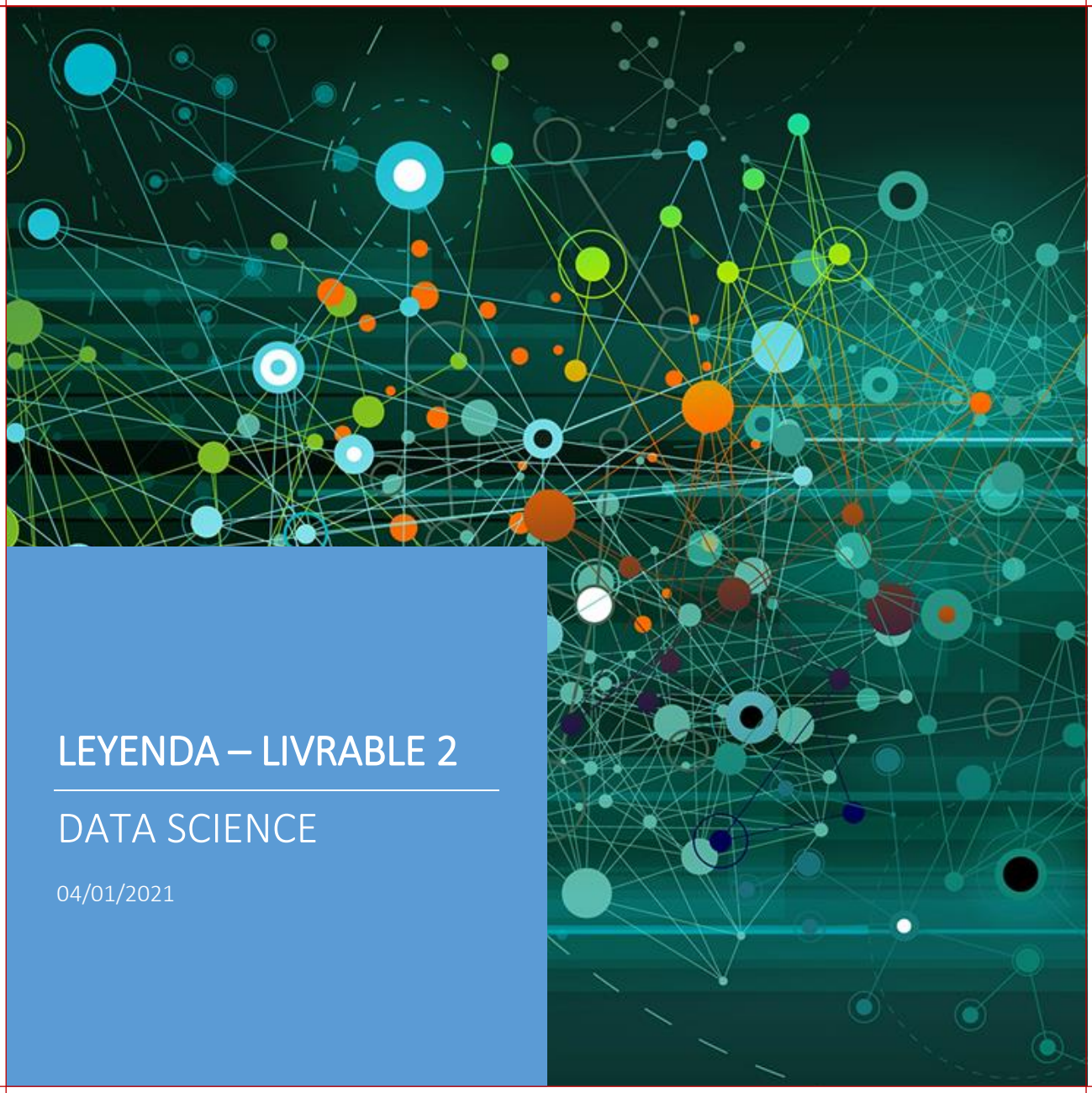


LEYENDA – LIVRABLE 2

DATA SCIENCE

04/01/2021

cesi ÉCOLE D'INGÉNIEURS

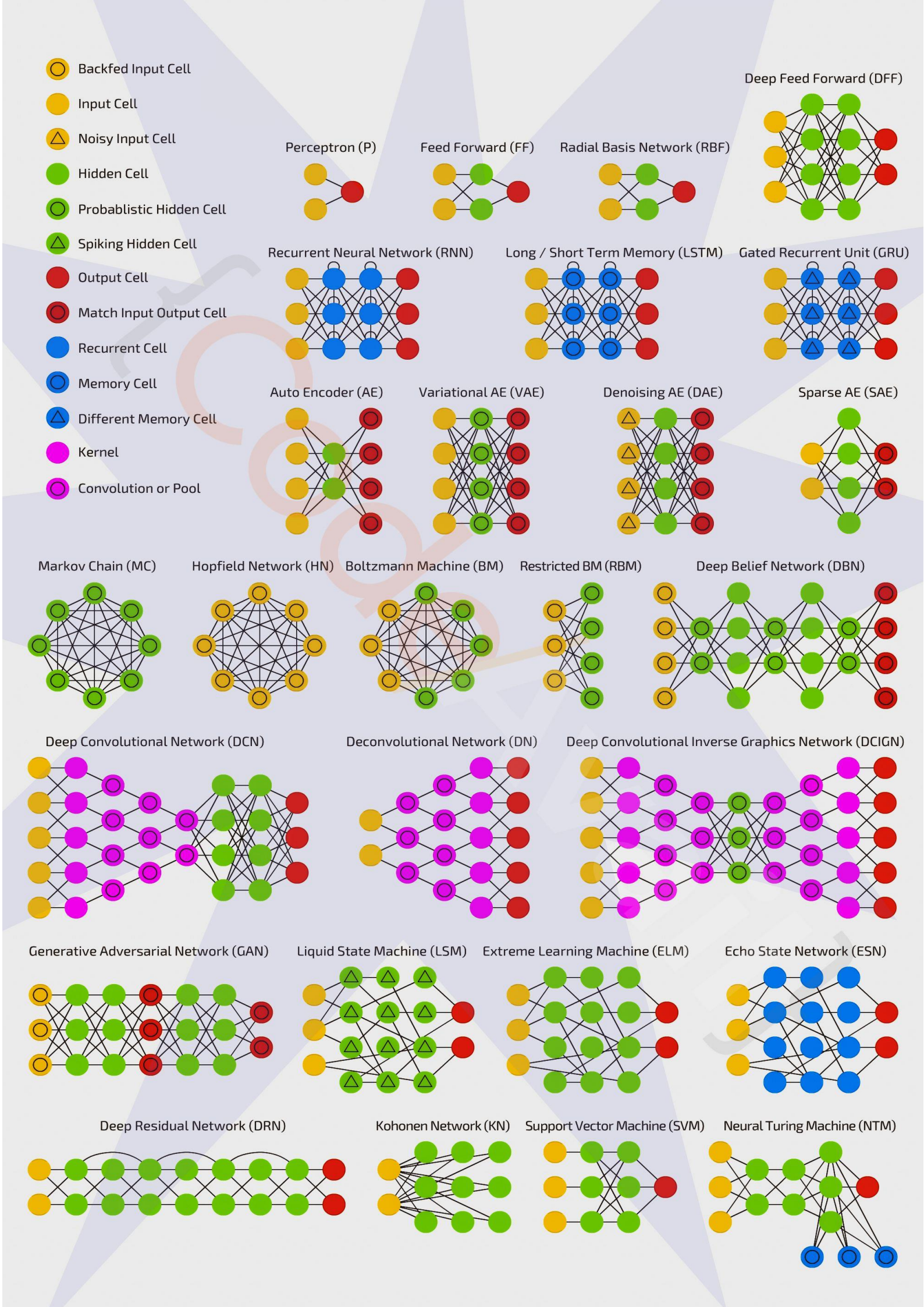
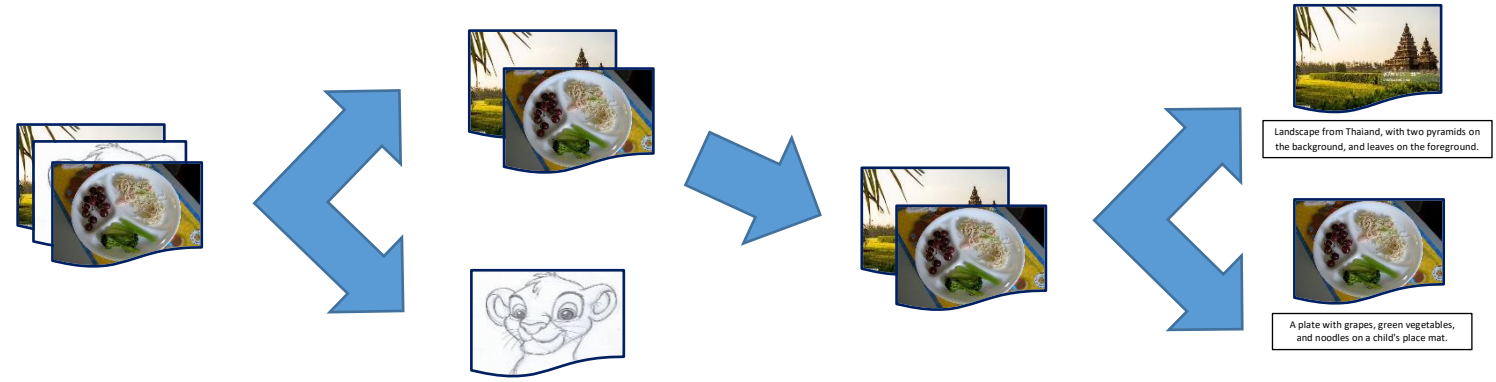

Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Perceptron (P)
Feed Forward (FF)
Radial Basis Network (RBF)
Deep Feed Forward (DFF)
Recurrent Neural Network (RNN)
Long / Short Term Memory (LSTM)
Gated Recurrent Unit (GRU)
Auto Encoder (AE)
Variational AE (VAE)
Denoising AE (DAE)
Sparse AE (SAE)
Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)
Deep Convolutional Network (DCN)
Deconvolutional Network (DN)
Deep Convolutional Inverse Graphics Network (DCIGN)
Generative Adversarial Network (GAN)
Liquid State Machine (LSM)
Extreme Learning Machine (ELM)
Echo State Network (ESN)
Deep Residual Network (DRN)
Kohonen Network (KN)
Support Vector Machine (SVM)
Neural Turing Machine (NTM)

Table des matières

1 Rappel du sujet

L'entreprise *TouNum* travaille sur la numérisation de documents (textes, images…). Leurs services sont souvent requis par des entreprises numérisant leur base de documents papier. Ils souhaitent étendre leur gamme de services pour inclure des outils de Machine Learning. En effet, certains de leurs clients ont une grande quantité de données à numériser, et un service de catégorisation automatique serait plus que valorisable.

Le workflow que vous devrez concevoir aura la forme suivante :



Chargement des images — *Classification binaire (photo/pas photo)* — *Prétraitement (denoising/sharpening…)* — *Captioning*

L'implémentation des algorithmes s'appuiera sur Python et les librairies SciKit et TensorFlow. Par ailleurs, la librairie Pandas sera utilisée dès qu'il s'agira de manipuler des dataset. ImageIO sera utile pour charger des images. Enfin, vous réutiliserez des bibliothèques de calcul avec lesquelles vous avez déjà fait connaissance, comme NumPy et MatPlotLib.

2 Livrable 2

L'entreprise voulant automatiser la sélection de photos pour l'annotations, le livrable 2 devra fournir une méthode de classification se basant sur les réseaux de neurones afin de filtrer les images qui ne sont pas des photos du dataset de départ.

Le livrable sera sous la forme notebook Jupyter et devra, pour être validé, intégrer :

1. Le code TensorFlow ainsi qu'un schéma de l'architecture du réseau de neurones. Toutes les parties doivent être détaillée dans le notebook : les paramètre du réseau, la fonction de perte ainsi que l'algorithme d'optimisation utilisé pour l'entrainement.
2. Un graphique contenant l'évolution de l'erreur d'entrainement ainsi que de l'erreur de test et l'évolution de l'accuracy pour ces deux datasets.
3. L'analyse de ces résultats, notamment le compromis entre biais et variance (ou sur-apprentissage et sous-apprentissage).
4. Une description des méthodes potentiellement utilisables pour améliorer les compromis biais/variance : technique de régularisation, drop out, early-stopping, …

3 Dataset

Plusieurs dataset sont à votre disposition. On a :

- Des peintures
- Des Schéma et graphes
- Des portraits dessinés en noir et blanc
- Des images de textes scannés
- Des photos

Les images ne sont pas étiquetées, mais ce n'est pas un problème puisqu'elles sont réparties dans des archives différentes.

Le but ultime est d'être capable de distinguer les photos parmi toutes ces images. Il est tout de même conseillé de commencer par les images les plus faciles à distinguer des photos, puis aller vers les dataset les plus difficiles à classifier (notamment, il y a dans le dataset peinture un certain nombre d'œuvres au rendu assez réaliste, qui devraient vous poser problème).

Ce livrable est à fournir pour le 18/01/2021