**Robust Machine Learning**

*Explain in your own words what an adversarially robust model is and give an example of an application that might benefit from such a model and why. Be sure to explain what is meant by robust and non-robust features.*

The main problem presented in this lecture is that the distributions we are training our model on are not the same distributions we will be testing it on (post-model deployment).

Several examples can be pointed out in order to shed a light on ML brightleness: adding a random nise invisible for humans, flipping a photography..

In this lecture, we explained how sensitive could certain models be to adversarial perturbations by introducing the concept of **robustness:** robust features and non-robust features. A robust feature is a feature that tends to learn complex representation under the risk of adversarial perturbations. A non-robust feature is a robus tthat would be far more ensitvie to adversarial perturbations. Intuituvely, we could understand non-robust features as ‘human-likes’ features, drawn from human representation fo things. Robust features, on the opposite, would try to encode the computer’s representation of things and a wonderful property of a robust feature is that it would be correlated with the label even under adversarial perturbations. Non-robust features are weak for building resistance to adversary components but are very useful for generalization.

In order to create adversarially robust models, we need to re-frame the way we **train** our models. This is a framework in order to prevent our model from being completely off when deploying them. Robust models are models that are only leveraging robust features (via leveraging priors in order to tell the model which priors it should use).

During my internship, we worked on ecg recordings and a big issue when extracting features was to deal with recording noise (interferences ..). This can be interpreted as an adversarial perturbation. Should wa have constructed robust features, we might have been able to get a better training of our model.