. Matisic, D.V. Niekirkd, D. Cox, N. Poulina, M. Follen, C. MacAulaya, Subvisual chromatin changes in cervical epithelium measured by texture image analysis and correlated with HPV, Gynecologic Oncology 99 (2005) S16–S23.
12. He, L. et. al. (2012). Histology image analysis for carcinoma detection and grading. Computer methods and programs in biomedicine , 538–55