

# Script d'Explication de la Présentation

Généré par Gemini

10 décembre 2025

## Slide 1 : Page de Titre

"Bonjour à tous. Cette présentation expose notre projet sur l'application de l'intelligence artificielle en mécanique, avec un focus sur la maintenance prédictive. Ce projet a été réalisé par les étudiants de l'École Nationale des Sciences Appliquées de Fès."

## Slide 2 : Plan de la présentation

"Voici le plan que nous allons suivre. Nous commencerons par une introduction générale, puis nous définirons le contexte et les objectifs du projet. Ensuite, nous détaillerons la méthodologie utilisée, nous présenterons et analyserons les résultats obtenus, et nous terminerons par une conclusion."

## Slide 3 : Introduction (Titre)

"Commençons donc par l'introduction."

## Slide 4 : Introduction (Contenu)

"Notre projet s'inscrit dans le contexte de l'Industrie 4.0, où la mécanique et l'intelligence artificielle se rencontrent. Le sujet précis est la maintenance prédictive des machines industrielles. Notre objectif est d'anticiper les pannes pour optimiser la production et réduire les coûts. Pour ce faire, notre approche a été d'utiliser des réseaux de neurones artificiels (RNA) sous l'environnement MATLAB."

## Slide 5 : Contexte et Objectifs (Titre)

"Abordons maintenant plus en détail le contexte et les objectifs."

## Slide 6 : Problématique

"Les approches de maintenance traditionnelles ont leurs limites. La maintenance 'Corrective' est réactive, elle n'a lieu qu'après la panne, ce qui engendre des coûts d'arrêt élevés et imprévisibles. La maintenance 'Préventive' est planifiée à des intervalles fixes, ce qui

peut conduire à un gaspillage en remplaçant des pièces encore fonctionnelles. La question clé est donc : comment intervenir au bon moment ?"

## Slide 7 : Objectif du Projet

"Notre projet est de développer un système de 'Maintenance Prédictive Intelligent'. La mission est de concevoir et d'entraîner un modèle de réseau de neurones capable de classifier l'état de la machine en temps réel. Il doit pouvoir faire la différence entre un 'Fonctionnement Normal' (codé par 0) et une 'Panne Détectée' (codé par 1), en se basant sur les données des capteurs comme la température, la vitesse, le couple, etc."

## Slide 8 : Méthodologie (Titre)

"Voyons maintenant la méthodologie que nous avons suivie."

## Slide 9 : Description des Données

"Nous avons travaillé avec un jeu de données nommé 'machine\_failure.csv'. Les variables d'entrée, aussi appelées 'features', sont : le type de produit, la température de l'air, la température du processus, la vitesse de rotation, le couple et l'usure de l'outil. La variable que nous cherchons à prédire, notre 'cible', est 'Machine Failure', qui indique si une panne a eu lieu (1) ou non (0)."

## Slide 10 : Implémentation sous MATLAB

"Nous avons utilisé MATLAB et son outil 'Neural Network Toolbox'. Les étapes clés ont été : 1. La **Préparation** : charger les données, convertir les variables textuelles en numériques. 2. L'**Architecture** : nous avons choisi un réseau de reconnaissance de formes ('patternnet') avec deux couches cachées de 100 et 50 neurones. 3. L'**Entraînement** : nous avons réparti les données (70

## Slide 11 : Architecture du Réseau

"Cette image est une capture d'écran de l'interface d'entraînement de MATLAB. Elle montre que l'entraînement s'est bien déroulé et s'est arrêté automatiquement après 43 cycles ('epochs') car les critères de performance sur les données de validation étaient atteints."

## Slide 12 : Résultats et Analyse (Titre)

"Passons maintenant aux résultats et à leur analyse."

## Slide 13 : Performance Globale

"Le modèle atteint une excellente précision globale de 97.5%. - La **Classe 0 (Normal)** est très bien détectée, avec une précision de 98- La **Classe 1 (Panne)** est plus complexe à identifier. La précision est de 69.2% et le rappel de 38.3%. Cela est dû au fait qu'il y a beaucoup moins d'exemples de pannes que de fonctionnement normal dans nos données, c'est ce qu'on appelle un déséquilibre des classes."

## Slide 14 : Matrices de Confusion

"Ces graphiques, appelés matrices de confusion, montrent en détail la performance pour chaque classe. Elles nous aident à voir précisément où le modèle se trompe. Par exemple, pour l'ensemble de test, on voit que sur toutes les pannes réelles, le modèle en a correctement identifié 38.3%."

## Slide 15 : Courbes ROC

"La courbe ROC est un autre outil pour évaluer la capacité du modèle à distinguer les deux classes. Idéalement, la courbe doit se rapprocher le plus possible du coin supérieur gauche. Nos résultats montrent une excellente performance, ce qui confirme la fiabilité du modèle."

## Slide 16 : Validation et Erreur

"Ces deux figures valident notre modèle. - À gauche, le graphique de 'Cross-Entropy' montre que l'erreur du modèle a diminué pendant l'entraînement et que celui-ci s'est arrêté au bon moment (époque 37) pour éviter le sur-apprentissage. - À droite, l'histogramme des erreurs montre que la plupart des erreurs sont très proches de zéro, ce qui est un excellent signe."

## Slide 17 : Conclusion (Titre)

"Pour finir, voici notre conclusion."

## Slide 18 : Conclusion (Contenu)

"En résumé, nous avons mené une réussite technique en mettant en place un réseau de neurones performant avec MATLAB, atteignant 97.5% de précision. Nous avons ainsi validé notre preuve de concept pour la maintenance prédictive à partir de données de capteurs. Comme perspectives, nous pourrions chercher à améliorer la détection des pannes (le 'rappel'), par exemple en gérant le déséquilibre des données. On pourrait aussi tester d'autres architectures comme le Deep Learning, et enfin, déployer ce système en temps réel."

## **Slide 19 : Merci**

"Merci de votre attention. Nous sommes maintenant prêts à répondre à vos questions."