**Theory questions:**

1. By definition, accuracy is the fraction of predictions our model got right. So, it seems to be a good parameter that give a good insight of the model.

Lets get an example and calculating accuracy for the following model that classified 1000 people as having T1D (+) or not having the pathology (-):

True positive : 10

False positive : 10

True negative : 900

False negative : 80

That’s a good value for the accuracy parameter but if we see closer we will remark that :

* On the 1000 people, 910 don’t have the pathology and FN+TP=90 have it.
* On the 910 healthy people, the model correctly identifies 900 as healthy.
* On the 90 people who have the pathology, the model correctly identifies 10. **That’s mean that** **80 people (on 90) are undiagnosed!**

So, at first glance, accuracy is a good parameter (by definition) and having a high value of accuracy is important but it is not sufficient to have a good insight of a data. Calculating sensibility will have tell us how much the model is bad at predicting correctly the pathology.

So model performance (sensitivity, specificity, positive predictive value, negative predictive value, 𝐹1…) is more important.

1. Let’s study the pros and cons of the given models:

“*the first uses only BP and BMI features*”

The pros

* Avoid the curse of dimensionality
* Shorter training time
* Easier to analyse and interpret

The cons

* Lack of data
* Can lead to wrong conclusions: too little features may lead to wrong conclusions about the diagnostic (for example, happiness and focus features can show that chocolate is good for health, but adding diabetes risks and BMI shows otherwise).

“*the other one uses all of the features available*”

The pros

* No lack of data
* Higher dimensionality can improve the diagnostic, as more data is added, therefore can lead to new correlations/better results

The cons

* Curse of dimensionality
* Higher training time/higher computational time complexity
* More difficult to analyse and interpret
* Some features can be irrelevant or redundant

1. While logistic regression uses a probabilistic approach, SVM uses a geometric one which is more efficient in the case of high dimension data (here, 12 features).

Moreover, from the given information that it’s “*difficult to distinguish them from the human eye, or by just looking at the features*”, we can suppose that the data won’t be easily linearly separable, linear models probably won’t be a good approach and we will prefer **Non-Linear SVM.**

1. Logistic regression uses a probabilistic approach while SVM uses a geometric one

Logistic regression : only linear solvable pb.

Hyperparameters in SVM and LR (regularization) allow to avoid overfitting but SVM seems more efficient

We can list the main differences between LR and linear SVM:

* While LR uses a statistical and probabilistic approach (that way, it works well for already identified independent variables, and for lower amount of features studied), linear SVM uses a more geometric one (that way, works better with unstructured data in general such as biopsies,…).
* Linear SVM aims to avoid overfitting risks compared to LR.
* Linear SVM tends to reduce the risk of misclassifications in the data by solving a maximum margins optimization problem, which LR does not.

LR and Linear SVM show similar performances, but according to the features taken into account (relying more on a geometric or probabilistic model), one can be preferrable on the other.

Regarding the hyperparameters:

* Linear SVM has C as hyperparameter. C is a regularization term, showing how much we are willing to let the algorithm make mistakes (inside / on the other side of the margins). Its choice will be efficient in order to reduce the risks of overfitting/underfitting the data. The higher is C, the more we allow mistakes in the classification.
* LR also has a hyperparameter C (inverse of regularization term), controlling the importance we give to the weight coefficients of the regression. A higher regularization term means we want to penalize high coefficients (to avoid overfitting, or taking into account noises). Its choice will regularize the underfitting/overfitting problem by regularizing the bias-variance tradeoff.

We can notice that the hyperparameters aim to fill a same goal (avoid overfitting and underfitting), while LR uses a statistical approach (regularization of bias-variance tradeoff by controlling the importance to the weights of the regression), while Linear SVM uses a more geometrical approach (controlling the misclassifications allowance while building the hyperplane). Generally speaking, linear SVM is better at this task than LR.