

Université Paris Cité

Dashboard – Analyse de données

Programmation Web

Bastien Hottelet
Décembre 2024

Table des matières

Introduction	3
Objectifs de l'étude	3
Description du jeu de données Titanic.....	3
Exploration initiale des données.....	4
Processus de chargement via l'interface	4
Aperçu des données : Premières observations.....	5
Détection et gestion des colonnes constantes.....	6
Préparation des données	7
Reconnaissance des types de variables	7
Catégorisation des colonnes : numérique, facteur, texte	7
Gestion des valeurs manquantes.....	8
Identification des colonnes concernées	8
Gestion des valeurs aberrantes	9
Méthodes utilisées (Winsorisation, suppression).....	9
Visualisations (boxplots) pour évaluer les impacts	9
Facteurs à haute cardinalité	10
Analyse des colonnes concernées	10
Déséquilibre des classes	11
Analyse de la distribution de la variable cible (survie)	11
Normalisation des données	12
Techniques de normalisation appliquées	12
Analyse exploratoire des données (EDA)	13
Analyse univariée	13
Statistiques descriptives et visualisations de chaque variable	13
Analyse bivariée	18
Relation entre variables clés (ex. survie et sexe, âge, classe)	18
Analyse des corrélations	22
Modélisation prédictive	24

Modèles testés : Random Forest, SVM, Régression logistique.....	24
Amélioration des hyperparamètres & Comparaison des modèles	26
Conclusion et perspectives	28
Suggestions d'amélioration pour l'interface ou l'analyse	28

Introduction

Objectifs de l'étude

L'analyse de données est une étape cruciale dans tout projet que pourrais mener un ingénieur en machine learning. Cette étape permet à l'ingénieur de comprendre les données mises à sa disposition en identifiant leurs forces et leurs faiblesses (valeurs manquantes, valeurs aberrantes, déséquilibre de classes, etc.)

Ce projet consistait à implémenter une interface qui permet d'importer n'importe quel jeu de données, le prétraiter et faire son analyse voir même faire des premières modélisation avec des algorithmes simples de classification supervisée comme « Random Forest ».

Dans le cadre de cette étude, nous utiliserons le jeu de données **Titanic**, un classique en science des données, pour mener une étude complète à l'aide de l'interface produite. Ce dataset est par ailleurs proposé par défaut sur l'interface pour tout utilisateur qui voudrait tester le fonctionnement de celle-ci avant d'y mettre ses propres données.

Cette étude vise à démontrer les fonctionnalités de l'interface tout en répondant à une problématique clé : comprendre les facteurs influençant la survie des passagers du Titanic.

Description du jeu de données Titanic

Le jeu de données Titanic est un ensemble bien connu en machine learning, contenant des informations sur les passagers du célèbre paquebot qui a fait naufrage en 1912. Les données incluent :

- **Des informations démographiques** : Âge, sexe, etc.
- **Des données contextuelles** : Type de billet, présence de membres de la famille à bord.
- **La variable cible** : Survie (Normalement 1 ou 0, transformé en Yes/No ici).

Cette richesse en variables en fait un excellent candidat pour illustrer les fonctionnalités de l'application et explorer une question centrale : quels facteurs ont influencé les chances de survie des passagers ?

Exploration initiale des données

Processus de chargement via l'interface

L'interface possède deux options de chargement de données :

- Chargement de CSV externe
 - Case à cocher pour préciser la présence ou non d'un Header
 - Séparateur et Caractère de Guillemet
- Chargement des données Titanic intégrées à l'application
 - Transformation appliquée à la colonne « Survived » pour remplacer les facteurs 0 / 1 par No / Yes pour une meilleur lisibilité.

Pour cette étude on utilisera donc les données Titanic préchargées dans l'application.

Step 1: Load Data

Upload a CSV file or use the built-in Titanic dataset. Ensure the file format and separators are correct.

Upload CSV File

Browse... No file selected

Load CSV Use Titanic

☒ Has Header?

Separator

,

Quote

"

Figure 1 – Interface de chargement des données

Aperçu des données : Premières observations

Une fois les données Titanic chargées, nous pouvons consulter une visualisation des 10 premières lignes pour s'assurer que les données sont bien celles que nous attendions.

Cela permet de valider les données et de se rendre compte de certaines potentielles incohérence très tôt.

Step 2: Data Preview

Preview of the first 10 rows of: Titanic Dataset

Search:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	No	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171	7.25		S
2	Yes	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	0	PC 17599	71.2833	C85	C
3	Yes	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 3101282	7.925		S
4	Yes	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803	53.1	C123	S
5	No	3	Allen, Mr. William Henry	male	35	0	0	373450	8.05		S
6	No	3	Moran, Mr. James	male		0	0	330877	8.4583		Q
7	No	1	McCarthy, Mr. Timothy J	male	54	0	0	17463	51.8625	E46	S
8	No	3	Palsson, Master. Gosta Leonard	male	2	3	1	349909	21.075		S
9	Yes	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27	0	2	347742	11.1333		S
10	Yes	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14	1	0	237736	30.0708		C

Showing 1 to 10 of 10 entries

Proceed

Figure 2 - Visualisation du jeu de données importé

Détection et gestion des colonnes constantes

Une étape invisible ici pour l'utilisateur et que le jeu de données Titanic ne déclenche pas est la suppression des colonnes qui ont une valeur constante. Dans certains jeux de données problématiques, une colonne peut avoir la même valeur sur chaque ligne, ce qui est purement inutile et néfaste pour la modélisation.

Dans de tels cas, la ou les colonnes en question sont retirés automatiquement et l'utilisateur est notifié du changement et de la raison.

Exemple sur le jeu de données « **IBM HR Analytics Employee Attrition & Performance** » :

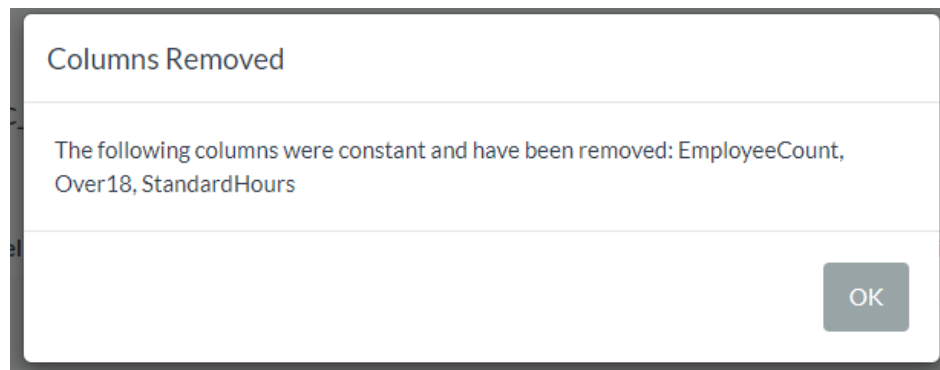


Figure 3 - Gestion des colonnes constantes

Préparation des données

Reconnaissance des types de variables

Catégorisation des colonnes : numérique, facteur, texte

Une fois les données validées, on propose à l'utilisateur de corriger la catégorisation des colonnes. En effet il est parfois difficile de classer des données dans les catégories numérique, catégorielles, textuelles.

The screenshot shows a web interface titled "Step 3: Variable Recognition". Below the title is a subtitle: "Review the proposed data types and adjust as necessary." The interface contains a grid of 12 dropdown menus, each for a different variable. The variables and their current selected types are: PassengerId (numeric), Survived (factor), Pclass (numeric, highlighted with a red box), Name (character), Sex (factor), Age (numeric), SibSp (numeric), Parch (numeric), Ticket (character), Fare (numeric), Cabin (factor), and Embarked (factor). At the bottom left of the grid is a dark red button labeled "Confirm & Proceed".

Variable	Proposed Type
PassengerId	numeric
Survived	factor
Pclass	numeric
Name	character
Sex	factor
Age	numeric
SibSp	numeric
Parch	numeric
Ticket	character
Fare	numeric
Cabin	factor
Embarked	factor

Figure 4 - Correction de la catégorisation des colonnes

Dans notre cas on peut notamment voir que la colonne « Pclass » pour « Passenger Class » est détectée comme numérique alors qu'elle est en réalité catégorielle avec des classes allant de 1 à 3.

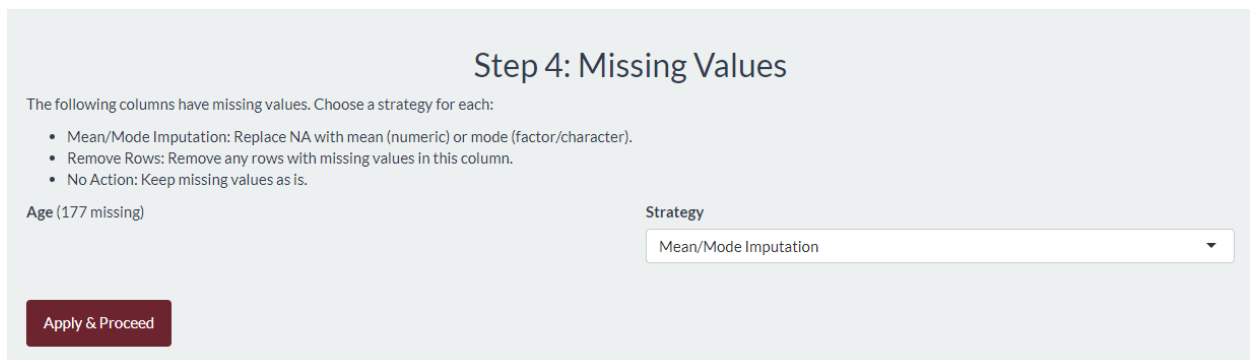
Nous pouvons donc grâce à l'interface corriger cela est ensuite confirmer le changement pour se rendre à la prochaine étape.

Gestion des valeurs manquantes

Identification des colonnes concernées

Une fois nos colonnes catégorisées, nous passons à la détection des valeurs manquantes, pour gérer ce problème dans les données, trois sélections sont proposés à l'utilisateur :

- No Action : Laisse les valeurs vides, aucun changement réalisé par l'interface.
- Remove Rows : Enlève les lignes pour lesquelles les valeurs sont vides
- Mean/Mode Imputation : Remplace les valeurs par :
 - La moyenne pour les valeurs numériques
 - La valeur la plus fréquente pour les valeurs catégorielles / textuelles



Step 4: Missing Values

The following columns have missing values. Choose a strategy for each:

- Mean/Mode Imputation: Replace NA with mean (numeric) or mode (factor/character).
- Remove Rows: Remove any rows with missing values in this column.
- No Action: Keep missing values as is.

Age (177 missing)

Strategy

Mean/Mode Imputation

Apply & Proceed

Figure 5 - Gestion des valeurs manquantes

Ici nous utiliseront la méthode « **Mean/Mode Imputation** » pour la variable « **Age** » pour laquelle 177 valeurs sont manquantes.

Gestion des valeurs aberrantes

Méthodes utilisées (Winsorisation, suppression)

Pour gérer les valeurs aberrantes, on propose encore à l'utilisateur trois solutions :

- No Action : Les valeurs aberrantes restent inchangées
- Winsorize : Remplace les valeurs extrêmes par des seuils définis :
 - o Valeurs basses : 5% centile
 - o Valeurs hautes : 95% centile
- IQR : Identifie et supprime les valeurs aberrantes en fonction de l'écart interquartile
 - o Valeurs basses : $Q1 - 1.5 * IQR$
 - o Valeurs hautes : $Q3 + 1.5 * IQR$

Visualisations (boxplots) pour évaluer les impacts

L'interface permet d'afficher un graphique Boxplot pour pouvoir visualiser la quantité de valeurs aberrantes et permettre à l'utilisateur de mieux choisir le traitement adapté.

La gestion des valeurs aberrantes est donc faite variable par variable.

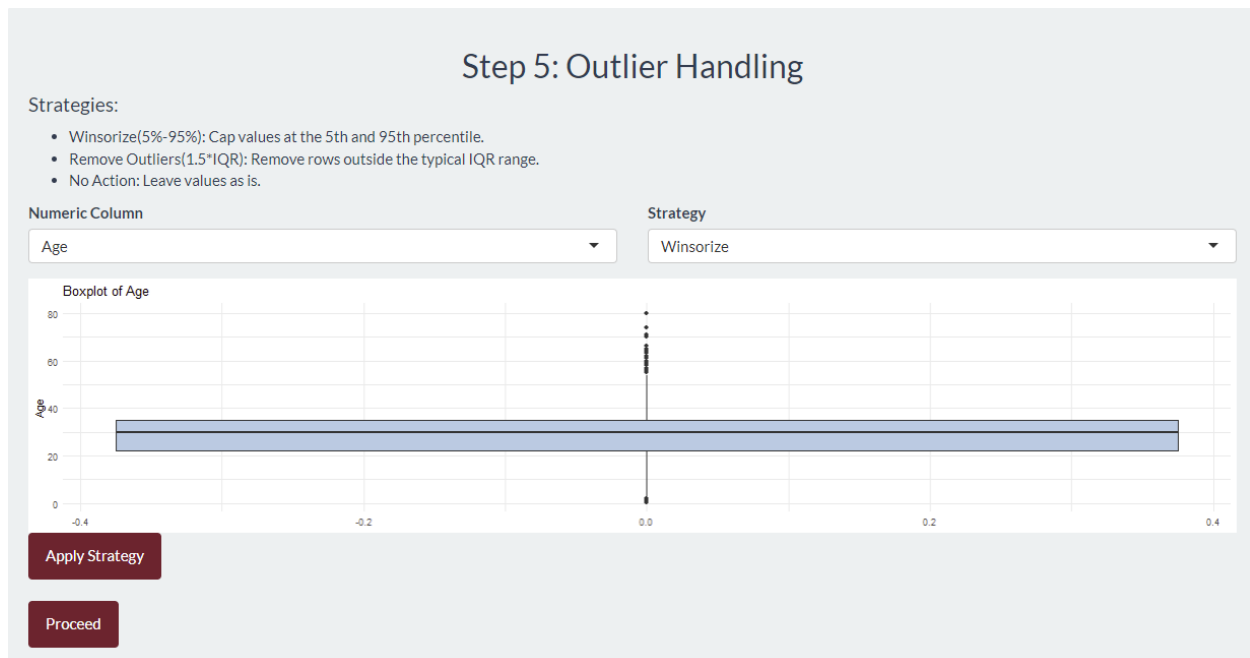


Figure 6 - Gestion des valeurs aberrantes

Dans notre cas, nous n'appliquerons **aucun traitement** pour les valeurs aberrantes étant donné que les âges extrêmement peuvent avoir une importance significative pour la survie des passagers.

Facteurs à haute cardinalité

Analyse des colonnes concernées

Pour des méthodes comme « Random Forest » les variables catégorielles ne peuvent pas avoir trop de modalités, pour cela on a implémenté une analyse de variables catégorielles à haute cardinalité.

On propose alors à l'utilisateur de ne rien faire, de retirer la colonne ou de la convertir en caractères.

Step 6: High Cardinality Factors

These factor columns have more than 20 levels. Choose a strategy:

- Drop Column: Remove the column entirely.
- Convert to Character: Treat it as a text feature.
- Do Nothing: Leave as is.

Cabin (148 levels)

Strategy
Drop Column ▼

Apply & Proceed

Dans notre cas on **retire la colonne** étant donné qu'elle n'apporte pas d'informations très importantes et que ses 148 niveaux seront impossible à traiter en tant que facteur.

Déséquilibre des classes

Analyse de la distribution de la variable cible (survie)

Si nous sommes dans le cadre d'une classification, nous pouvons choisir notre variable cible et analyser sa répartition.

Ensuite nous pouvons prendre la décision de faire l'action suivante : Suréchantillonner la minorité ou sous-échantillonner la majorité ou ne rien faire.

Step 7: Class Imbalance

If you plan classification, select a factor target. Then view distribution and choose a balancing strategy.

Target Variable (for Classification)

Survived

Class distribution for: Survived

Class	Count
No	~450
Yes	~350

Balancing strategies:

Strategy

Oversample Minority

Oversample: duplicate minority classes. Undersample: remove majority samples.

Apply & Proceed

Dans notre cas, nous allons **suréchantillonner la minorité** étant donné que l'écart n'est pas trop grand cela ne devrait pas poser de problème.

A noter tout de même que sur des jeux de données où le déséquilibre est conséquent (ex. 99% contre 1%), il serait judicieux d'éviter ce genre d'opération qui mènerait probablement au surajustement sur les exemples de la classe minoritaire.

Normalisation des données

Techniques de normalisation appliquées

Pour normaliser les données, trois options sont proposées à l'utilisateur :

- **No Action** : Aucune transformation n'est appliquée.
- **Standard (z-score)** : Centre et réduit les données en utilisant la moyenne et l'écart-type :
 - Formule : $z = \frac{x - \mu}{\sigma}$
 - μ : moyenne des données
 - σ : écart-type des données
- **Min-Max [0-1]** : Transforme les données pour qu'elles soient comprises entre 0 et 1 :
 - Formule : $x_{\text{norm}} = \frac{x - \min(x)}{\max(x) - \min(x)}$

Dans le cadre de la prédiction des survivants du Titanic, les extrémités en termes d'âges aurait tendance à ne pas avoir les mêmes chances de survie que le reste.

Pour cette raison nous allons utiliser la normalisation standard (z-score), cette méthode nous permettra de centrer les données autour de leur moyenne et les réduire en fonction de l'écart type, ce qui conserve les écarts extrêmes mais dans une échelle plus uniforme.

Step 8: Feature Engineering (Normalization)

Choose a normalization method for each numeric variable:

- Standard (Z-score): $(x - \text{mean})/\text{sd}$
- Min-Max [0,1]: $(x - \min)/(\max - \min)$
- None: leave as is.

PassengerId
None

Age
Standard (Z-score)

SibSp
Standard (Z-score)

Parch
Standard (Z-score)

Fare
Standard (Z-score)

Apply & Proceed

Figure 7 - Normalisation des données

Analyse exploratoire des données (EDA)

Analyse univariée

Statistiques descriptives et visualisations de chaque variable

Pour commencer cette analyse univariée, nous allons commencer par vérifier que le sur-échantillonnage de la classe minoritaire de la colonne « Survived » a bien eu lieu :

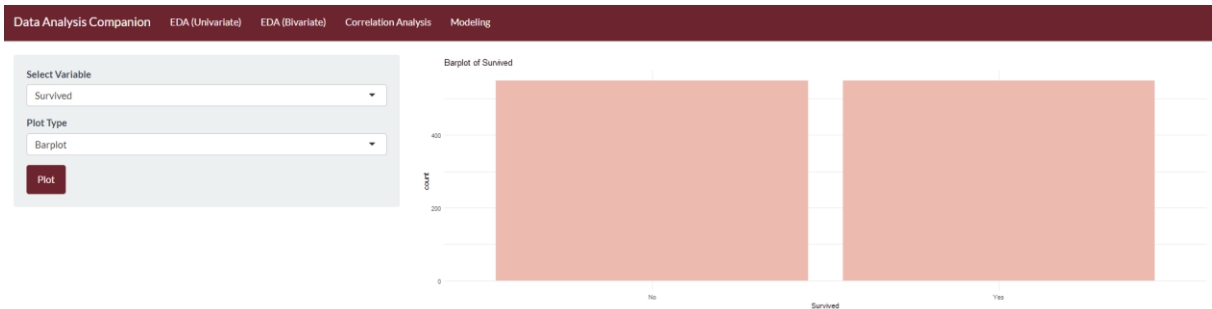


Figure 8 - Vérification du sur-échantillonnage de la classe minoritaire

Nous pouvons voir que les deux classes ont le même nombre d'exemples ce qui nous permet de valider que nos opérations précédentes se sont bien déroulées.

Pour regarder les autres classes, à un but d'interprétation pure et pas de visualisation des données pré-entraînement, nous importerons le jeu de données sans réaliser de prétraitement en terme de normalisation et de sur-échantillonnage.

Ensuite regardons la colonne « Pclass » qui contient les informations relatives aux classes socio-économique des passagers en quelques sortes :

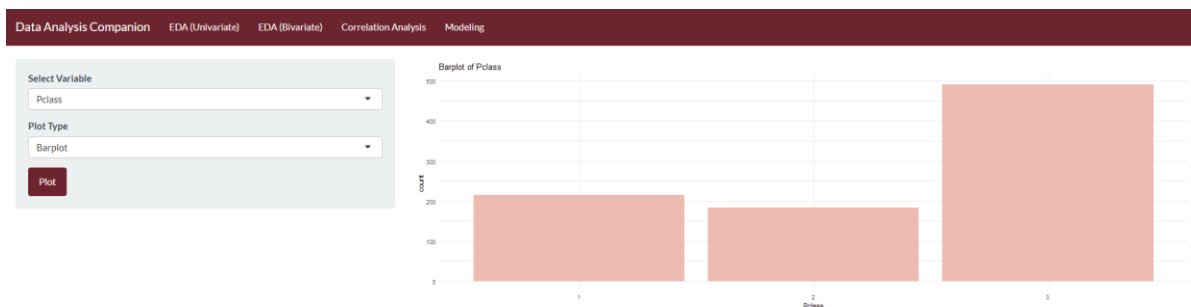


Figure 9 - Comparaison des situations socio-économiques des passagers

On peut ici voir que pas mal de passagers se trouvaient en 1^{ère} et en 2^{ème} classes mais la majorité était tout de même en 3^{ème} classe qui est donc la catégorie la « moins chère » parmi les trois. Il sera intéressant de mettre en corrélation la classe socio-économique (représentée par le prix du billet ici), avec la survie ou non des passagers.

Plus anecdotique qu'autre chose, la colonne « Name » affiche-t-elle un nuage de mot des mots les plus représentés dans les valeurs textuelles.

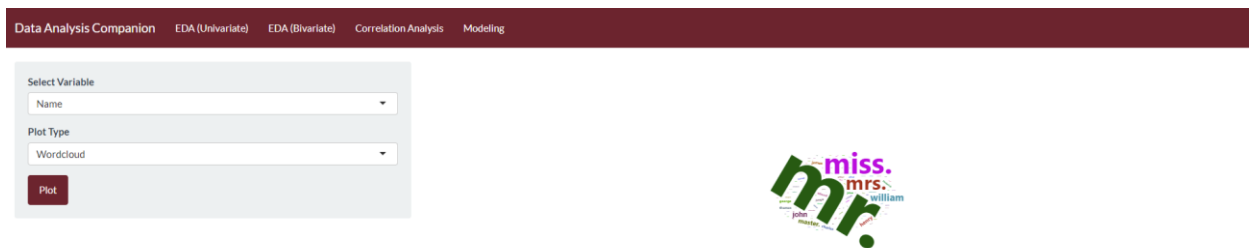


Figure 10 - Nuage de mots pour les colonnes textuelles

On peut donc naturellement y voir les appellations « Mr. », « Miss » et « Mrs » en tant que dominant mais on peut aussi apercevoir des prénoms récurrents comme « William » et « John ». Cela ne nous apporte dans notre cas pas beaucoup d'informations mais il est important de souligner cette fonctionnalité qui peut se révéler utile sur certains jeux de données.

Pour ce qui est de la prochaine colonne à laquelle nous allons nous intéresser, il s'agit de la colonne « Sex ». Elle a une corrélation particulière avec la colonne « Survived » car durant l'accident tragique du Titanic, les femmes avaient été évacuées en priorité.

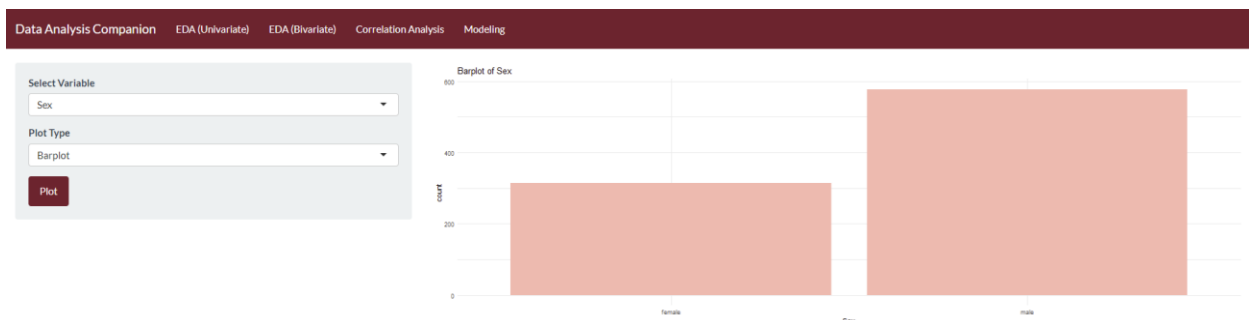


Figure 11 - Répartition des sexes sur le navire

On peut voir sur ce graphique que sur le navire il y avait environ deux fois plus d'hommes que de femmes, c'est un écart assez remarquable et intéressant.

Pour ce qui est de l'âge des passagers, l'interface nous propose, comme pour toutes les autres variables numériques, trois visualisations :

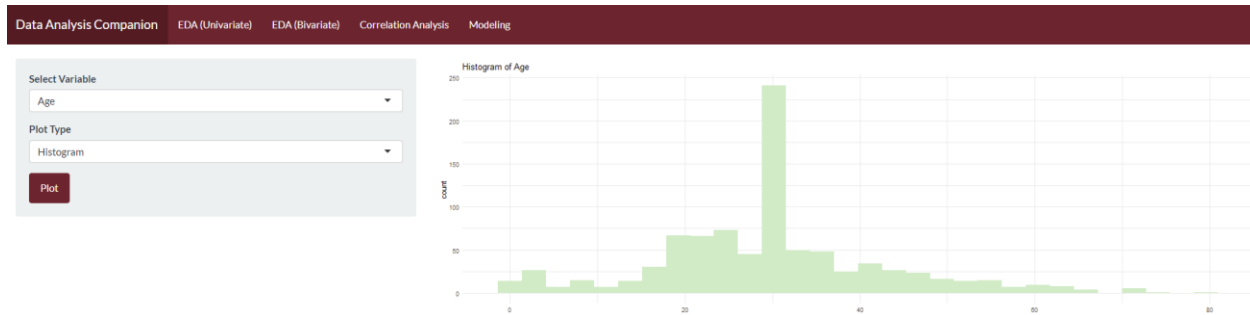


Figure 12 - Histogramme de l'âge des passagers

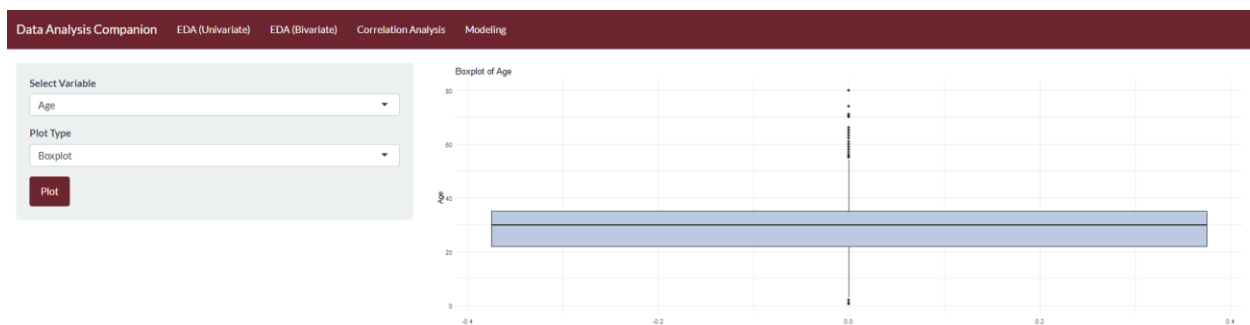


Figure 13 - Boxplot de l'âge des passagers



Figure 14 - Résumé Statistique de l'âge des passagers

Sur ces trois graphiques, plusieurs informations peuvent être dégagés :

- L'âge moyen des passagers est de 30 ans
- Une grande majorité des passagers ont entre 18 et 37 ans (Histogramme)
- Une partie notable des passagers est mineur
- Une infime partie des passagers à au-delà de 55 ans.

Pour continuer, intéressons-nous aux colonnes « SibSp » et « ParCh » pour « Siblings & Spouse » et « Parents & Children », ces colonnes donnent des informations plus personnelles sur les passagers. Grâce à ces colonnes nous pouvons savoir le nombre de membre de famille des passagers, qui ici est donc divisé en deux catégories, les frères et sœurs et les conjoints puis une autre catégorie étant les parents et les enfants.

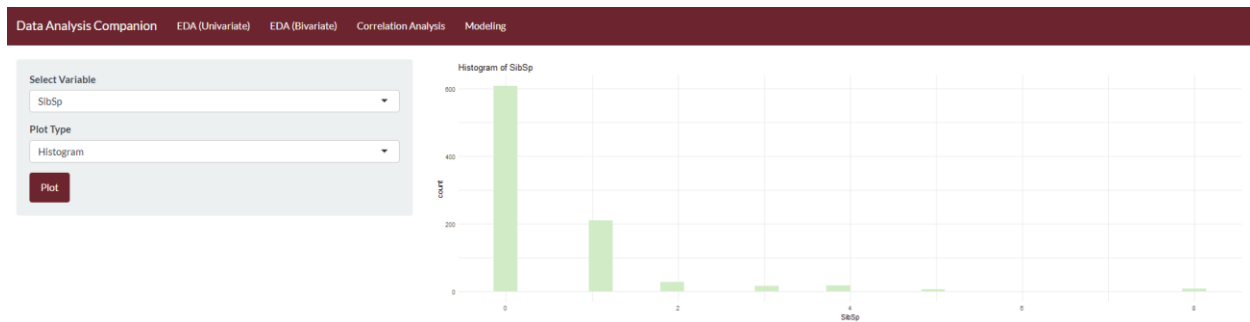


Figure 15 - Analyse du nombre de frères, sœurs et conjoints

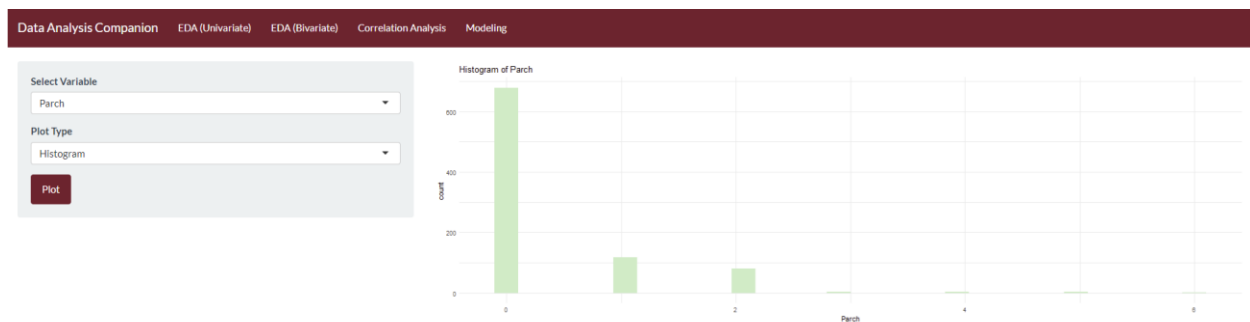


Figure 16 - Analyse du nombre de parents et enfants

On peut voir grâce à ces deux graphiques que beaucoup de voyageurs étaient seuls (la majorité) et une partie des voyageurs étaient accompagnés de quelques membres de famille, la plupart avec très certainement leur conjoints et leurs enfants (entre un et deux souvent).

On peut noter de rare cas ou plus de trois enfants / parents était présents (jusqu'à six pour le maximum). Et de même pour les frères / sœurs / conjoints avec de rare dépassements de quatre d'entre eux à bord (avec un maximum allant tout de même jusqu'à huit).

La dernière colonnes que nous analyserons est la colonne « Embarked » qui donne l'information du port auquel les passagers ont embarqués.

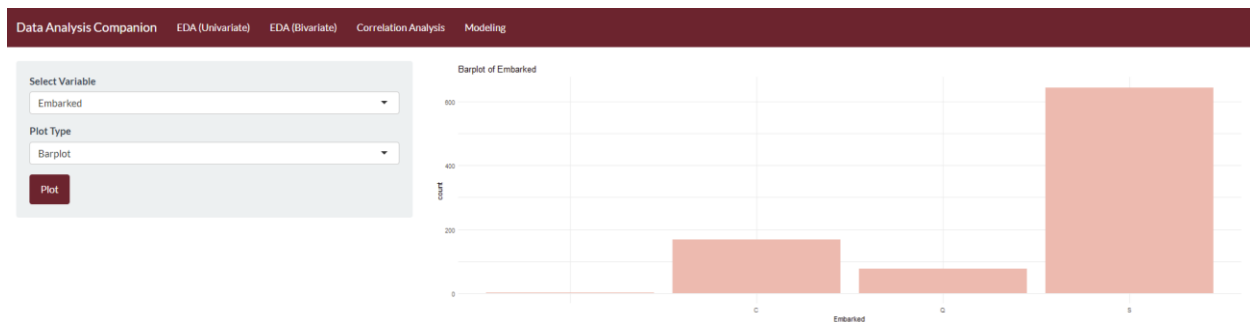


Figure 17 - Analyse de répartition du port d'embarquement

Comme on peut le voir sur ce graphique, une partie notable des passagers ont pris le point d'embarquement « C » qui fait référence à Cherbourg en France. Une minorité à embarqué au point « Q » qui fait référence à l'ancienne ville de Queenstown en Irlande. Enfin la majorité des passagers à embarqué au point « S » qui est la ville de Southampton en Angleterre.

Analyse bivariable

Relation entre variables clés (ex. survie et sexe, âge, classe)

Dans cette partie nous allons principalement nous concentrer sur la relation entre la colonne « Survived » et les autres colonnes catégorielles et numériques.

Dans cette catégorie de comparaison, le plus connu et le plus logique étant donné le contexte du drame du Titanic, le premier graphique intéressant sera celui de la survie ou non d'un passager en fonction de son sexe.

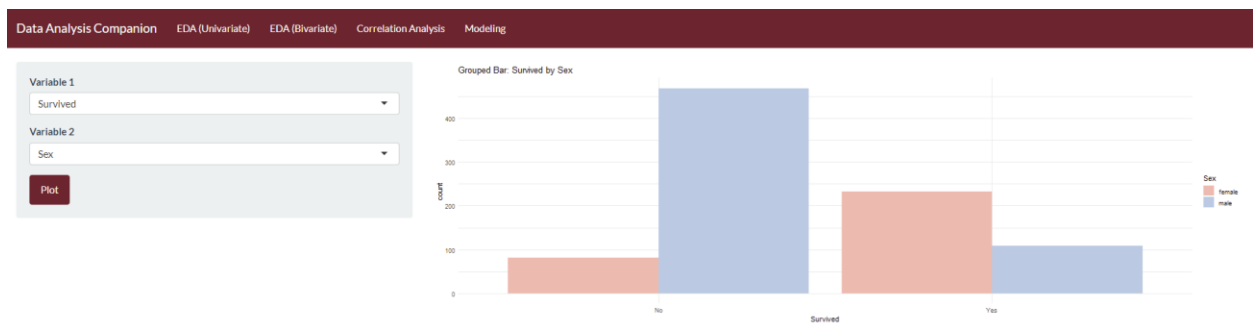


Figure 18 - Analyse de la survie par rapport au sexe du passager

On peut voir une nette corrélation entre ces deux colonnes, visuellement le graphique est très parlant et semble visiblement confirmer qu'en tant qu'homme (n'ayant pas la priorité d'évacuation pour la plupart) il était plus incertain de sortir vivant de ce drame.

Pour ce qui est de la survie en fonction de l'âge, nous voyons sur le graphique ci-dessous que la plupart des « jeunes » (entre 0 et 18 ans) ont un taux de survie plus élevé, ensuite les personnes aux alentours de 30 ans ont eu un taux de survie bien inférieur à 50% (courbe bleu bien en dessous de la courbe rose) puis nous pouvons voir que la plupart des personnes âgées ne s'en sont pas sorties non plus (probablement pour laisser leur place à des enfants par exemple).

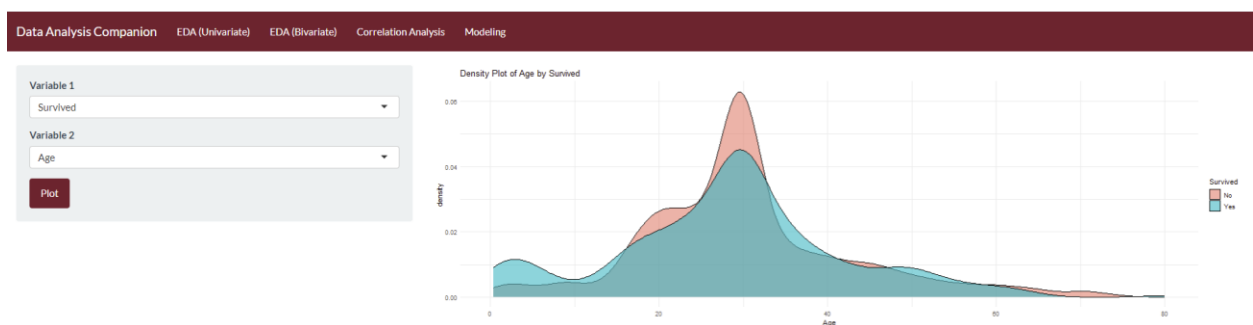


Figure 19 - Analyse de la survie en fonction de l'âge

Une autre analyse intéressante qui découle des deux analyses précédentes et la répartition des sexes en fonction de l'âge.

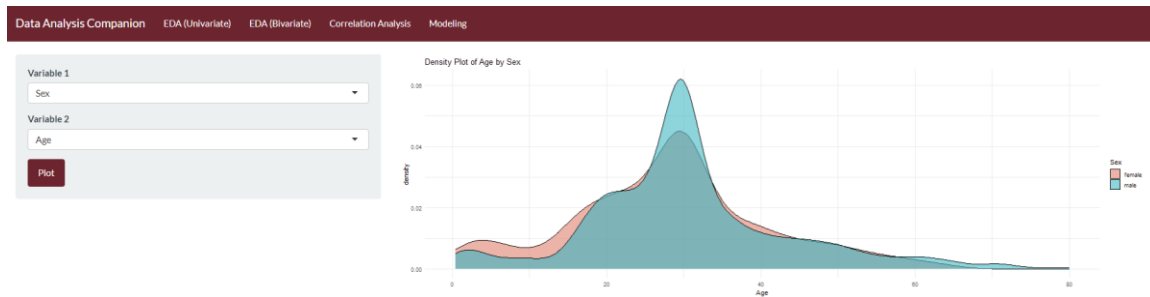


Figure 20 - Répartition des sexes en fonction de l'âge

En effet, sur ce graphique on peut voir que la majorité des trentenaires qui ont un taux de survie très bas, est en réalité des hommes. La même chose s'applique pour les personnes âgées et la réciproque s'applique pour les « jeunes » qui ont un plus grand taux de survie.

Nous allons maintenant analyser la relation entre le nombre de membre de famille et la survie des passagers.

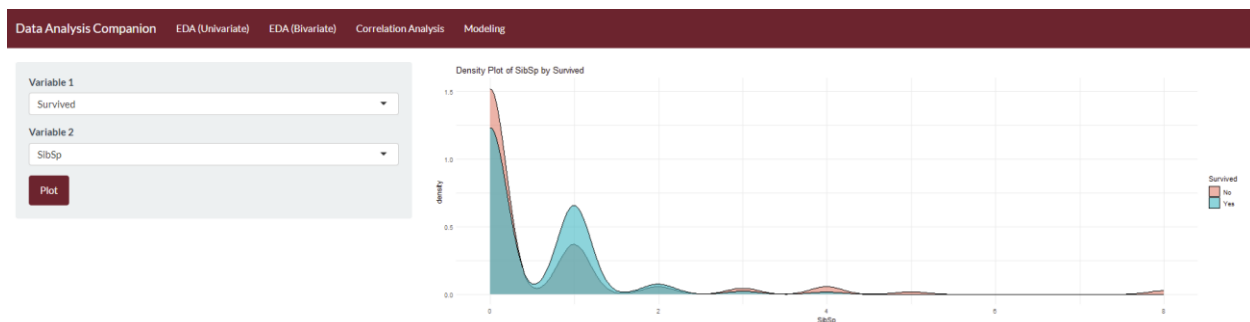


Figure 21 - Survie en fonction du nombre de Frères / Sœurs / Conjoints

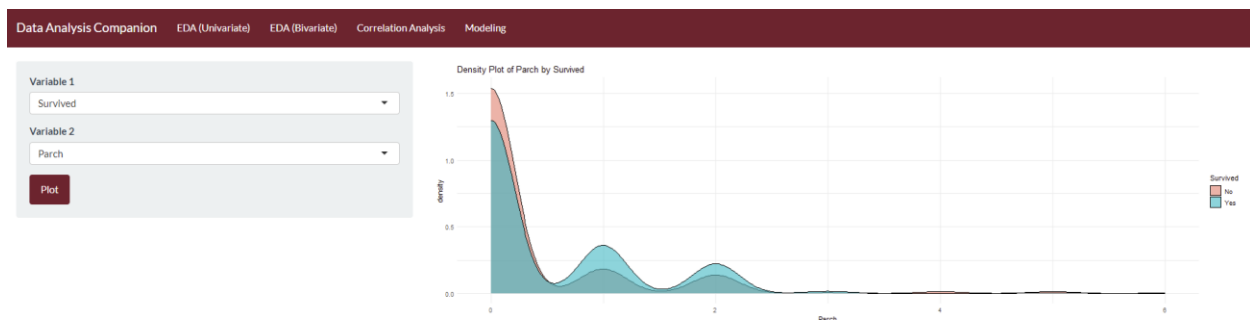


Figure 22 - Survie en fonction du nombre d'enfants / parents

Comme on aurait pu s'y attendre, les personnes ayant peu de membre de famille à bord ont un taux de survie plus bas que les personnes ayant un nombre plus important de membre de leur famille.

Une autre mise en relation primordiale est la survie en fonction de la classe socio-économique. Analysons donc la relation entre la survie et la classe des passagers (billet acheté).

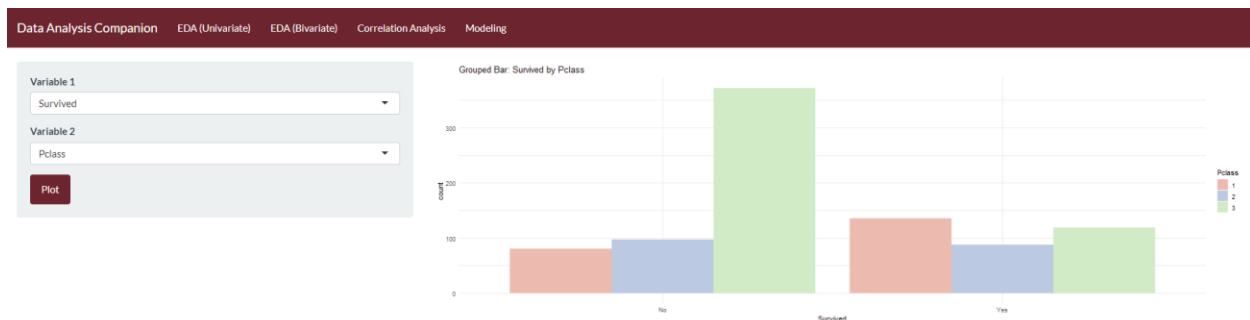


Figure 23 - Survie en fonction de la classe socio-économique

Malheureusement, le graphique correspond bien à nos attentes et les personnes ayant un billet de 3^{ème} classe (billet le moins chère) ont un pourcentage de survie bien plus bas que les autres classes. On peut voir qu'un favoritisme certain à été fait pour les 1^{ère} classe qui ont un très bon taux de survie (le seul étant positif).

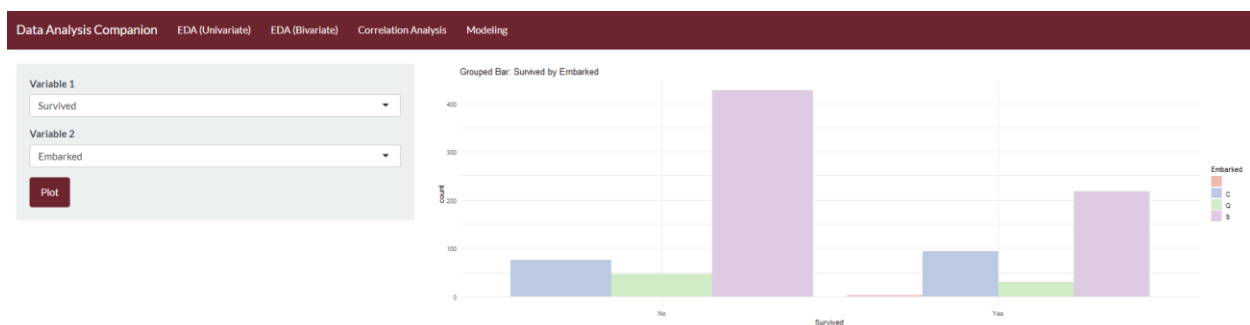


Figure 24 - Survie en fonction du point d'embarquement

Etrangement, nous pouvons remarquer que le taux de survie est différent selon les points d'embarquement, ce qui n'a pas tant de sens étant donné que peu importe le point d'embarquement d'une personne, ses chances de survie dans un accident sont supposément les mêmes.

On creuse donc plus loin et on trouve bien évidemment une explication logique à cela, la proportion des classes socio-économiques ayant embarqués dans ces différents ports :

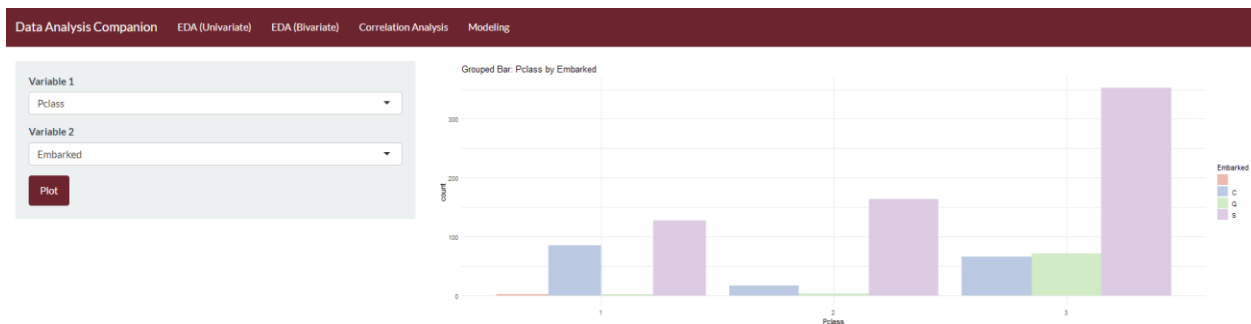


Figure 25 - Classe socio-économique en fonction du point d'embarquement

On peut apercevoir sur ce graphique une majorité des premières classes auraient embarqué à « Cherbourg » ce qui expliquerait le grand taux de survie de ce point et réciproquement la plupart des troisièmes classes ont embarqué dans la ville de Southampton ce qui explique le taux de fatalité énorme de ce point d'embarquement.

Certains autres graphiques peuvent être intéressants comme la répartition des sexes à travers les classes socio-économiques, là où l'on peut voir que les hommes représentent une écrasante majorité de la 3^{ème} classe et une majorité pas si dominante de la 1^{ère} classe.

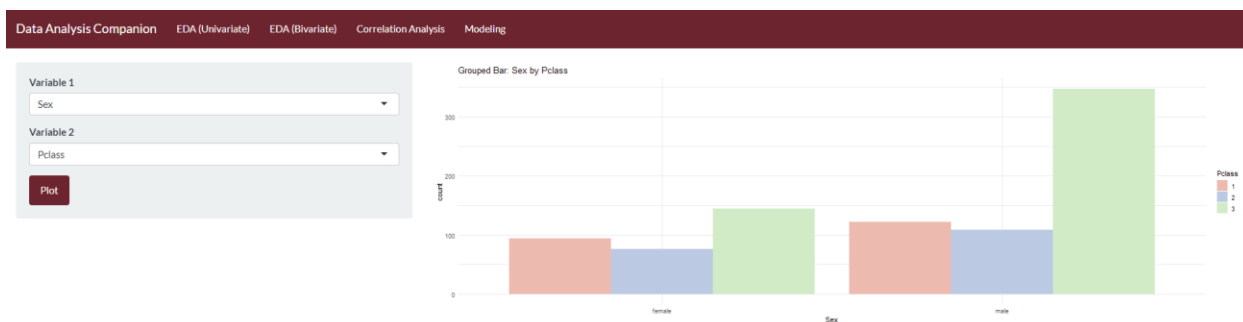


Figure 26 - Répartition des sexes en fonction des classes socio-économiques

Analyse des corrélations

Maintenant cette analyse exploratoire des données, nous pouvons également analyser les corrélations entre les différentes colonnes.

Notre interface intègre cette fonctionnalité avec une matrice de corrélation pour les variables numériques et une matrice de p-valeurs de test de χ^2 pour les variables catégorielles.

Voici le rendu pour notre jeu de données :

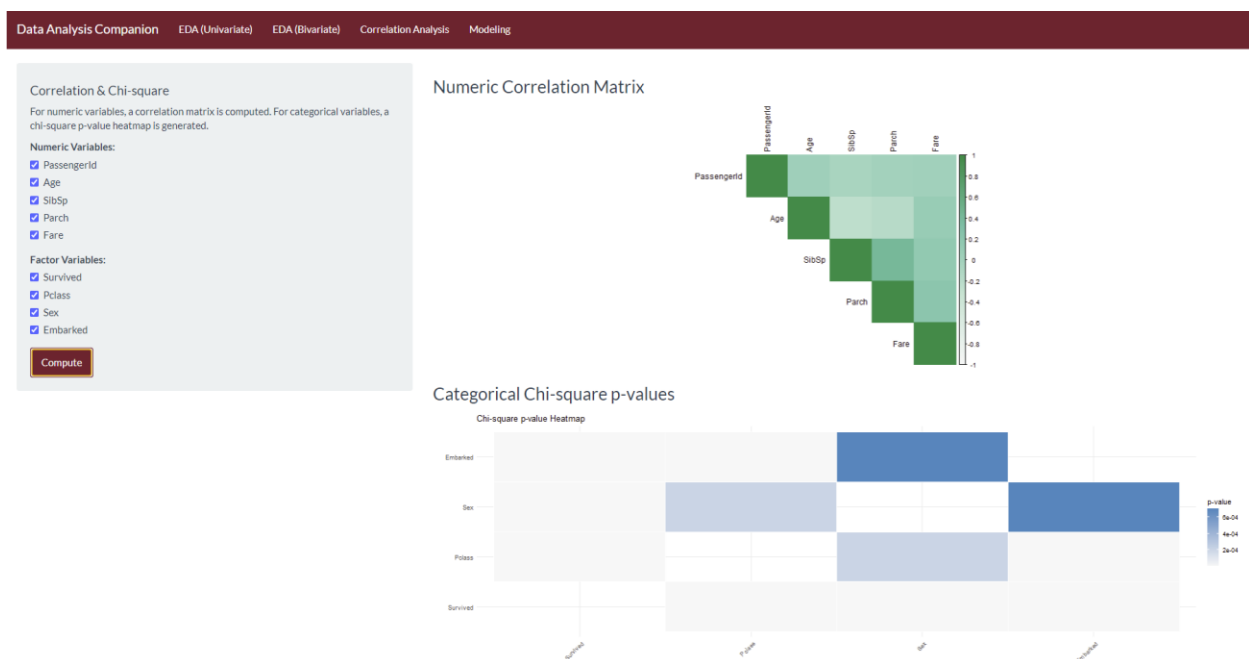


Figure 27 - Analyse de corrélation

Nous allons analyser un par un les éléments ici, commençons par la matrice de corrélation des variables numériques :

- Nous pouvons voir que « SibSp » et « Parch » sont corrélé
 - o Assez logique, les personnes ayant des frères / sœurs / conjoints sur le navire sont plus susceptible d'également avoir des parents / enfants (particulièrement vrai pour la combinaison conjoint / enfants)
- Les autres variables semblent assez peu corrélées entre elles
 - o « PassengerId » n'a pas lieu d'avoir de corrélation avec d'autres valeurs étant donné que c'est un identifiant.
 - o « Age » semble avoir une légère corrélation négative avec « SibSp » et « Parch », en effet les personnes âgées auront tendance à avoir moins de famille à bord alors qu'un enfant est presque certain d'avoir ses parents à bord.

Pour ce qui est de la matrice de p-valeurs de test de χ^2 pour les variables catégorielles :

- Toutes les classes semblent être étroitement liées, ce qui correspond avec notre exploration bivarié.
- La plus grande p-valeur étant 6e-4, nous pouvons ainsi rejeter l'hypothèse nulle d'indépendance entre les variables.

Modélisation prédictive

Modèles testés : Random Forest, SVM, Régression logistique

Pour entrainer notre modèle, on utilisera comme dit précédemment les données prétraités, c'est-à-dire, contrairement à notre analyse exploratoire, nous utiliserons les données normalisées et suréchantillonnées.

Nous commencerons par un modèle de base de classification sur la variable « Survived » avec toutes les autres variables (non textuelles) en tant que prédicteurs :

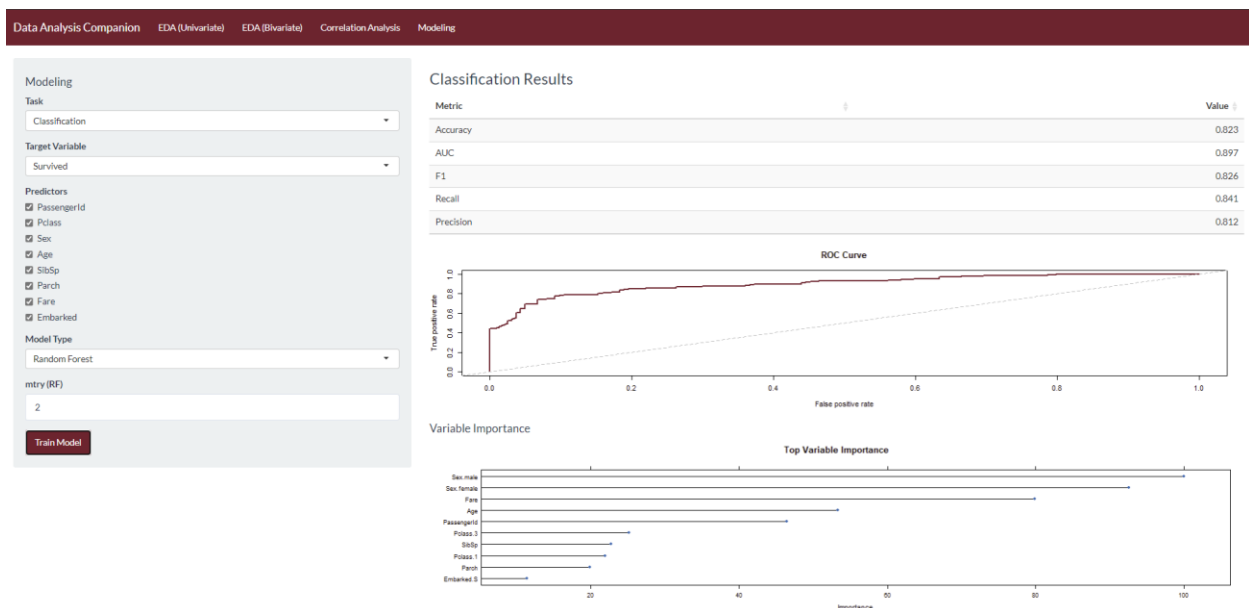


Figure 28 - Entrainement d'un modèle de base Random Forest

Avec les paramètres de base, on atteint tout de même un très bon score avec un F1-score de 0,826 et une accuracy de 0,823.

Nous pouvons voir notre courbe ROC qui as un très bon AUC de 0,897. En dessous, nous pouvons également voir les variables qui explique le plus notre modèles, dans notre cas sans surprise par rapport à notre analyse, le sexe de l'individu est le principale explicateur de sa survie ou non.

En troisième place « Fare » qui est le prix payé pour le ticket qui donc représente d'une façon numérique ce que représente également « Pclass ». Puis en quatrième nous retrouverons donc l'âge de l'individu qui comme on l'avait vu as tout de même un effet important sur la survie.

Nous essayons ensuite les deux autres modèles avec toujours les hyperparamètres de base :

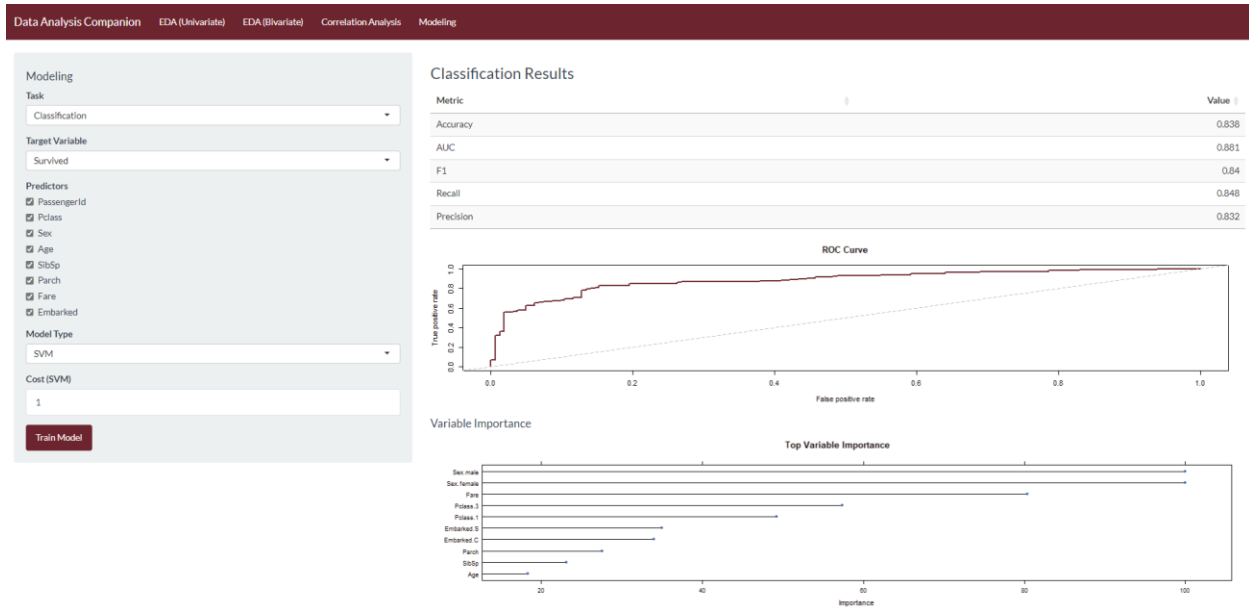


Figure 29 - Entrainement d'un modèle de base SVM

Le SVM atteint de meilleurs résultats en terme d'accuracy et de F1-score avec des scores respectivement à 0,838 et 0,84 mais reste légèrement en dessous en terme d'AUC.

On peut voir au niveau de l'explicabilité que celui-ci se base bien moins sur l'âge que « Random Forest ».

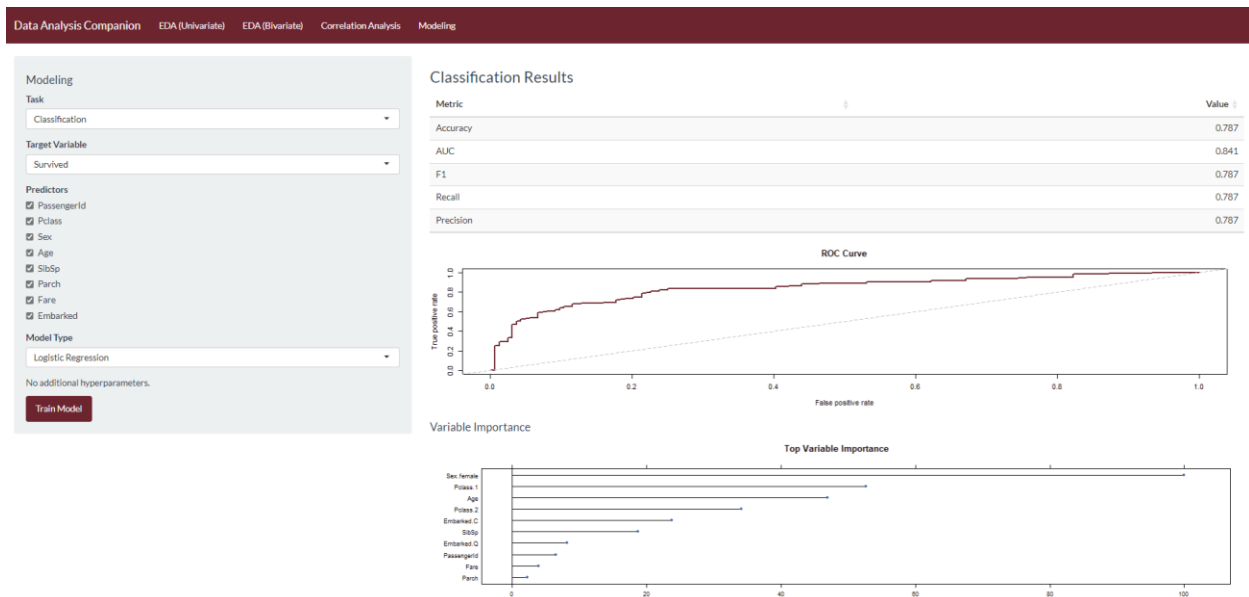


Figure 30 - Entrainement d'un modèle de base Régression Logistique

Avec le modèle de régression logistique on obtient des scores en tout points, on peut voir notamment que le modèle se base bien moins sur la variable « Fare » ce qui est bien dommage étant donné qu'elle donne beaucoup d'informations.

Amélioration des hyperparamètres & Comparaison des modèles

Dans cette partie nous essayerons de jouer sur les paramètres des modèles Random Forest et SVM qui sont les suivants :

- Mtry (Random Forest) : Nombre de variables sélectionnées à chaque split
- C (SVM) : Coût de la régularisation, contrôle le compromis entre la marge maximale et l'erreur de classification.

Nous ferons également varié les colonnes sélectionnées en tant que prédicteurs afin de débruiter le modèle et de le rendre le plus simple possible pour son maximum d'accuracy.

Tableau 1 - Amélioration des hyperparamètres et résultats

Modèle	Paramètre	Prédicteurs	Accuracy	AUC	F1-score
Random Forest	Mtry = 4	PassengerId Sex Age SibSp Parch Fare	0.872	0.937	0.863
SVM	C = 1	Toutes	0.838	0.881	0.84
Régression Logistique	Aucun	Pclass Sex Age SibSp	0.799	0.834	0.795

Dans le tableau ci-dessus, les meilleurs paramètres pour chaque modèle avec nos données.

Comme on peut voir dans le tableau, le modèle Random Forest qui à la base était moins bon que le SVM, avec un peu d'amélioration sur ses hyperparamètres devient au final meilleur et surpasse le modèle SVM qui était déjà à son maximum de score.

Nous obtenons donc un F1 score final de 0.863 même avec un modèle simple et une AUC de 0.937 !

Cependant lorsqu'on regarde l'explicabilité de ce modèle on peut voir cela :

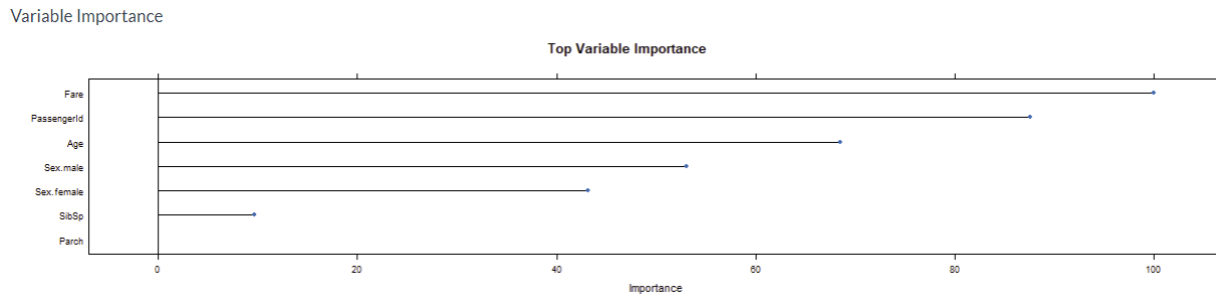


Figure 31 - Explicabilité du meilleur modèle

On voit que « PassengerId » a une importance beaucoup trop importante par rapport à la nature de la colonne. Cela pourrait être dû à notre suréchantillonnage, étant donné que certains cas ont été dupliqués, le modèle apprend à retenir les identifiants des passagers plutôt que leurs caractéristiques.

Nous devrions donc, pour avoir des résultats objectifs, retirer cette variable du set des prédicteurs car elle pose problème éthique au niveau de l'évaluation qui nous pousse à penser que une grande partie de cette variable entraîne un surajustement du modèle qui est bénéfique sur nos métriques mais est mauvais en terme de généralisation et d'évaluation objective.

Ici nous séparons nos données d'entraînement et de test, c'est une bonne pratique et permet d'évaluer le modèle sur des données qu'il ne connaît pas. Cependant cet identifiant unique peut aller à l'encontre de cette approche, en effet les valeurs dupliquées peuvent très bien se retrouver éparpillées dans les deux sets ce qui crée de la contamination de nos données de tests.

Avec ces modifications, nous atteignons les scores suivants :

- Modèle : Random Forest
- Paramètre : Mtry = 4
- Prédicteurs :
 - o Sex / Age / SibSp / Parch / Fare
- Accuracy : 0.848
- AUC : 0.907
- F1-Score : 0.842

Ces résultats restent très bon et meilleurs que les autres modèles.

Conclusion et perspectives

Suggestions d'amélioration pour l'interface ou l'analyse

Dans de futures versions, si plus de ressources sont disponibles, des modèles plus complexes pourraient également être implémentés ainsi qu'une fonction de diagnostic automatique des meilleurs hyperparamètres.

Pour le moment de telles fonctionnalités sont dur à imaginer car des limitations matérielles et temporelles imposent une interface raisonnablement compliquée.