

## Rapport personnel de RVI

Jules Vittone-Burnel

### Présentation générale du projet

L'objectif de ce projet était de représenter dans un environnement de réalité virtuelle les informations obtenues à l'aide d'une analyse en composante principale (ACP) faite dans le cadre du cours de Data Science.

L'intérêt de la représentation en 3D d'une ACP est principalement de pouvoir accéder à une dimension d'information en plus. En effet, les résultats se présentant sous la forme de tableau contenant  $N \times M$  entrées,  $N$  étant le nombre d'éléments à étudier (Dans notre cas, les éléments sont les différents secteurs de l'économie française en 2013) et  $M$  étant le nombre de critères étudiés (Dans notre cas, les critères sont les comptes du bilan comptable utilisé dans le cadre de l'établissement de la comptabilité nationale).

Nous avons fait le choix de nous servir d'un casque de réalité virtuelle qui était fourni dans le cadre de ce cours (HTC Vive) et des télécommandes associées.

## Travail réalisé

Dans ce projet, j'eus la charge de m'occuper du traitement des données avant leur intégration et leur utilisation par le logiciel. La première difficulté à laquelle nous fûmes confronté, fut la quantité de données en entrée, en effet, nous avons plus de mille secteurs concernés par l'étude de l'INSEE et après les avoir affichées une première fois à l'écran, nous nous sommes rendu compte qu'il n'était pas possible de tous les afficher en même temps sous peine de noyer complètement l'utilisateur sous les informations.

Nous avons donc pris le parti de regrouper les différents secteurs au sein d'ensemble plus grands, ces ensembles ayant été définis par nos soins, ils ne représentent pas forcément un découpage idéal mais ils nous semblent malgré tout plutôt représentatifs.

Je m'occupai par la suite de créer les différents scripts en C# qui permettaient à la fois de générer la scène à partir des préfab créés par Sullivan Honnet.

La scène se décompose en un objet principal qui est l'utilisateur, l'objet représentant l'utilisateur dans Unity comporte un script qui se démarre au lancement de l'application, il crée un Game Object qui est celui qui contrôle réellement tous les éléments avec lesquels l'utilisateur interagit.

La raison de cette construction est un problème que nous ne sommes pas parvenu à résoudre autrement et qui vient directement de l'utilisation d'un casque de réalité virtuelle HTC Vive et de l'utilisation de la librairie Steam VR. Les interactions des manettes ne sont pas reconnus et traités comme des interactions à la souris et nous n'avons pas trouvé une manière satisfaisante d'associer les différents éléments à l'écran aux manettes.

En associant tous les éléments à l'utilisateur, nous pouvions accéder à tous ses composants et donc aux manettes, c'est pour cette raison que le Game Object qui nous sert à contrôler et créer tous les objets se trouve dans l'utilisateur. Si nous avions eu plus de temps à consacrer à ce projet, il aurait été intéressant de chercher comment nous passer de cette astuce.

Le Game Object contient un script qui va se déclencher au moindre clic de l'utilisateur et créer les deux graphiques qui s'affichent à l'écran à partir d'un préfab. Ces graphiques servent à donner une échelle de comparaison à l'utilisateur et à indiquer le centre de la sphère de corrélation dont toutes les autres sphères font partie. Ils permettent également de déterminer plus simplement si des éléments sont corrélés (s'ils s'affichent dans la même région, ils sont corrélés), anti corrélés c'est-à-dire qu'ils évoluent de manière inverse (s'ils s'affichent dans des régions de part et d'autre du centre du graphe) et décorrélés (s'ils ont un angle par rapport au centre du graphe de 90°).

Les deux graphes servent à représenter l'un (celui de gauche) les différents critères de l'étude comme le nombre d'unité légale ou les actifs des secteurs, l'autre (celui de droite) représente l'étude ACP des secteurs ou ensemble de secteurs.

Les sphères représentant les secteurs sont issues d'un préfab différent des autres sphères et contiennent chacune un script qui s'exécute à leur création et qui va créer des sphères correspondant aux secteurs de l'ensemble associées à la sphère correspondant à leur ensemble.

Ces sphères ne seront pas affichées à l'écran mais dès qu'une sphère est sélectionnée et que l'utilisateur clique dessus toutes les sphères d'ensemble sont effacées et les sphères de l'ensemble sont affichées.

Il est à noter que ce préfab est facilement réutilisable, il serait donc possible si nécessaire de créer des ensembles contenant des sous-ensembles contenant eux-mêmes les secteurs. Ce choix ne nous a pas semblé pertinent dans le cadre de ce projet mais ayant été envisagé, le code est prévu pour qu'il soit possible de faire ça.

La taille des sphères des secteurs et des ensembles est variable et dépend de leur influence sur les résultats globaux de l'étude, si elles ont une importance très forte, elles seront bien plus grosses que les autres. Contrairement à celle des critères car les critères tendent à avoir des importances comparables et les variations de taille étaient moins significatives.

Le zoom est une fonctionnalité que nous avons prévu dès le début du projet car ce fut l'un des problèmes récurrents que nous eûmes durant notre projet de data science à savoir que certains secteurs ayant une importance démesurée par rapport à d'autres, nous aurions apprécié de pouvoir zoomer sur les images produites par le logiciel que nous avons utilisé mais ce n'était pas possible.

Nous avons choisi pour le zoom de faire s'éloigner du centre les sphères pour que l'utilisateur puisse toujours garder son point de repère fixe et éviter qu'il ne se perde trop car le volume de la pièce qui entoure l'utilisateur est très grand.