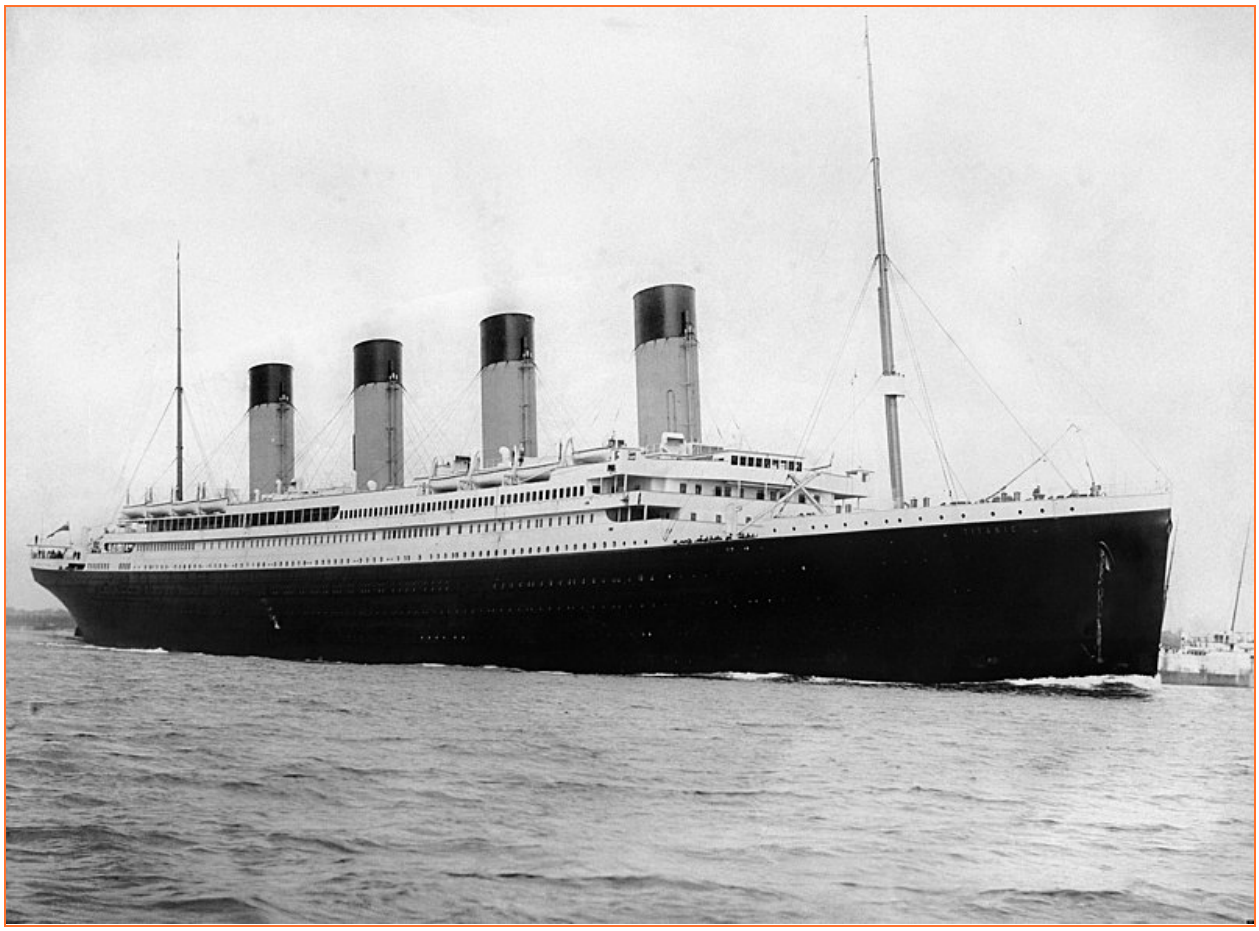

Cahier des charges

L2I1 : Machine Learning from disaster

—

Hongxiang Lin
Melissa Merabet
Mathieu Antonopoulos
Timothé Miel
20/02/2023

Université Paris Cité

Sommaire

1. Introduction

Le projet vise à construire un modèle prédictif par Machine Learning, en utilisant les données relatives aux passagers du *Titanic*, afin de prédire qui aurait le plus de chances de survivre au naufrage. Le projet se base sur un ensemble de données, '*train.csv*', contenant les informations sur les passagers tels que le nom, l'âge, le sex, la classe socio-économique, etc. Les outils mathématiques tels que *matplotlib* et *seaborn* seront utilisés pour analyser les résultats et déterminer le modèle le plus efficace. Le produit final sera un modèle prédictif codé en *Python*, accompagné d'une documentation détaillée, explicitant le fonctionnement du modèle et les analyses effectuées, ainsi qu'une interface web (optionnelle) pour une visualisation interactive des résultats. Le but est également d'obtenir le meilleur classement possible dans le concours *Kaggle* 'Titanic: Machine Learning from Disaster'.

2. Concepts de base

Dans ce projet, nous allons utiliser des ensembles de données similaires qui comprennent des informations sur les passagers comme le nom, l'âge, le sex, la classe socio-économique, la situation familiale, le numéro de cabine, le prix du ticket, le port d'embarquement, etc.

Un dataset est intitulé 'train.csv' et l'autre est intitulé 'test.csv'.

- 'train.csv' contiendra les détails d'un sous-ensemble de passagers à bord (891 pour être exact) et surtout, révélera s'ils ont survécu ou non.
- 'test.csv' permettra d'évaluer les performances du modèle.

En utilisant les modèles que nous avons trouvés dans les données train.csv, nous allons prédire si les 418 autres passagers à bord (trouvés dans test.csv) ont survécu.

Lors de l'analyse des données, nous allons utiliser divers outils mathématiques tels que matplotlib et seaborn afin d'analyser les résultats et mieux déterminer le modèle que nous utiliserons pour prévoir la survie des passagers à bord.

Enfin, nous utiliserons Streamlit pour construire une application web permettant de mieux visualiser les résultats de notre analyse de données.

Notre projet sera développé avec les outils suivants :

- Jupyter Notebook (avec Python dans l'environnement d'Anaconda 3)
- Serveur SVN (via le logiciel client Tortoise SVN.)

- Numpy
- Pandas (data management) : https://pandas.pydata.org/
- Seaborn (https://seaborn.pydata.org/examples/index.html)
- Matplotlib (https://matplotlib.org/stable/gallery/index)
- Les différents modèles de Machine Learning : Linear Models, etc.
- Streamlit (https://streamlit.io/)
- Plotly (Plotly: Low-Code Data App Development)
- Python (version 3.9)
- Scikit-learn pour le Machine Learning. (https://scikit-learn.org/stable/index.html)

3. Contexte

Le naufrage du Titanic est l'un des naufrages les plus tristement célèbres de l'histoire. Le 15 avril 1912, lors de son voyage inaugural, le RMS Titanic, largement considéré comme "insubmersible", a coulé après avoir heurté un iceberg. Malheureusement, il n'y avait pas assez de canots de sauvetage pour tout le monde à bord, ce qui a entraîné la mort de 1502 des 2224 passagers et membres d'équipage. Bien qu'il y ait eu une part de chance dans la survie, il semble que certains groupes de personnes aient eu plus de chances de survivre que d'autres.

Dans ce défi, notre équipe va construire un modèle prédictif qui répond à la question suivante : "Quelles sont les personnes qui ont le plus de chances de survivre ?" en utilisant les données des passagers (nom, âge, sexe, classe socio-économique, etc.).

4. Description de la demande

I. Les objectifs

Voici la liste des objectifs fixés :

- Élaborer un modèle de prédiction par Machine Learning en python.
- Obtenir le meilleur classement possible au concours *Kaggle : Titanic: Machine Learning from Disaster.*
- Trouver les facteurs les plus influents sur les chances de survie des passagers.
- Fournir une documentation retraçant les conjectures et hypothèses émises par le groupe afin d'élaborer le modèle de prédiction. La documentation devra aussi expliciter le fonctionnement du modèle et fournir une analyse des données d'entraînement.
- Permettre une visualisation simple et accessible des données à travers une interface web (optionnel).
- Permettre à l'utilisateur de comprendre les résultats obtenus en exposant certaines données et analyses sur l'interface.

II. Produit du projet

Le produit est un modèle prédictif codé en python. Ce modèle sera accompagné d'une documentation explicitant le fonctionnement du modèle. Une interface web (optionnelle) permettra la visualisation intéractive des résultats. L'interface sera réalisée avec l'outil *'Streamlit'*.

III. Fonctions du produit

Le modèle doit permettre de répondre à la question suivante : « A l'aide des données relatives aux passagers au Titanic (nom, âge, sexe, classe socio-économique, etc.), qui a le plus de chance de survivre ? ». Ainsi, le produit doit être en mesure de prédire la survie ou non d'un passager à partir de ses données avec un taux de précision maximum. Il retournera donc une liste et complètera pour chaque passagers la colonne 'survie' avec '1' si le passager survit et '0' sinon.

La documentation retrace les étapes de conception du modèle en s'appuyant sur les analyses, hypothèses et résultats obtenus tout au long du développement. Elle soulignera les résultats obtenus afin de déterminer les facteurs influents.

(*optionnel*) De plus, le produit doit permettre la visualisation interactive des données et résultats obtenus à travers une interface simple et visuelle. L'application permet notamment d'exposer les facteurs de survie et de relever les données pertinentes à travers un support intuitif (*streamlit*). De plus, des représentations graphiques expliciteront les résultats et données de façon visuelle.

IV. Critères d'acceptabilité et de réception

Le modèle prédictif doit être soumis à l'ensemble des règles de compétition «*Kaggle*». Ainsi on mesurera la qualité du modèle prédictif avec le score « *Kaggle* » obtenu pour le produit final. L'un des critères de réussite est donc le classement obtenu. (relatif au nombre de participants)

Concernant la documentation, elle doit permettre de comprendre les étapes de conception du modèle. Elle doit aussi fournir les indications nécessaires à l'utilisation complète du modèle et de son interface. Enfin elle doit permettre la compréhension des données et de leur analyse.

(*optionnel*) L'interface doit être simple et lisible. Elle doit permettre à toute personne la compréhension des résultats obtenus. De plus, le choix de support Web permet de consulter l'application depuis tous les supports sans contraintes matérielles, avec une simple connexion internet.

5. Contraintes

I. Contraintes de coûts

Le projet est principalement coûteux sur le plan humain : une équipe de quatre développeurs travaillant sur 12 semaines. Pour ce qui est de l'échelle de données, cette dernière étant relativement basse (vu que nous travaillons sur une base de données de 891 lignes), l'hébergement d'un serveur n'est pas nécessaire : l'utilisation de nos ordinateurs personnels est suffisante. De ce fait aucune contrainte de coût majeure n'est imposée.

II. Contraintes de délais

Le projet doit être mené à bien en 12 semaines. Nous aurons différents documents à rendre tout le long du projet et ce, à différentes dates :

Dans un premier temps, le présent cahier des charges à rendre en semaine 3, s'en suivra du cahier de recette la semaine d'après (s. 4). La 5e semaine, les conceptions, générale et détaillée, seront à rendre. Les manuels d'utilisation et d'installation, le plan des tests, la documentation interne du code et le code source du programme seront tous à rendre pour la 11e semaine. Enfin, avant la soutenance, le rapport du projet, le résumé et les diapositives sonorisées seront également à rendre.

III. Contraintes matérielles

Dans ce projet les contraintes matérielles restent minimes. Nous avons 2 documents CSV (test.csv et train.csv) de ce fait le temps de calcul est négligeable. L'ensemble du projet (hors Web conception) se faisant sous Jupyter et par conséquent sous Python, nous utiliserons Anaconda afin de le réaliser. L'hébergement du projet sera donc sur chaque ordinateur personnel. L'utilisation d'un serveur SVN (fourni par l'université) afin de permettre une meilleure synchronisation entre les développeurs sera de rigueur ; l'exploitation du serveur peut se faire via le terminal de l'ordinateur ou via des clients graphiques (tels que RapidSVN ou TortoiseSVN). Afin de réaliser l'interface web, nous utiliserons Dash Enterprise, une galerie d'app web qui nous permettra de visualiser notre projet plus clairement.

6. Déroulement du projet

I. Planification

Pour réussir notre projet, nous l'avons divisé sur 3 parties :

1. **L'analyse des données en deux temps (2-4 semaines) :**

   - L'analyse de la qualité des variables : dans cette partie, nous allons analyser les données en termes de disponibilité et complétude c'est-à-dire l'accès facile et rapide aux données et repérer les informations manquantes. Dans notre base de données, nous avons remarqué qu'il manque 177 lignes dans la colonne 'Age' qui représente 20% des données, et nous avons mappé les valeurs des colonnes 'Embarked', 'Sex' et 'Pclass' pour faciliter la migration des données.
   - L'analyse des facteurs : cela consiste à explorer les données, trouver les liens pouvant exister entre les différentes variables de notre base de données (Age, Sex, Pclass, Embarked...) et déduire les informations statistiques qui permettront d'optimiser le modèle prédictif.

2. **La construction d'un modèle prédictif (4-6 semaines) :**

   Nous allons choisir parmi les différents modèles de machine Learning celui qui permettra de modéliser au mieux les survivants. Nous visons avec notre encadrant, un taux qui dépasse 89% de réussite.

3. **Application Web (2-4 semaines) (Optionnel) :**

   Nous allons développer une application web qui permettra de mettre en profit notre travail sur la prédiction des survivants après avoir renseigné les informations nécessaires. Cette partie est optionnelle.

II. Ressources

Les ressources humaines : Les ressources humaines de ce projet sont 4 étudiants en 2ème année de licence informatique/mathématique et informatique ainsi que M. Paul Boniol, notre encadrant.

Les ressources matérielles : nous disposons de nos ordinateurs personnels. Nous allons coder avec le langage *Python* (version 3.9) sur *Jupyter* et utiliserons *Pandas, seaborn* et *matplotlib* pour visualiser les données. Nous allons également nous servir de différentes ressources recommandées par notre encadrant, *Kaggle*, pour s'inspirer des différents projets portant sur le même thème, *'Streamlit'* pour réaliser l'interface web et enfin *'Scikit-Learn'* pour ce qui concerne les modèles de Machine Learning.