Mini-rapport données US Census

Quelques expériences sur les données du recensement américain de 1995…

Premières observations

Les données sont importantes en volume (en tout environ 300 000 instances). Elles sont décrites sur 40 dimensions variées : caractéristiques individuelles (sexe, âge), informations relatives à la situation professionnelle (équivalent des codes CSP), familiale et à la fiscalité (revenus, imposition).

Une bonne partie des dimensions est catégorielle (33) et correspond à des questions à choix multiple dans le formulaire de recensement.

Il faut tout d'abord noter que beaucoup d'instances ont des valeurs manquantes (environ 52%), les principales dimensions incomplètes correspondant à la question du déménagement.

Une première description simple des données a été réalisée en calculant les premiers moments statistiques et les percentiles sur les dimensions numériques d'une part, et le mode et sa fréquence, ainsi que le nombre de valeurs uniques sur les dimensions catégorielles. Une vérification rapide sur les dimensions "sex" et "age" permet de voir que les données ne sont pas aberrantes a priori car assez bien réparties (le détail est donné ci-dessous).

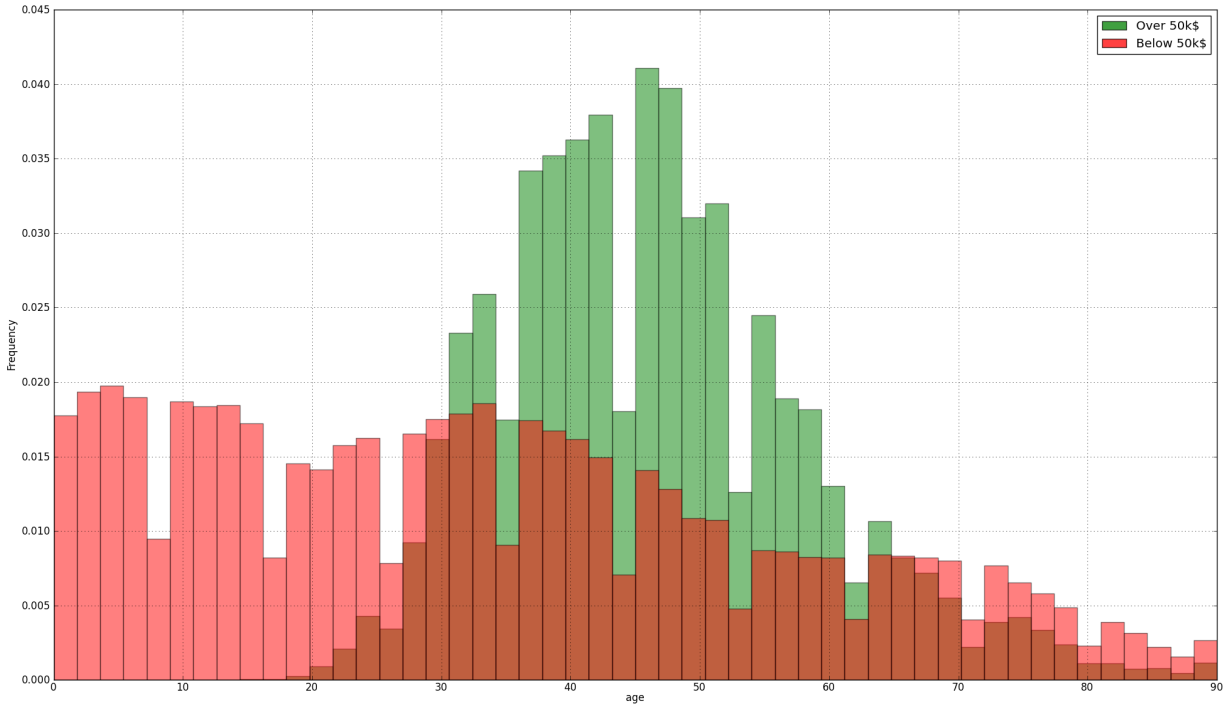
```
age                              sex
count    199523.000000           count      199523
mean         34.494199           unique          2
std          22.310895           top        Female
min           0.000000           freq       103984
25%          15.000000
```

```
50%          33.000000
75%          50.000000
max          90.000000
```

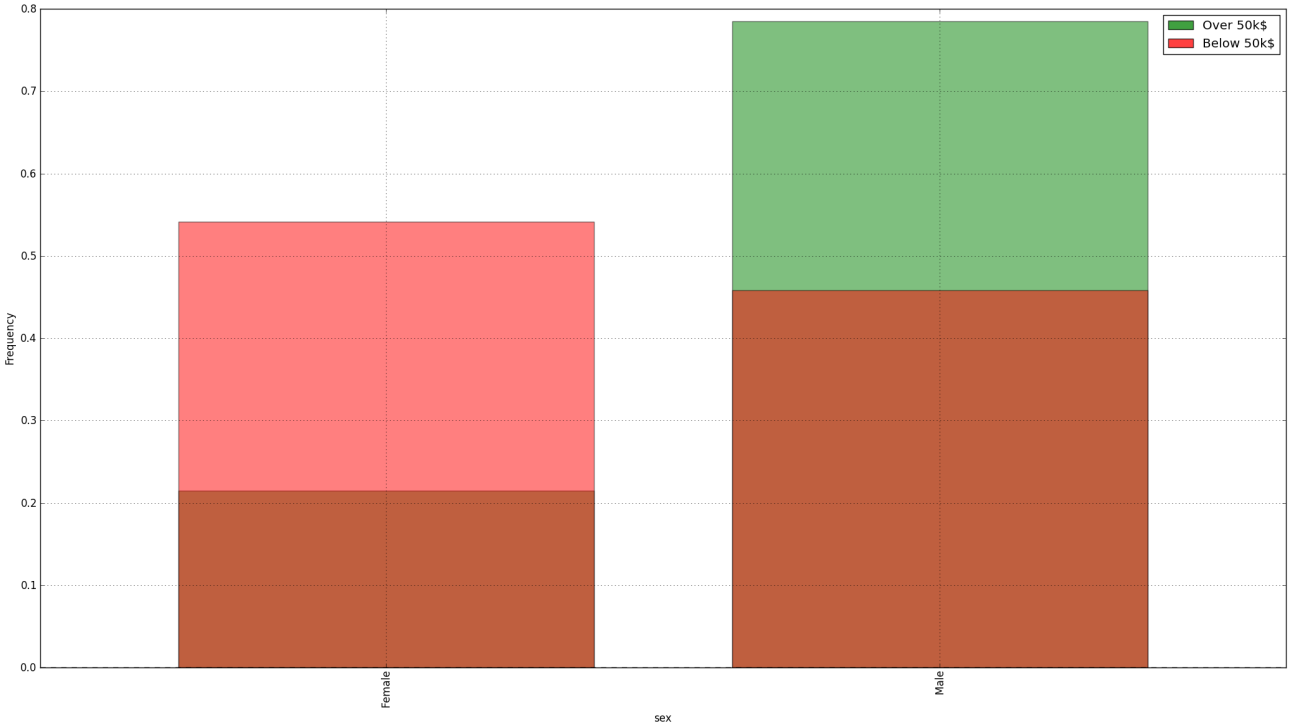
Comparaison des distributions entre les deux classes

Les données ont été séparées selon le revenu, avec un seuil binaire à 50 000$. Pour y voir plus clair dans les différences entre ces deux classes, j'ai inspecté tout d'abord les distributions univariées.

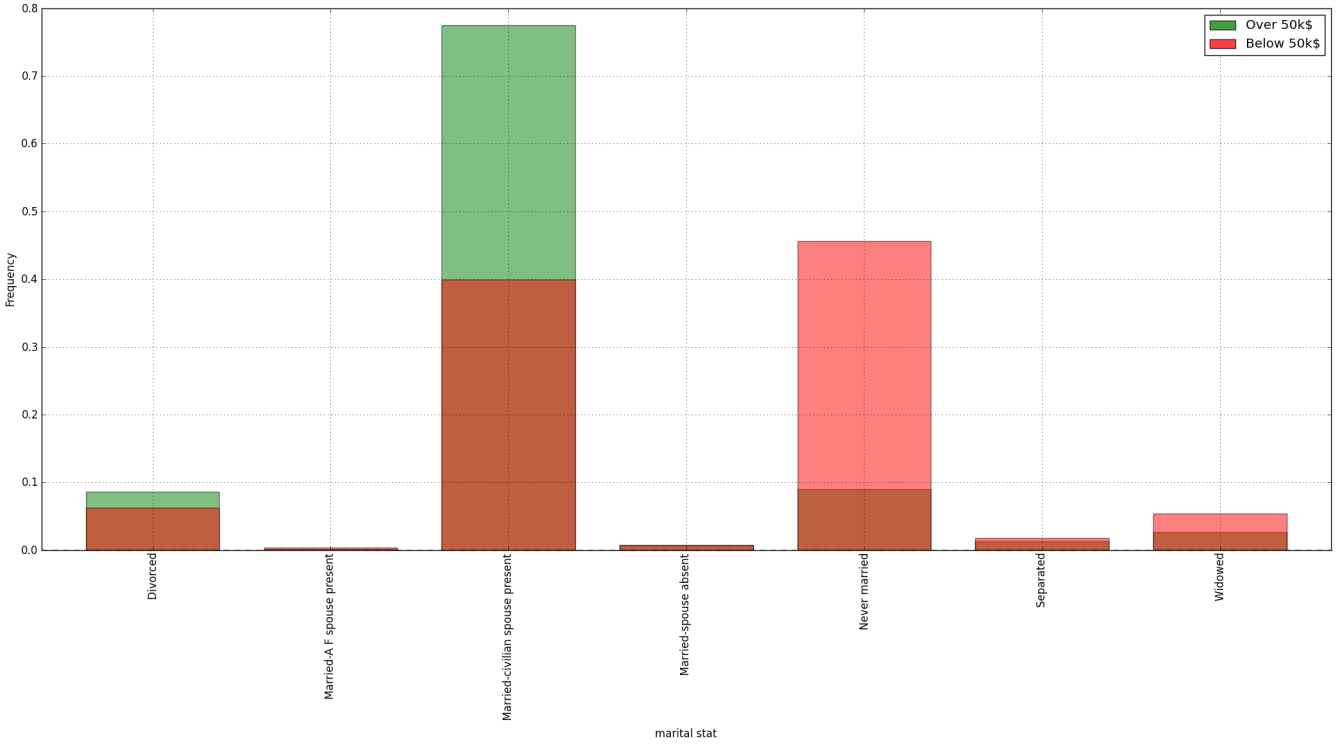
Des faits attendus émergent : les personnes gagnant plus de 50 000$ (classe "Over" dans la suite ; "Below" pour moins de 50 000$) sont plus âgées, mariées (corrélé avec l'âge) ; ce sont des Blancs de manières plus marquée que pour la classe "Below". Et le fait le plus marquant est que parmi la classe "Over", il y a très peu de femmes (environ 15%).



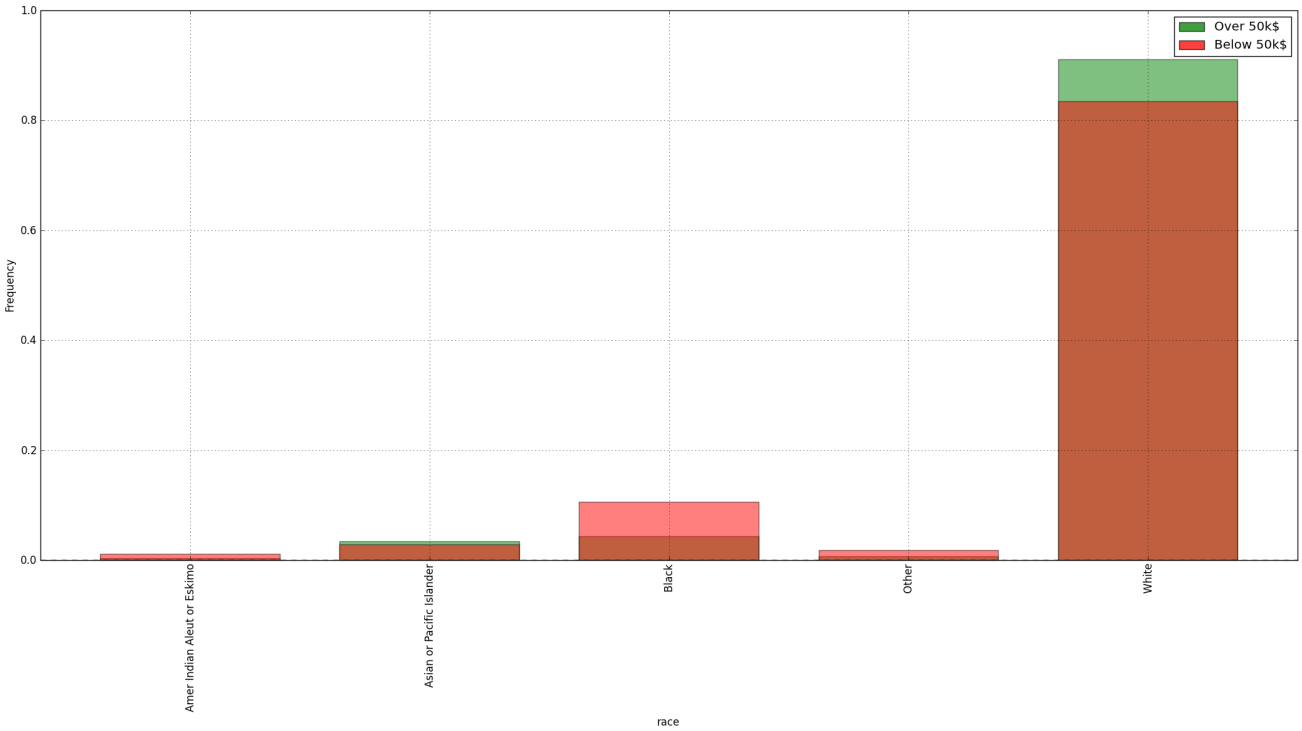
Distribution de l'âge.



Distribution de la variable “sexe”.

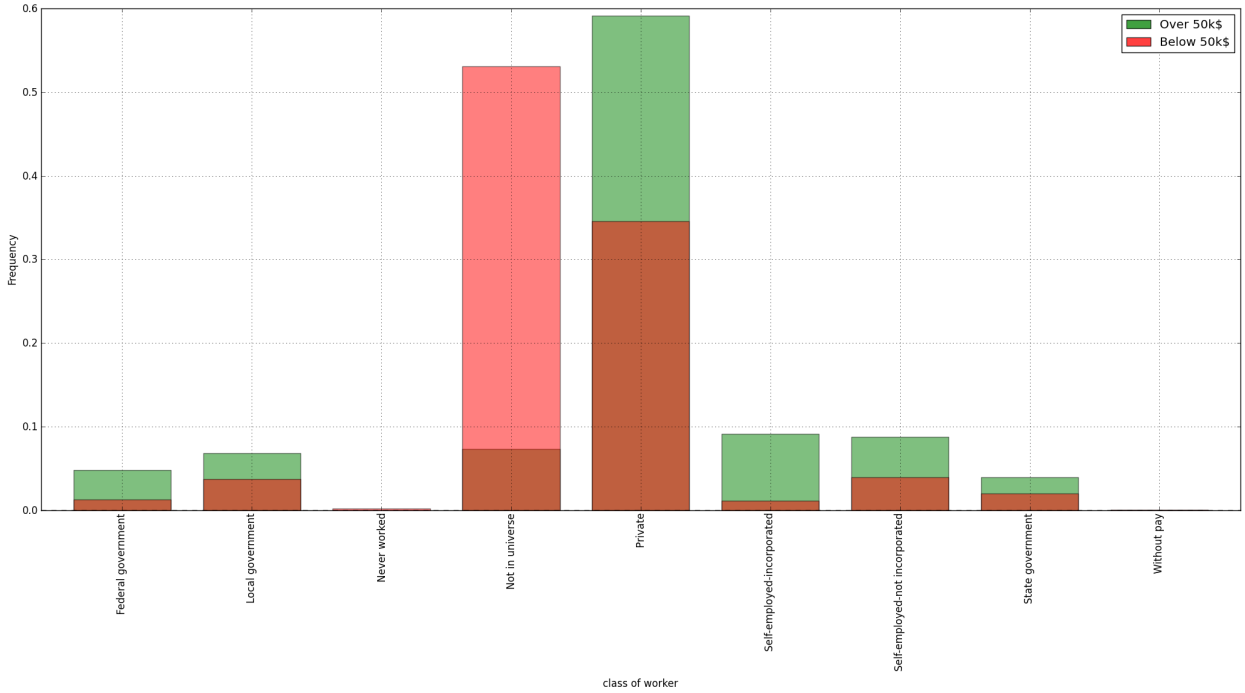


Statut marital.

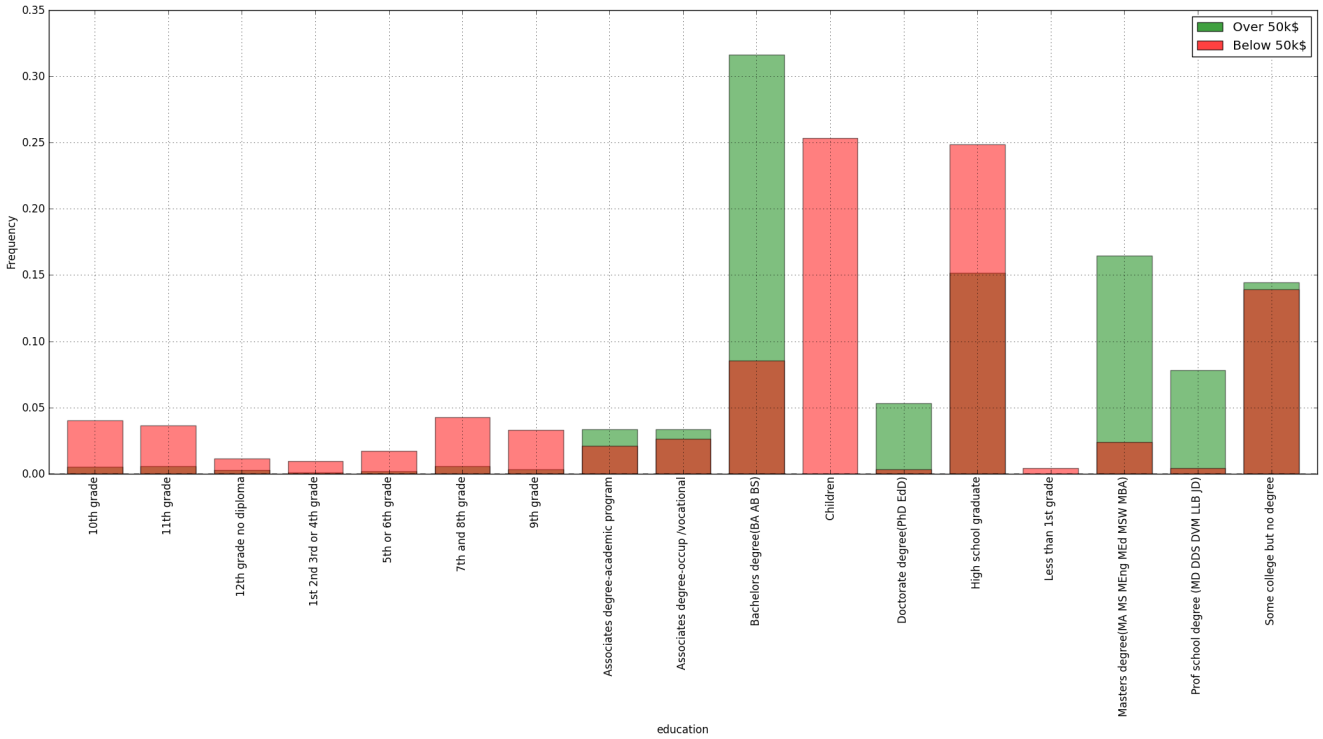


Race.

Parmi les autres différences facilement observables entre les deux conditions, on peut citer la catégorie socio-professionnelle, l'éducation ou le fait d'être propriétaire.



Secteur de l'emploi.



Education.

On peut dresser le portrait du citoyen moyen pour chaque classe et inspecter leurs différences en examinant la médiane pour les données numérique et le mode pour les données catégorielles. Le résumé de ces différences est donné dans le tableau ci-dessous.

Paramètre	Over 50k$	Below 50k$
Age	45	31
Sexe	Homme	Femme
Secteur de l'emploi	Privé	N/A
Education	Diplôme du niveau Bachelor (équivalent License)	"Enfant"[1]
Situation maritale	Marié, époux présent	Jamais marié
Secteur industriel	Manufacturing-durable goods	N/A

[1] Pas très clair dans les méta-données, correspond vraisemblablement au niveau primaire.

Type de poste	Executive admin and managerial	N/A
Statut fiscal	Imposable (conjoints actifs)	Non-imposable
Statut familial	Propriétaire	Enfant de moins de 18 ans
Semaines travaillées par an	52	0

On voit facilement que, si l'on peut caractériser assez précisément la classe "Over", c'est plus difficile pour la classe "Below" car elle inclut les enfants et les jeunes.

Paramètres pertinents

On a déjà une idée des paramètres pertinents pour la tâche de classification, on va cependant les trier en utilisant pour critère la corrélation avec la classe.

Pour cela, il faut numériser les paramètres catégorielles. La méthode usuelle consiste à transformer un paramètre catégoriel en N paramètres booléens, où N est le nombre de valeurs uniques. Comme cela engendre une augmentation du nombre de paramètres et que j'ai rencontré des problèmes de mémoire, j'ai choisi de laisser tomber certains paramètres à cause de leur nombre de valeurs uniques trop important et de leur relative indépendance à la classe, jugée visuellement d'après les histogrammes.

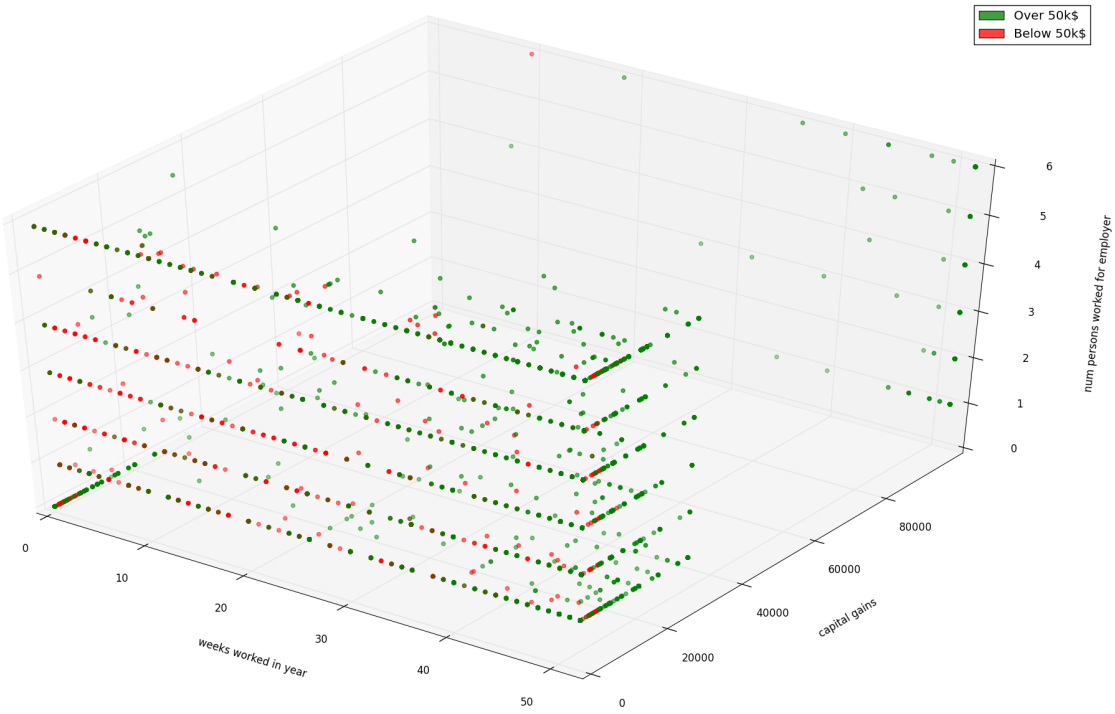
Après avoir supprimé 7 paramètres et converti les paramètres catégoriels restant, on récupère un problème à 188 dimensions.

Après calcul des corrélations, les dix premiers paramètres (en valeur absolue) sont représentés dans le tableau ci-dessous.

Paramètre	Corrélation avec la classe
weeks worked in year	0.262316136132
major occupation code_ Executive admin and managerial	0.24158429802
capital gains	0.240724819716
num persons worked for employer	0.222684028201

major industry code_ Not in universe or children	-0.221969231463
major occupation code_ Not in universe	-0.221969231463
class of worker_ Not in universe	-0.220913520224
detailed household summary in household_ Householder	0.212777563852
major occupation code_ Professional specialty	0.212685738419
tax filer stat_ Joint both under 65	0.20519309151

On sélectionne les trois premiers paramètres pour construire une visualisation du problème dans l'espace (scatter plot 3D).



Représentation en 3D du problème avec les trois paramètres les plus corrélés à la classe.

Cette représentation nous guide pour le choix d'un classifieur puisque le problème paraît séparable linéairement avec une erreur acceptable. Point intéressant, on peut aussi remarquer sur la figure ci-dessus un cluster d'instances de la classe "Over" qui déclare ne pas travailler (retraités ou rentiers).

Entraînement d'un classifieur et évaluation

Après avoir choisi un classifieur linéaire (ou plutôt une régression linéaire dont j'ai seuillé la fonction d'erreur pour qu'elle se comporte comme un classifieur), j'ai entraîné un modèle sur les données d'apprentissage.
La prédiction sur les données de test donne les performances suivantes :

	precision	recall	f1-score	support
Below	0.99	0.81	0.89	93576
Over	0.23	0.89	0.37	6186
avg / total	0.94	0.81	0.86	99762

Dans notre cas, c'est plutôt le rappel qui nous importe puisqu'on cherche à bien identifier les personnes de la classe "Over". Les scores sont donc assez satisfaisant avec un simple modèle linéaire.

Un modèle bayésien naïf permet cependant d'améliorer le rappel sur la classe "Over" au détriment de la précision (le rappel chute également sur la classe "Below") :

	precision	recall	f1-score	support
Below	0.99	0.56	0.71	93576
Over	0.12	0.95	0.22	6186
avg / total	0.94	0.58	0.68	99762

J'ai également entraîné un SVM linéaire, qui apporte un peu de mieux par rapport au modèle bayésien :

	precision	recall	f1-score	support
Below	0.99	0.61	0.76	93576
Over	0.14	0.95	0.24	6186
avg / total	0.94	0.64	0.73	99762

Remarques - difficultés

Toute l'analyse a été effectuée en Python avec les bibliothèques pandas et scikit-learn pour me conformer à la consigne ; je ne les connaissais que vaguement, j'ai donc passé un temps non-négligeable à me documenter sur leur fonctionnement.

J'ai l'habitude d'utiliser Python pour la manipulation des données et pour des algorithmes ne demandant pas des performances poussées. Vu la taille des données, j'ai rencontré des problèmes de mémoire qui m'ont obligé à faire des choix non-triviaux vu la contrainte de temps : j'ai notamment sous-échantillonné la classe "Below". Je me suis également forcé à utiliser mon ordinateur personnel, j'aurai passé du temps sur une méthode d'évaluation plus poussée autrement. La cross-validation pour l'apprentissage du SVM n'était pas faisable sur ma machine ; c'est le genre de tâche que je passe sur un cluster de calcul en temps normal.

Avec plus de ressources, j'aurais lancé une sélection de paramètres plus sophistiquées (du type sélection flottante) et fait un peu de clustering...