

## Melanoma detection from images

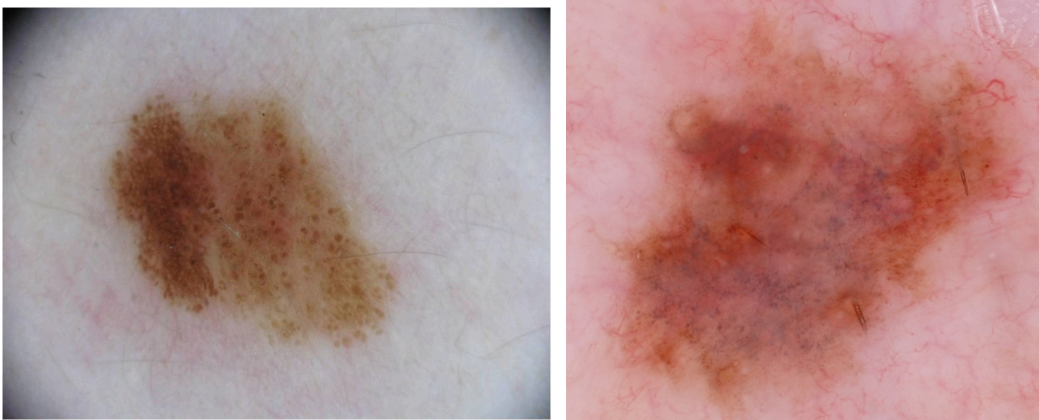
### Introduction:

Le but principal de ce projet est d'utiliser des **descripteurs morphologiques**, d'**intensité** et de **texture** les plus pertinents afin de classer des images en groupes distincts (développement d'un cancer de la peau ou non par exemple). Pour les SVM (Support Vector Machine, voir en annexe pour plus de détails), est utilisé pour la classification supervisée. Dans chacun des cas, la qualité des prédictions sera jugée à l'aide d'indicateurs comme *l'accuracy* (justesse ou non de la prédiction), le **F1-score** et la courbe **ROC**.

### **I. Mise en contexte**

Dans ce projet, nous utilisons les techniques vues en analyse d'images pour prédire au mieux la présence de mélanome (type de cancer de la peau). Cela permettra en particulier aux médecins disposer d'un outil complémentaire de diagnostic et de prévenir le développement de formes sévères.

Nous disposons d'une base d'images labélisées selon la présence de mélanome (label=1) ou non (label=0). Nous séparons notre base de données au total  $n=200$  exemples en ensemble d'apprentissage et de test.



*Figure 10: Lésion bénigne VS mélanome*

L'objectif est alors de construire un classifieur  $C_\theta : D \rightarrow \{0, 1\}$  qui soit capable de prédire la classe d'appartenance d'une image non encore rencontrée:  $C_\theta(x_{new}) = 1$  si  $x_{new}$  appartient à la classe 'high' et 0 sinon.

$\theta$  est l'ensemble des paramètres du classifieur qu'il faudra optimiser à partir des images déjà labélisées (ensemble d'apprentissage) pour qu'il puisse faire des prédictions optimales. Typiquement, pour un réseau de neurones,  $\theta$  est une matrice de poids à ajuster, pour une régression ou un SVM il s'agira d'un vecteur de paramètres à optimiser etc.

Pour construire l'ensemble de départ  $D$ , une première idée serait de prendre l'espace des images c'est-à-dire l'ensemble des matrices de taille  $1024 \times 1024$ . Le problème avec une telle approche est que l'optimisation du paramètre  $\theta$  du classifieur  $C_\theta$  demande un **très grand nombre de données d'apprentissage** car l'espace de départ  $D$  est de très grande dimension (il faudrait typiquement plusieurs milliers d'exemples). Cela n'est pas envisageable pour notre étude puisqu'on ne dispose que de **200** données au total. Il est alors nécessaire de **compresser l'information (réduction de la dimension)**, en extrayant les caractéristiques intéressantes de chaque image. Plus formellement, il s'agit d'utiliser un extracteur  $\phi : D \rightarrow D'$  avec associant à chaque image  $I \in D$ , un ensemble de descripteur  $\phi(I) \in D'$ , avec  $\dim(D') \ll \dim(D)$ . Le classifieur final aura alors pour ensemble de départ  $D'$  au lieu de  $D$ .

## II. Sélection des descripteurs et classification

Pour chaque individu, nous disposons d'une image original, de superpixels ainsi que de l'image segmentée. Le type d'image utilisée dépend évidemment de la nature du descripteur choisi.

### a. Descripteurs morphologiques

Les images segmentées sont utilisées pour la sélection des descripteurs morphologiques. Pour commencer, nous utilisons pour la classification des images 10 descripteurs morphologiques (*Shape Features*) que sont: le *périmètre de la région*, la *surface*, le *diamètre équivalent*, l'*excentricité*, le *demi-grand axe* et le *demi-petit axe* de l'ellipse circonscrite, le *ratio R* de ces 2 axes, la *solidité* et enfin le pourcentage de pixels dans la région par rapport au reste de l'image.

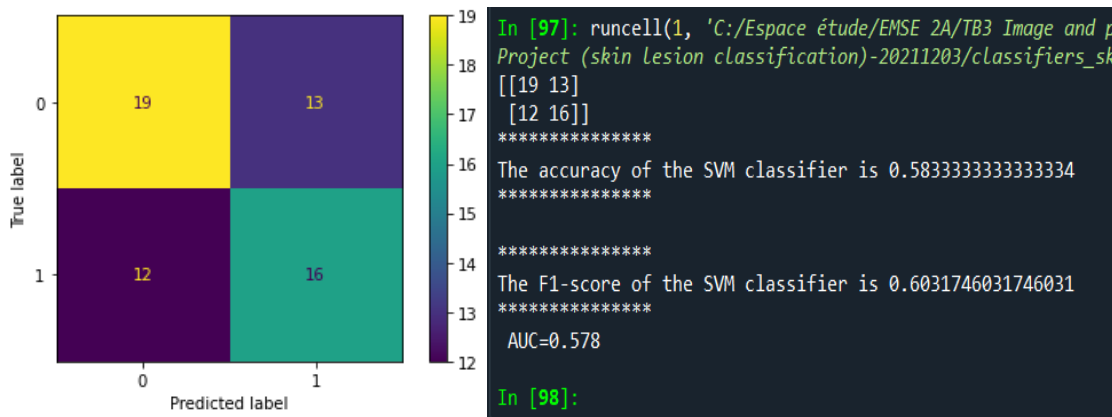


Figure 11: Résultats en utilisant les 10 descripteurs ShapeFeat

Le résultat est alors **très médiocre**, l'**accuracy étant de 58%** et l'**AUC (Area Under the (ROC) Curve) de 57%**. Ce modèle ne fait pas très largement mieux qu'un modèle naïf qui consisterait à prédire les classes de manière aléatoire. Cela se justifie par le fait que les descripteurs utilisés (périmètre et aire de la région d'intérêt, longueur des axes de l'ellipse circonscrite...) ne concernent que des **caractéristiques assez globales** de l'image et surtout sont des grandeurs **non normalisées**. Ainsi, il est nécessaire d'exploiter des **caractéristiques plus locales** de l'image et aussi de définir des **grandeurs normalisées indépendantes du niveau de zoom des photos**.

Nous choisissons alors certains **descripteurs normalisés**. Notons A, P, d, D et z respectivement l'aire, le périmètre, le grand diamètre de Frénet et le périmètre de Crofton. Nous définissons:

$$R_1 = \frac{4\pi A}{P^2}$$

$$e = d/D \text{ (elongation)}$$

$$t = \frac{r}{D} \text{ (thinness)}$$

$$rd = \frac{4 * A}{\pi * D^2} \text{ (roundness)}$$

Nous gardons aussi les grandeurs déjà normalisées comme l'excentricité dans l'ensemble des features de *ShapeFeat* et utilisons le **périmètre de crofton**. Voici les résultats obtenus:

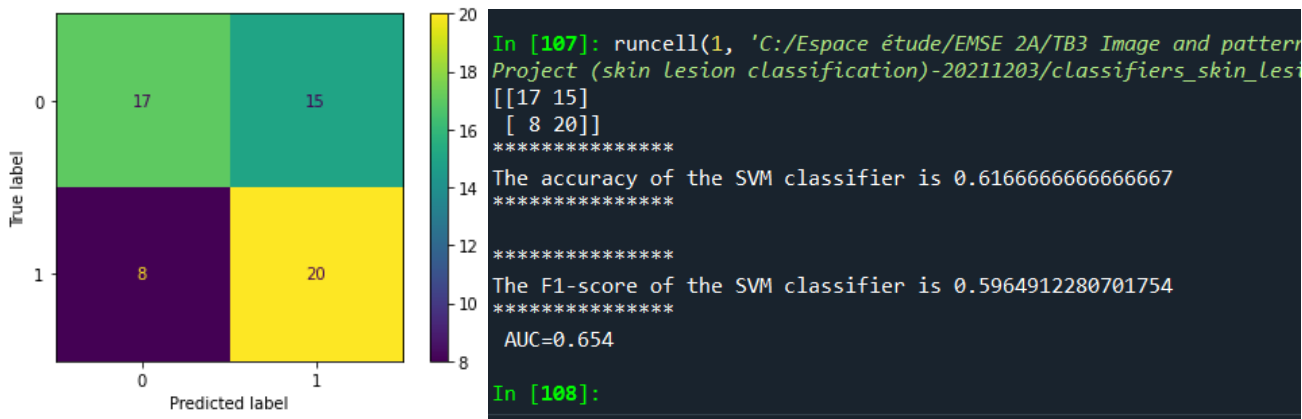


Figure 12: Utilisation de 8 descripteurs morphologiques normalisés

Bien qu'on note une amélioration, le résultat n'est toujours pas très satisfaisant. On se rend compte alors des **limites d'utiliser des descripteurs morphologiques** uniquement pour la classification de lésions cutanées. Il faut donc utiliser des **descripteurs d'intensité** et de texture qui ont l'avantage d'exploiter **des informations locales**.

## b. Descripteurs d'intensité et de texture

Nous utilisons dans cette partie les images originales ainsi que les superpixels pour les descripteurs d'intensités. Le premier descripteur de texture concerne les LBP (Local Binary Pattern). Il permet de comparer chaque pixel à l'ensemble de ses 8 voisins comme illustré sur la figure 13 ci-dessous.

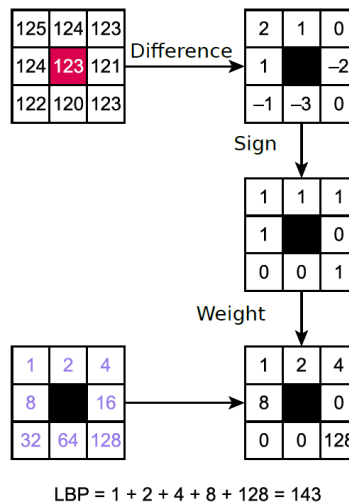


Figure 13.1: Local binary pattern. From wikipedia, author Xiawi, CC-BY-SA.

On peut alors générer un **histogramme de fréquence** pour chaque image, en comptant la fréquence des valeurs de LBP de ses pixels (le code est le même que celui du TP). Voici un histogramme les histogrammes obtenus pour de cas "confirmé" et "non confirmé" de melanoma.

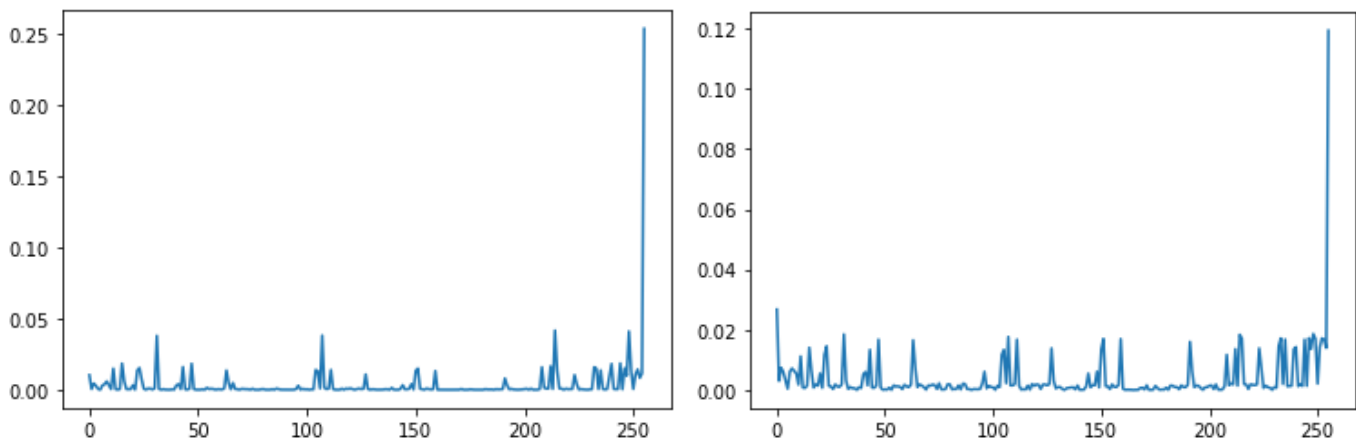


Figure 14: Histogramme LBP d'un cas non confirmé (à gauche) et d'un cas confirmé de melanoma (à droite)

Les deux histogrammes assez distincts permettent d'espérer qu'on pourra bien séparer les deux classes grâce aux LBP.

Toutefois, le problème est qu'on ne peut **ni donner en entrée aux classifieurs les points de l'histogramme** (nombre de features=256 beaucoup trop grand vu le nombre de données  $n=200$ ), **ni utiliser la distance entre histogrammes** comme en TP où l'on utilise l'algorithme de clustering K-means qui n'utilise que des distances entre individus.

Comme expliqué dans [1], nous choisissons d'extraire que les caractéristiques essentielles de chaque histogramme en utilisant les descripteurs statistiques suivants (on note  $P(g)$  l'histogramme de fréquence précédente, c'est donc le pourcentage des pixels qui ont un LBP égal à  $g$ ):

$$\text{Moyenne : } m = \sum_{g=0}^{255} g * P(g)$$

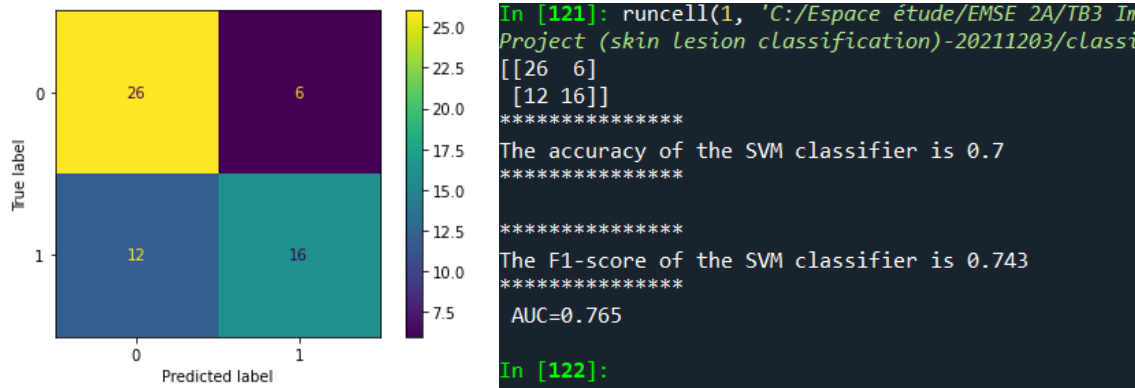
$$\text{Ecart type : } \sigma = \sqrt{\sum_{g=0}^{255} (g - m)^2 * P(g)}$$

$$\text{asymetrie : } skew = \frac{1}{\sigma^3} \sum_{g=0}^{255} (g - m)^3 * P(g)$$

$$\text{Energie : } E = \sum_{g=0}^{255} P(g)^2$$

$$\text{Entropie : } Ent = - \sum_{g=0}^{255} P(g) * \log(P(g))$$

Voici les résultats obtenus:

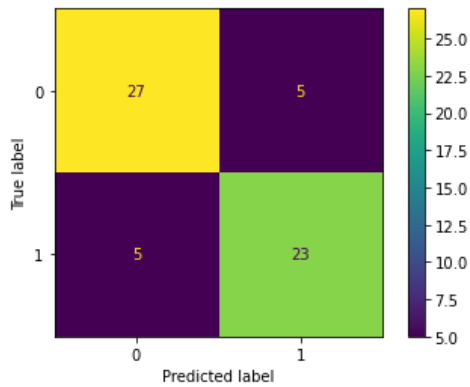


Le résultat se trouve très nettement amélioré avec l'utilisation des LBP. On a un taux de 70% de bonne prédictions et plus de 76% d'AUC.

Les 5 valeurs statistiques précédentes (moyenne, écart-type, énergie...) peuvent aussi être directement calculées à partir de l'image d'origine directement. En utilisant 2 des 3 composantes de des images d'origine (elles sont en RGB), on obtient l'histogramme  $P(g)$  des fréquences de chacun des 256 niveaux de gris (pour plus de détails voir fonction '**histogramProperties(l)**' du fichier '*project\_skin\_lesion.py*'). Les nouveaux features obtenus sont alors des **descripteurs d'intensité**.

On combinant ces nouveaux descripteurs avec les précédents on obtient un ensemble de **19 descripteurs** caractérisant aussi bien des **aspects morphologiques** que les **variations d'intensité et de texture**.

Voici le résultat obtenu.



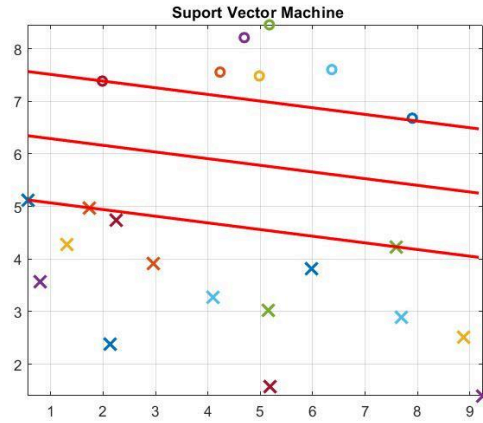
```
In [172]: runcell(1, 'C:/Espace étude/EMSE 2A/TB3 Image and patter
Project (skin lesion classification)-20211203/classifiers_skin_les
[[27  5]
 [ 5 23]]
*****
The accuracy of the SVM classifier is 0.833
*****
*****
The F1-score of the SVM classifier is 0.844
*****
AUC=0.862
In [173]:
```

Cette méthode qui combine des descripteurs de plusieurs types donne **un taux de bonnes prédictions de 83.3% largement meilleur que les autres** (plus de 10% de différence). Le F1 score ainsi que l'AUC sont aussi du même ordre (respectivement 84.4% et 86.2%). Nous retenons cette dernière approche comme la méthode finale. Certes on pourrait essayer d'optimiser le seuil mais on est dissuadé par le fait que le nombre de faux positifs et celui de faux négatifs soient égaux ( $FP=FN=5$ ): un post-traitement n'améliora pas grandement le résultat.

### Conclusion:

Ce dernier projet a permis de mettre en valeur la complémentarité des divers types de descripteurs. En effet, détecter un cancer de la peau nécessite de connaître des informations géométriques sur la zone intérêt mais aussi des aspects de texture et d'intensité. Pour améliorer de modèle on peut envisager d'augmenter le nombre de données d'apprentissage pour assurer la robustesse des algorithmes. Il pourrait aussi être intéressant d'intégrer d'autres descripteurs issus des opérateurs morphologiques d'ouverture et de fermeture.

### **ANNEXE: Détail sur les SVM**



Dans le cas linéaire, on construit un hyper plan de séparation  $\pi = w^T x + b$  de marge maximale où les variables recherchées sont  $w \in R^n$  et  $b \in R$ . La fonction de décision est alors  $f(x) = w^T x + b$ . La distance d'un point à l'hyperplan est alors  $d(x_k, \pi) = \frac{|W^T x_k + b|}{||W||} = \frac{l_k(W^T x_k + b)}{||W||}$ . **Les points vérifiant  $l_k(W^T x_k + b) > 1$  sont situés hors de la marge de largeur  $\frac{2}{||w||}$  que l'on veut maximiser.** Ce qui est équivalent à minimiser  $||w||$  ou encore  $\frac{1}{2} ||w||^2$ . On en déduit la formulation primale du problème:

$$\text{Min}(\frac{1}{2} ||w||^2) \text{ avec } \forall k \ 1 - l_k(w^T x_k + b) \leq 0$$

Le lagrangien associé est alors:

$$L(w, b, \alpha) = \frac{1}{2} ||w||^2 + \sum_{k=1}^p \alpha_k (1 - l_k(w^T x_k + b))$$

A l'optimim, l'annulation du gradient permet de déduire une formulation duale du problème:

$$\begin{cases} \text{Max}(H(\alpha)) \\ \sum_{k=1}^p l_k \alpha_k = 0 \\ \alpha_k \geq 0 \end{cases}$$

où  $H(\alpha) = -\frac{1}{2} \alpha^T A \alpha + \langle u, \alpha \rangle$  avec  $u = (1, \dots, 1)^T$  et  $[A]_{i,j} = l_i l_j \langle x_i, x_j \rangle$

A étant symétrique défini positif. En effet, on a:

$$\forall v \in R^n, \text{ on a } v^T A v = \sum_{i,j} v_i v_j \langle x_i, x_j \rangle = ||\hat{v}||^2 > 0 \text{ avec } \hat{v} = \sum_{i=1}^p v_i l_i x_i$$

Comme la matrice hessienne est  $Hess(H) = -A$ , on en déduit que H est une forme quadratique concave (d'où l'existence de son maximum). On en conclut l'**existence de la solution**. De plus, l'ensemble des contraintes est convexe (cadrant de  $R^n$ ) d'où l'**unicité**. Le problème est donc **bien posé**.

L'algorithme du gradient conjugué permet alors de trouver le point selle  $\alpha^*$

Référence:

[1] [https://www.researchgate.net/publication/4322456\\_Color\\_histogram\\_features\\_based\\_image\\_classification\\_in\\_content-based\\_image\\_retrieval\\_systems](https://www.researchgate.net/publication/4322456_Color_histogram_features_based_image_classification_in_content-based_image_retrieval_systems)