Machine Learning in Datatscience - Project Report
Project 2

Matthias Krug
Alper Savas
Ali Bektas

January 20, 2022

Feature selection - Dataset preparation

Decide on which features to use for clustering. Note that the dataset consists of features of different types. Justify your design choices. What does your final dataset for the clustering task look like? Please list and describe the selected features.

As we already know the dataset from Project 1 not much work needs to be done exploring the adult dataset. We took the Important Variables: age, education_num, occupation, relationship status, martial-stat, work class and hours per week. Then we turned them Numerical and Normalized them. For Occupation, we used the percentages of people making over 50K in said working class as the numerical value. Then to reduce the number of Dimensions, we performed a PCA and kept only the 3 (4 for DBSCAN) most significant Components.

Clustering

Choice of algorithms

Choose at least two clustering algorithms from the ones covered in the lecture

We selected DBSCAN and K-Means to get both a Global and a density based Clustering Algorithm, so that we could compare how well they work on our dataset. We also tried out one hierarchical clustering. In the End, K-Means performed significantly better than DBSCAN. probably because our Dataset is so discrete and high dimensional, which causes Points to be both dense (ass many data points have same sex, occupation, relationship etc) and roughly equal distant (thanks to the high dimensionality)

Note: Our code for the K Mean algorithm can be found in the file *pca_ kmeans_ clustering.ipynb* and the code for the DBSCAN algorithm is contained in the *pca_ kmeans_ clustering.ipynb* file. Please make sure that the dataset *adult.data* is located in the same directory from where your run the algorithms.

Parameter Tuning

How are the parameters for your algorithms set? Justify your decisions.

For both algorithms we tried some parameter values by intuition and also systematically with for example the elbow plot. We will explain more when we get to said algorithm.

Why we decided to apply dimensionality reduction using PCA?

We first tried to tune our models KMeans and DBSCAN with our dataset without dimensionality reduction. With K Means which was the better performing algorithm the best silhouette score we were able to achieve was around 0.3 which was not so good. Another problem we were facing is that some features with less number of meaningful values like the capital gain got to much importance for our algorithm. Removing this feature made our evaluation even worse.

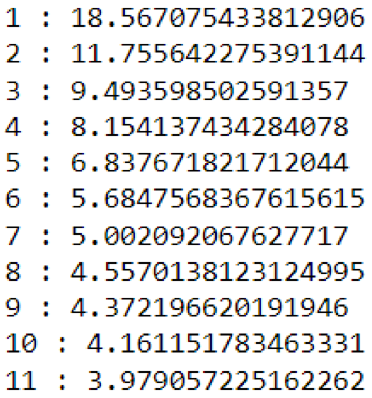
After experimenting with different subset of features with no significant improvement on our evaluation results we decided to try our with dimensionality reduction using PCA.

KMeans

We used the well known Elbow method to find the optimal number of k for the KMean algorithm on our dataset. On Figure 1 one can see that the distortion drops a lot at $k = 2$. From the plot it is not very clear at which k the distortion starts to decrease in a linear faction.

We decided to combine our observation from the elbow plot with our visual observation of the 3 Principle components we plotted on Figure 2. On this plot we can visually capture about 4 clusters.

Figure 1: Elbow plot for the K Mean algorithm applied on the first 3 PC



```
1 : 18.567075433812906
2 : 11.755642275391144
3 : 9.493598502591357
4 : 8.154137434284078
5 : 6.837671821712044
6 : 5.6847568367615615
7 : 5.002092067627717
8 : 4.5570138123124995
9 : 4.372196620191946
10 : 4.161151783463331
11 : 3.979057225162262
```

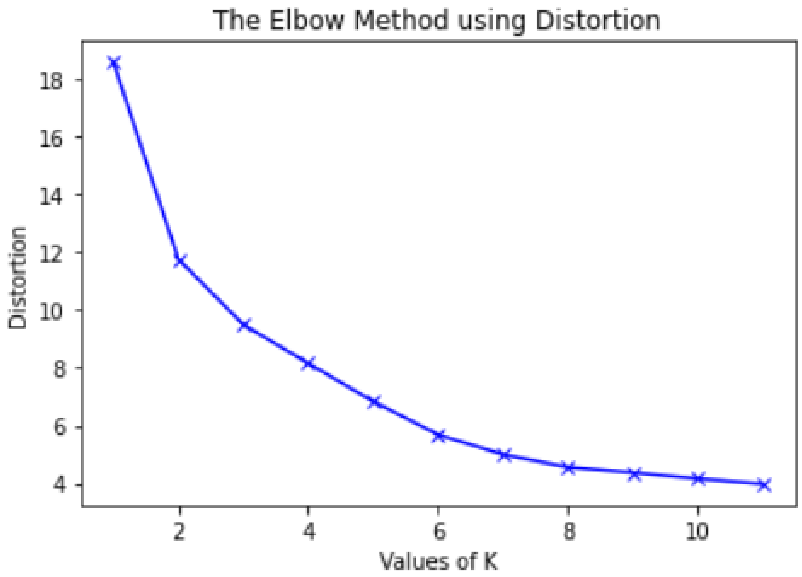
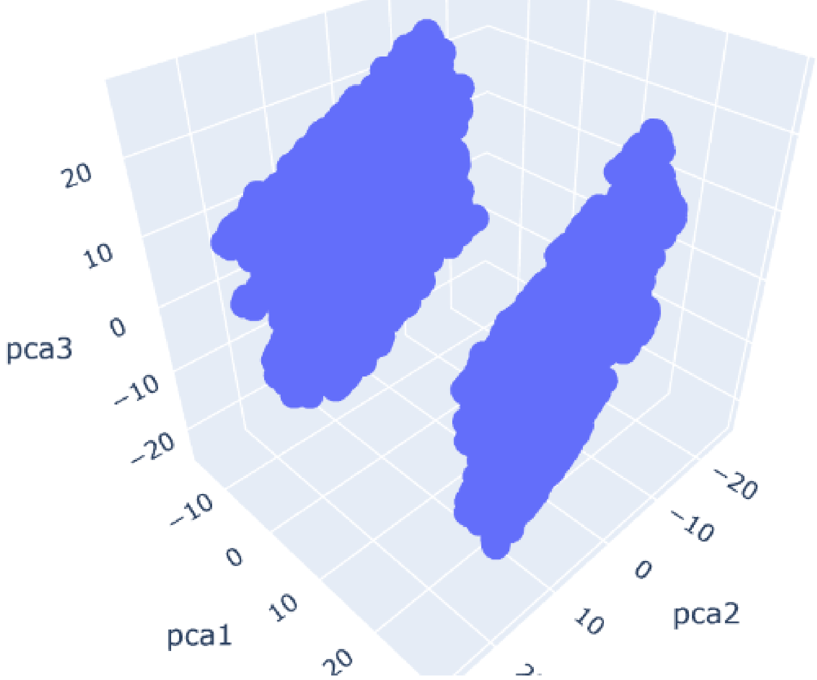


Figure 2: Elbow plot for the K Mean algorithm applied on the first 3 PC



With this observation we then tried out our K Mean algorithm from $k = 2$ to $k = 4$ and compared the internal evaluation measure with the silhouette score and the external evaluation score with the homogeneity applied for the external column income. The outcome of our evaluation we describe in the K Means part of the *valuation measures* section.

DBSCAN

So the 2 main Parameters of DBSCAN are:

- ϵ which tells us how large the Neighbourhood is around a point to determine whether it's a core, border, or noise point.
- *min_ samples* which tells us how many points need to be in a point's Neighbourhood to make it a core point.

We also can use different metrics like L-1 instead of Euclid, but we decided to stick with the boring Euclidean one to not make it too complicated.

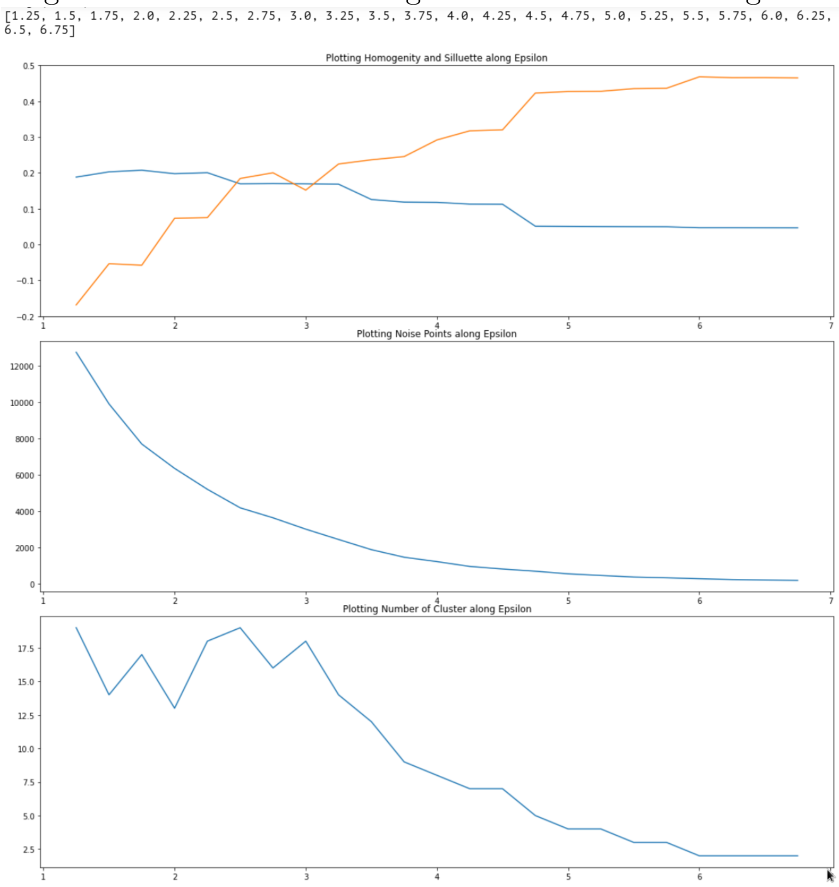
What we did was running DBSCAN with a guessed *min_ samples* for a range of ϵ and looked for the "Best". Then run DBSCAN with the found ϵ for a range of *min_ samples* and again picked the "Best". Then we repeated the process 2-3 times until both parameters converged. We especially used Silhouette (Orange) as an internal measure of how good the Clusters are and Homogeneity (Blue) as an external measure of how good the cluster encode the external over under 50k$ income label. Here the last 3 iterations:

We see in Figure 3, to small ϵ have bad silhouette, too many clusters and outliers while only homogeneity is somewhat good. At large ϵ the silhouette gets better, but that's because everything collapses into 1-2 clusters, which also is not that meaningful. Therefore, we want an ϵ somewhere in the middle around 3.2 where homogeneity is still high, but Silhouette is already decent.

We run DBSCAN with $\epsilon = 3.2$ across a range of *min_ samples* and got the results you can see in: Figure 4. Small *min_ samples* have good Silhouette but bad homogeneity, lots of clusters and don't filter noise. To get not too many Clusters, filter many outliers, have both good silhouette and homogeneity we decided to pick 75 as it corresponds to a local minimum of cluster, and is both high enough to filter much noise and has good internal and external measures.

With *min_ samples* = 75 we ran DBSCAN again with better resolution for a range of ϵ. The results can be seen in Figure 5. As we see there is a nice local drop in clusters and a local max in Silhouette at 3.15. When running DBSCAN for different *min_ samples* again with 3.12 instead of 3.2 we got basically the same as in Figure 4, which told us we found are optimal parameters and can Stop.

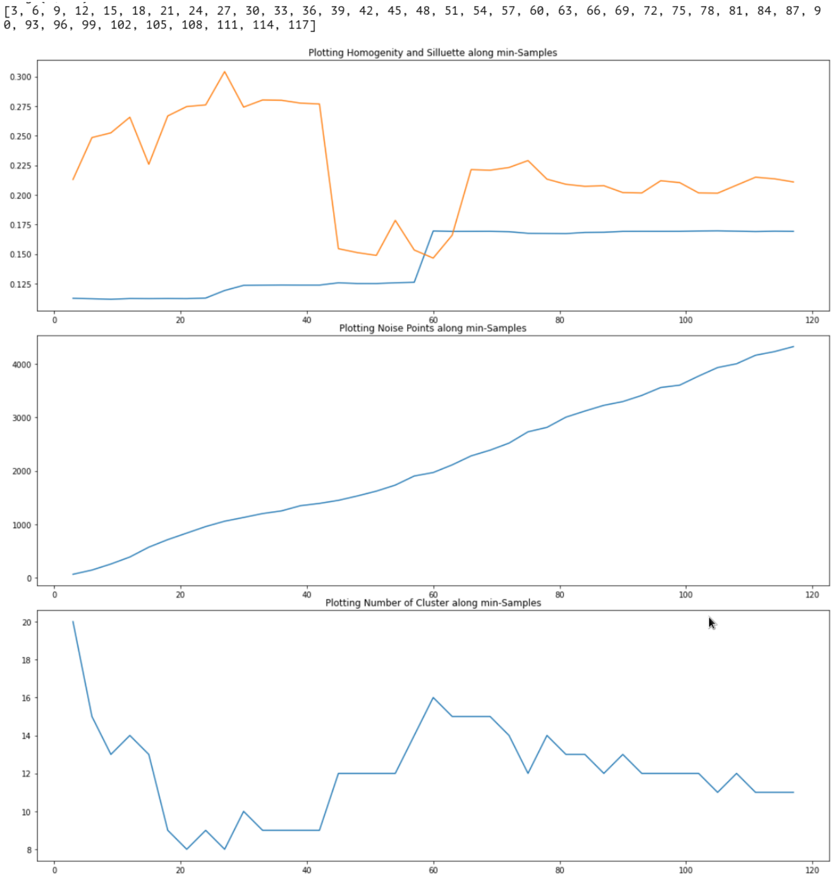
Figure 3: Different Clustering Attributes across a range of ϵ



```
[1.25, 1.5, 1.75, 2.0, 2.25, 2.5, 2.75, 3.0, 3.25, 3.5, 3.75, 4.0, 4.25, 4.5, 4.75, 5.0, 5.25, 5.5, 5.75, 6.0, 6.25,
6.5, 6.75]
```

Plotting Homogenity and Silluette along Epsilon
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
1
2
3
4
5
6
7
Plotting Noise Points along Epsilon
12000
10000
8000
6000
4000
2000
0
1
2
3
4
5
6
7
Plotting Number of Cluster along Epsilon
17.5
15.0
12.5
10.0
7.5
5.0
2.5
1
2
3
4
5
6

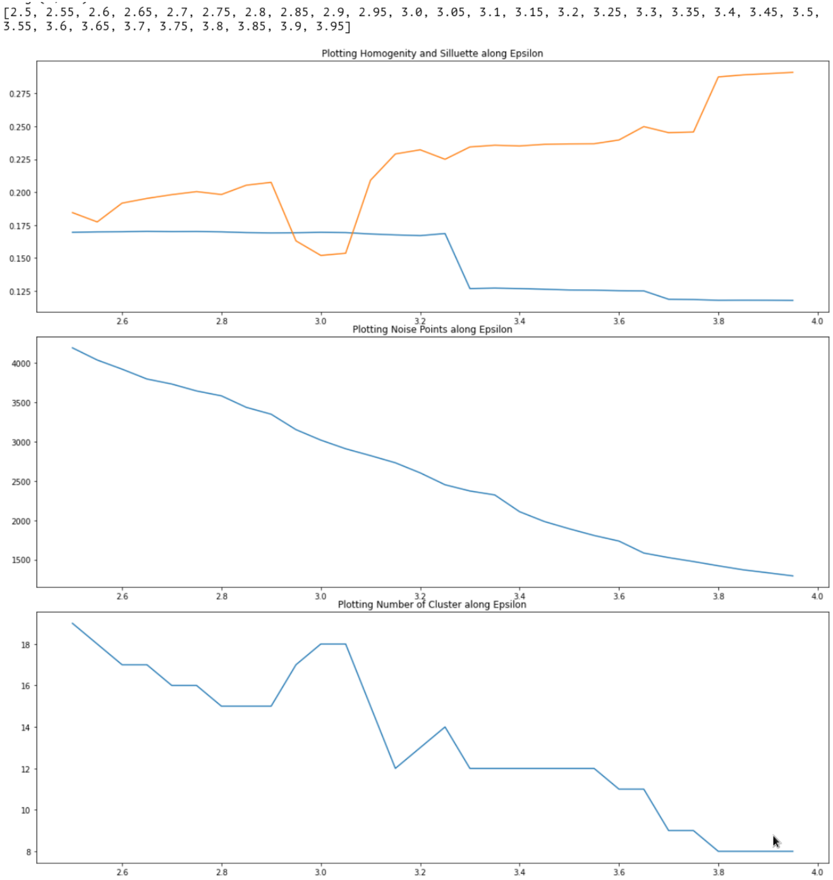
Figure 4: Different Clustering Attributes across a range of ϵ



```
[3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 9
0, 93, 96, 99, 102, 105, 108, 111, 114, 117]
```

Plotting Homogenity and Silluette along min-Samples
0.300
0.275
0.250
0.225
0.200
0.175
0.150
0.125
0
20
40
60
80
100
120
Plotting Noise Points along min-Samples
4000
3000
2000
1000
0
0
20
40
60
80
100
120
Plotting Number of Cluster along min-Samples
20
18
16
14
12
10
8
0
20
40
60
80
100
120

Figure 5: Different Clustering Attributes across a range of ϵ



```
[2.5, 2.55, 2.6, 2.65, 2.7, 2.75, 2.8, 2.85, 2.9, 2.95, 3.0, 3.05, 3.1, 3.15, 3.2, 3.25, 3.3, 3.35, 3.4, 3.45, 3.5,
3.55, 3.6, 3.65, 3.7, 3.75, 3.8, 3.85, 3.9, 3.95]
```

Evaluation measures

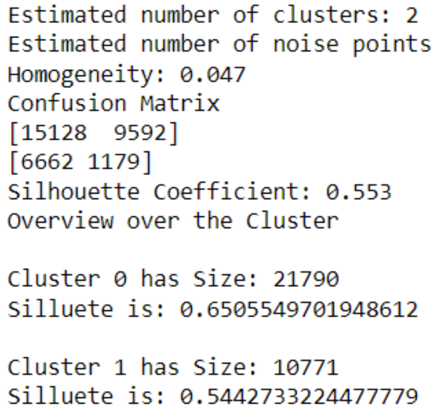
Evaluate the clustering quality using both internal and external measures (if applicable)

K Mean

The results of our internal and external evaluation metrics for $k = 2$ to $k = 4$ is shown on the Figures 6 to 8.

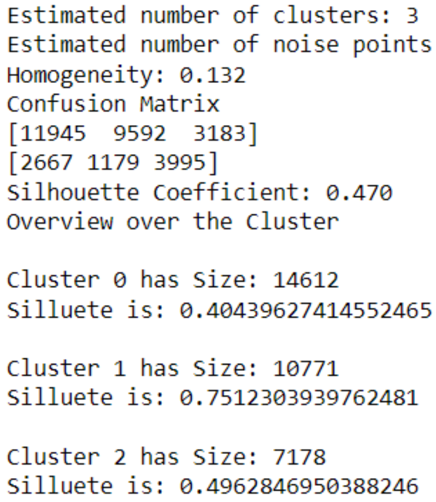
This evaluation confirms our observation from the elbow plot to some degree. The best silhouette coefficient for our clustering we archive with $k = 2$. But also the worst homogeneity. There is no much difference in regards to silhouette coefficient and homogeneity between $k = 3$ and $k = 4$.

Figure 6: Interval and external evaluation of kmean with $k = 2$



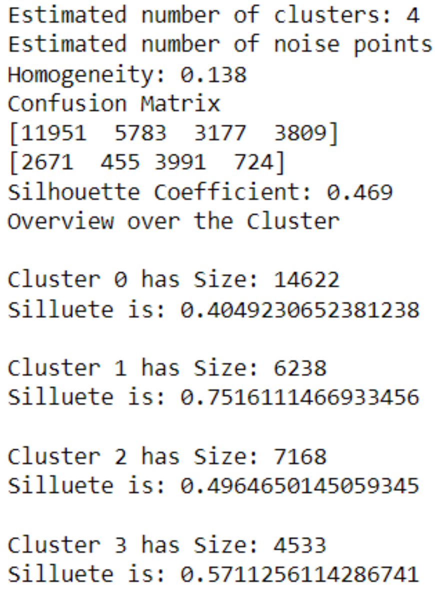
```
Estimated number of clusters: 2
Estimated number of noise points: 0
Homogeneity: 0.047
Confusion Matrix
[15128  9592]
[6662 1179]
Silhouette Coefficient: 0.553
Overview over the Cluster

Cluster 0 has Size: 21790
Silluete is: 0.6505549701948612

Cluster 1 has Size: 10771
Silluete is: 0.5442733224477779
```

Figure 7: Interval and external evaluation of kmean with $k = 3$



```
Estimated number of clusters: 3
Estimated number of noise points: 0
Homogeneity: 0.132
Confusion Matrix
[11945  9592  3183]
[2667 1179 3995]
Silhouette Coefficient: 0.470
Overview over the Cluster

Cluster 0 has Size: 14612
Silluete is: 0.40439627414552465

Cluster 1 has Size: 10771
Silluete is: 0.7512303939762481

Cluster 2 has Size: 7178
Silluete is: 0.4962846950388246
```

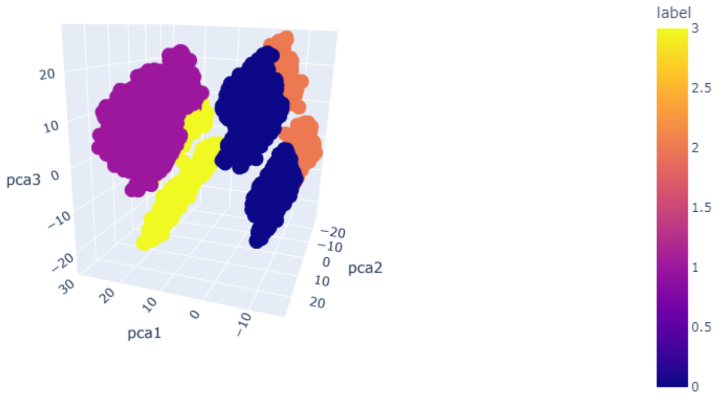
Figure 8: Interval and external evaluation of kmean with $k = 4$



```
Estimated number of clusters: 4
Estimated number of noise points: 0
Homogeneity: 0.138
Confusion Matrix
[11951  5783  3177  3809]
[2671  455 3991  724]
Silhouette Coefficient: 0.469
Overview over the Cluster

Cluster 0 has Size: 14622
Silluete is: 0.4049230652381238

Cluster 1 has Size: 6238
Silluete is: 0.7516111466933456

Cluster 2 has Size: 7168
Silluete is: 0.4964650145059345

Cluster 3 has Size: 4533
Silluete is: 0.5711256114286741
```

Because the homogeneity with $k = 4$ being the best, having one cluster with a quite high silhouette score and based on our visual observation we decided to choose $k = 4$. With this decision the clusters separation looks like on figure 9.

Figure 9: Kmeans with k = 4 applied on first 3 PC



DBSCAN

Let us evaluate the DBSCAN-Clustering with the found optimal parameters $min\_samples$ = 75, ϵ = 3.15:

- Number of clusters: 12
- Number of noise points: 2730
- Homogeneity: 0.168
- Silhouette Coefficient: 0.229

Both Homogeneity and Silhouette are positive and not completely bad, 12 Clusters are manageable and removing 2730 outliers out of a roughly 50k entry large dataset seams reasonable.

Model interpretability/visualization

Describe the resulting clusters in a human comprehensible way (labeling).

Interpretation of the labels for K-Means on the PCA components:

In the following we plot different columns of our dataset and try to interpret the labels assignments from the **KMeans** algorithm.

After we applied the K-Means algorithm on the first 3 Principle Components of the dataset we plotted different combinations of the dataset columns with the labels produced by the K Means algorithm encoded as color. In this way we were able to find for the combination of the features ***Sex***, ***Occupation*** and ***Relationship*** a notable pattern. Figure 10 shows our 3d scatter plot for these 3 features and the labels encoded in the color. On this plot 3 patterns for the distribution of the 4 labels is notable:

1. The labels are clearly separated by *gender* where the male persons got the *blue* (label id: 0) and *orange* (label id: 2) labels assigned and the female persons the *magenta* (label id: 0) and *yellow* (label id: 3) labels.
2. There is a separation of the labels along the relationship and the occupation axis notable.
3. There is a difference notable for the separation along the occupation axis between the married males and females (Husbands, Wives) and the unmarried (Own-child, Unmarried, Other-relationship). For the people *Not-in-family* all females got the same label assigned while for the males in the same relationship status there is a separation along the occupation axis.

As the features *relationship* and *occupation* are discrete, many samples overlap in this plot and there is a risk that this plot is miss-leading. Therefore we take a deeper look on the distributions along these features separately with bar and box plots.

Figure 10: 3d plot for occupation, sex and relationship features

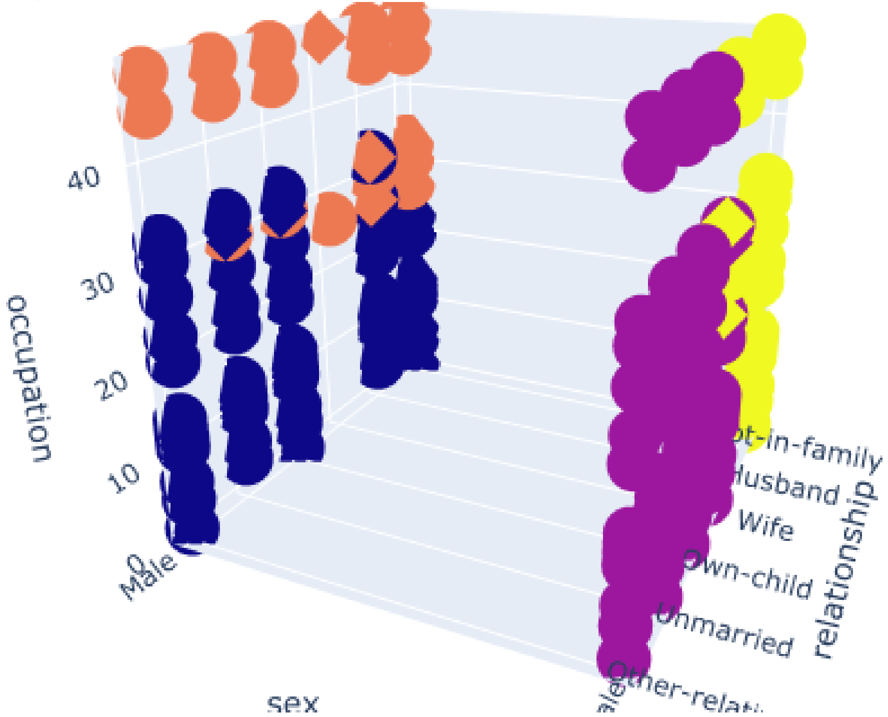
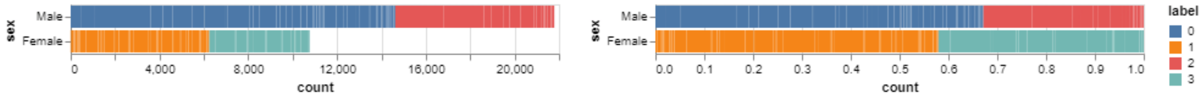


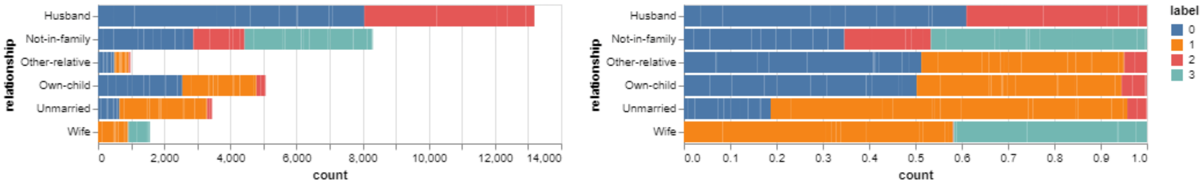
Figure 11 is showing the distribution of the labels divided by the *sex* attribute. On the absolute distribution on the left side we see the huge imbalance between the male and female samples. The normalized plot on the right side helps us comparing the proportion of the label distribution. We can confirm our observation from the 3D plot that the male and female samples got 2 distinct labels assigned. It is also notable that there is a difference in proportion.

Figure 11: Label distribution along gender attribute



On Figure 12 the label distributions along the relationship attribute can be seen. For *Husbands* and *Wives*distinct label distributions for the previous is reflected. Other than that a clear difference is notable between the relationships statuses *Unmarried, Own-child, Other-relative* which got mainly the blue (0) and orange (1) label assigned while the relationship status *Not-in-family* got more blue (0) and green labels assigned.

Figure 12: Label distribution along relationship attribute



By viewing the distribution of the labels along the different occupation (see Fig. 13) and grouping the occupations by the labels which are dominant in the distribution we can group the occupations as following (for orange (1) and green (3) we compare the proportions only between these two labels):

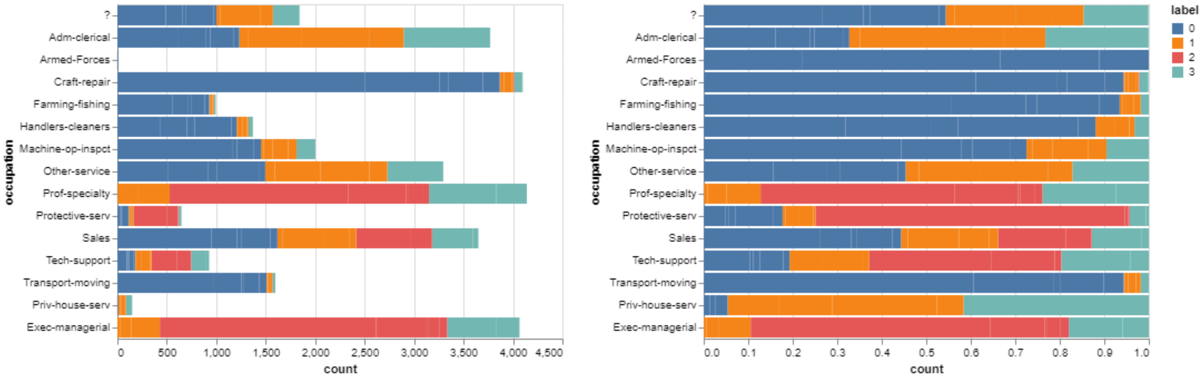
- **Blue** label is dominant: *Armed-Forces, Craft-repair, Farming-fishing, Handlers-cleaners, Machine-op-inspct, Transport-moving.*

- **Red** label is dominant: *Prof-speciality, Protective-serv, Exec-managerial, (Tech-support)*

- **Orange** label is more dominant in: *Private house-serve,?, Adm-clearical, Other-services, Machine-op-inspct, Handlers-cleaners*

- **Green** label is more dominant in: *Prof-speciality, Exec-managerial*

Figure 13: Label distribution along occupation attribute



As we noted a difference in *working hours per week* (*hpw*) in our data analysis part of the previous project between *Male* and *Female* respectively *Husband* and *Wife* we also took a close look on this feature.

On Figure 14 it is visible that for both labels which are assigned to male samples the upper quartile is concentrated obve the median (40h) while for the females the upper quartile lies on the median (40h).

By viewing the boxplot for the *Relationship* on Figure 15 this observation for males and females is reflected for the *Husbands* but for *Wives* there is a difference notable for the green label (3). The red box for the Wives are actually 2 male samples and can be ignored in this plot. Another observation is that all labels (except the red label 2) on other relationship status (except *Not-in-family*) have their upper quartile on the median at around 40h. People with the red label assigned or are males and not in a family tend to work longer.

Figure 14: Label distribution along occupation attribute

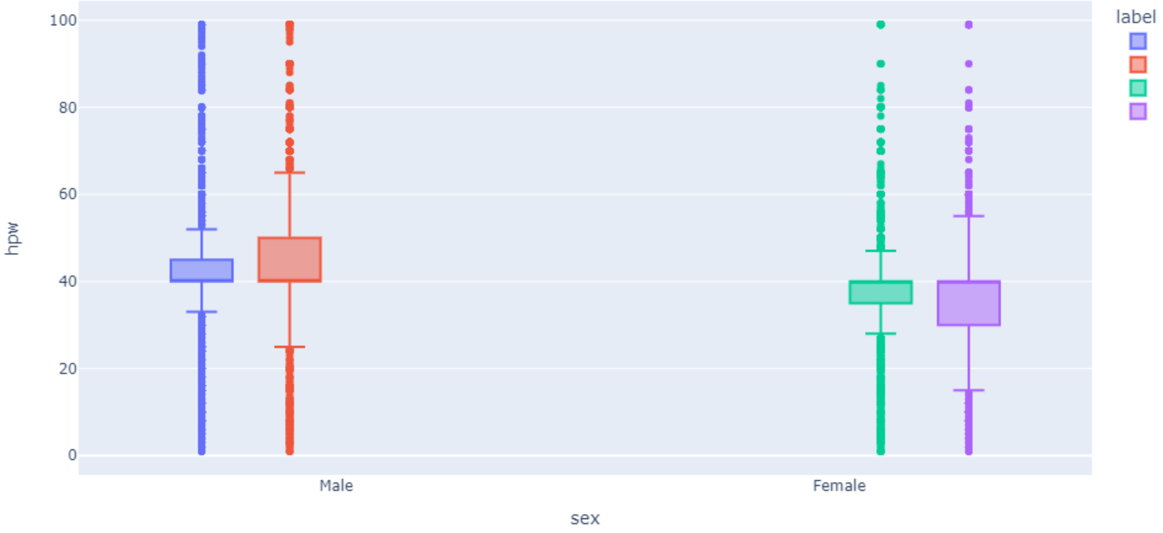
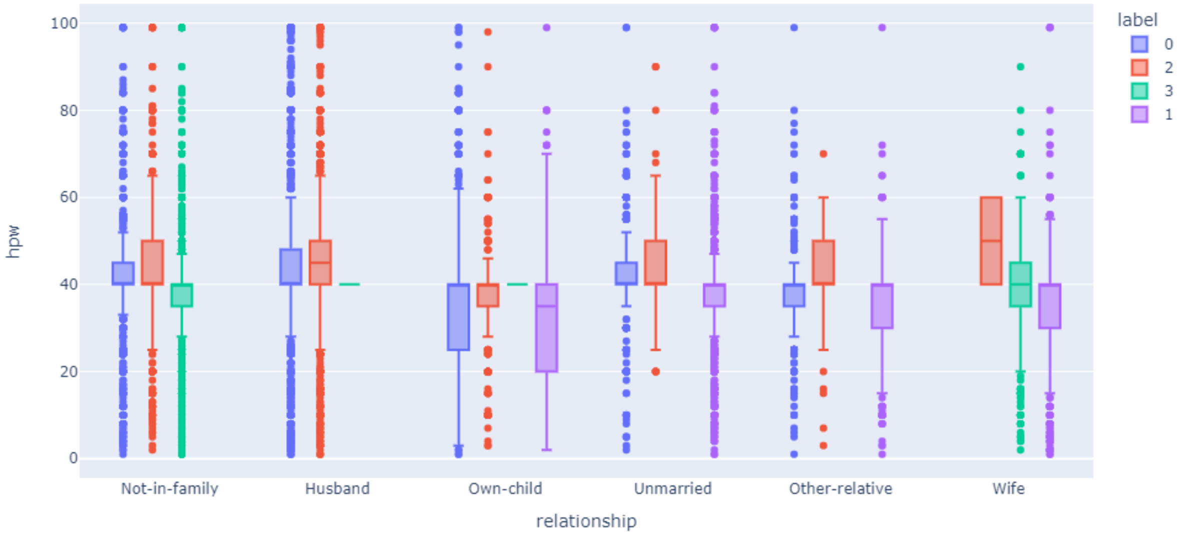
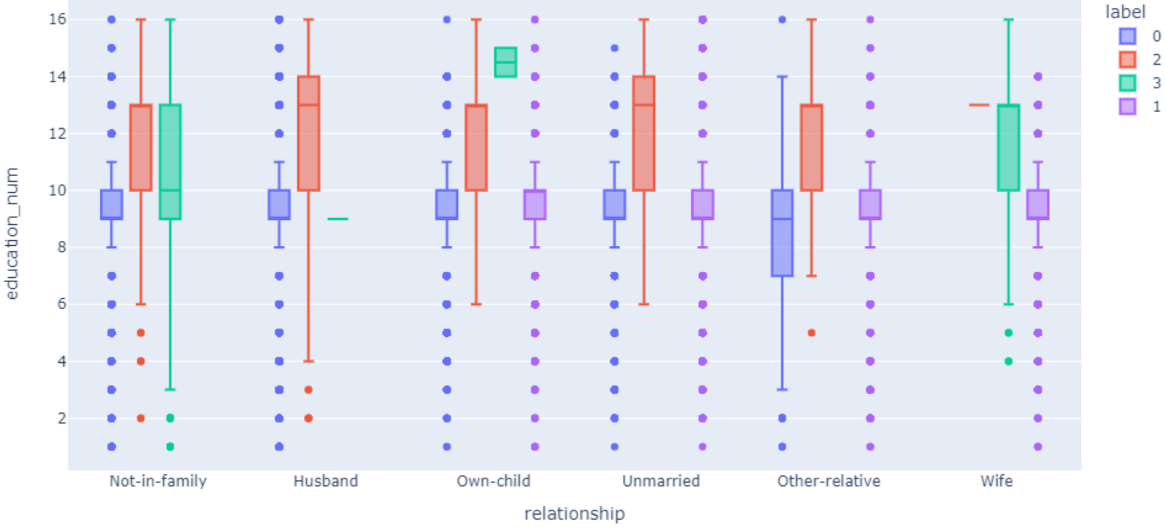


Figure 15: Label distribution along occupation attribute



From Figure 16 it is notable that People who got the red (2) and green (3) labels spent more time on education.

Figure 16: Label distribution along occupation attribute



Now after taking a closer look on label distribution for these features we can make following conclusion for the labels we got with applying KMeans on the first 3 principle components of our dataset:

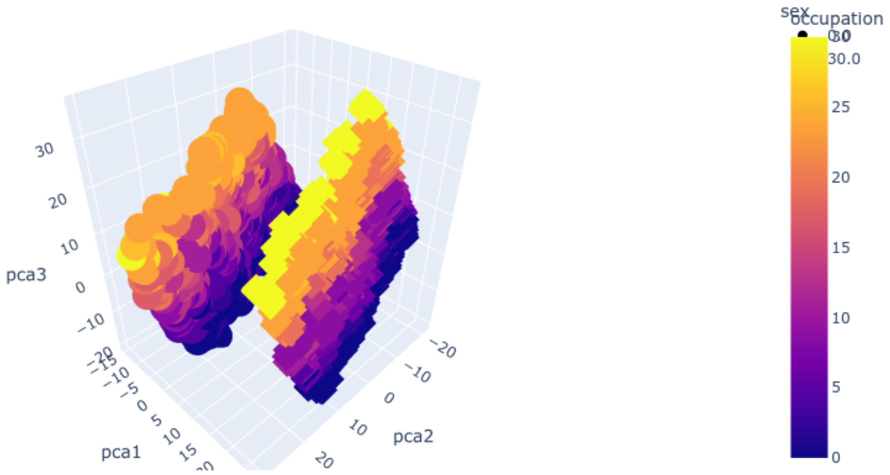
- **Blue (0)** label: Lower educated male people working in more (sterotypic masculine and physical jobs). Spending mainly the around 40h per week at work.

- **Orange (1)** label: Lower educated female people working in more (sterotypic masculine (physical) jobs) and in (sterotypic feminine jobs like private-house-serv, Adm-clearical). Spending around 40h and lesser per week at work.
- **Red (2)** label: Higher educated male people working in more managing and higher earning jobs. Spending often more the 40h per week at work.
- **Green (3)** label: Higher educated female people working in more managing and higher earning jobs. But in difference to Label red also distributed along other lower earning jobs. Working around 40h per week and lower per week.

0.1 Model interpretability/visualization of the DBSCAN

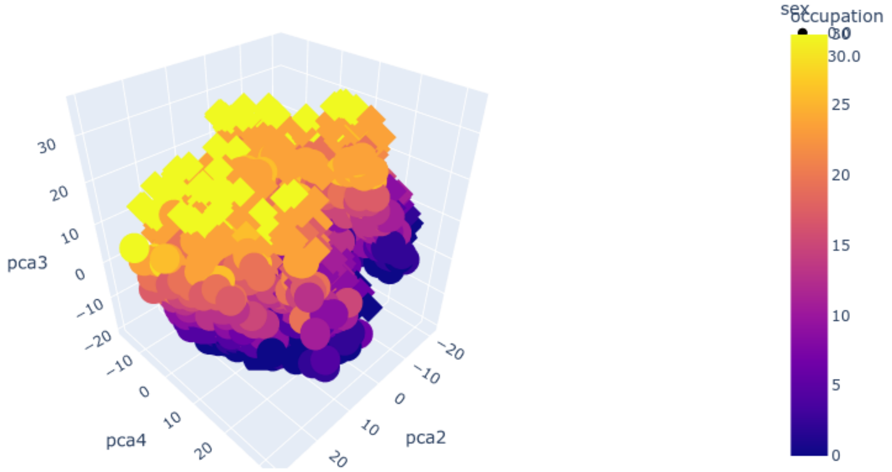
Before we describe the Clusters lets take a rough look at the Data after taking the 4 dimensional PCA:

Figure 17: PCA, colors=occupation, symbols=sex



In figure 17 we see the first 3 PCA components. The first pca1 component is basically just sex, and the pca3 component corresponds quite well with occupation.

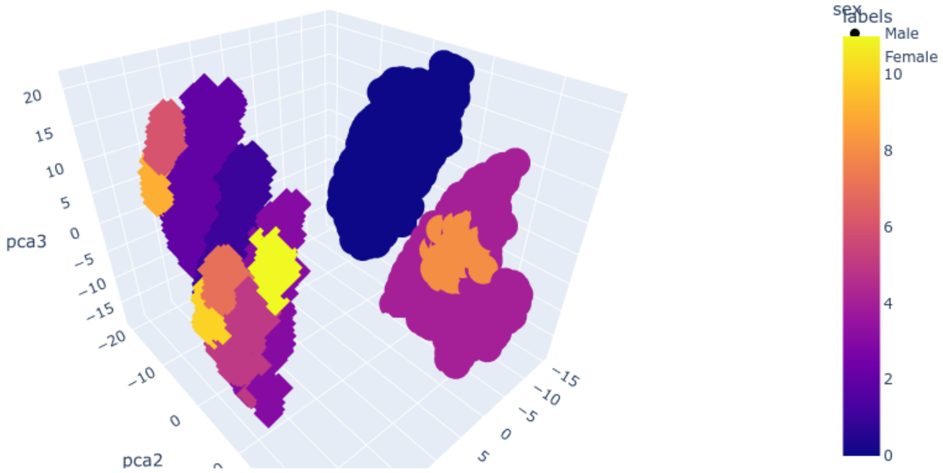
Figure 18: PCA, colors=occupation, symbols=sex



In figure 18 we see the PCA components 2,3,4. The without pca1 component the data becomes a blob. Pca4 component is as expected in a right angle to components 2 and 3.

Now let us finally start visualizing and describing the clusters we are seeing. To prevent cluttering and allow us to actually see what's going on with the clusters, we have removed the noise completely from the plots.

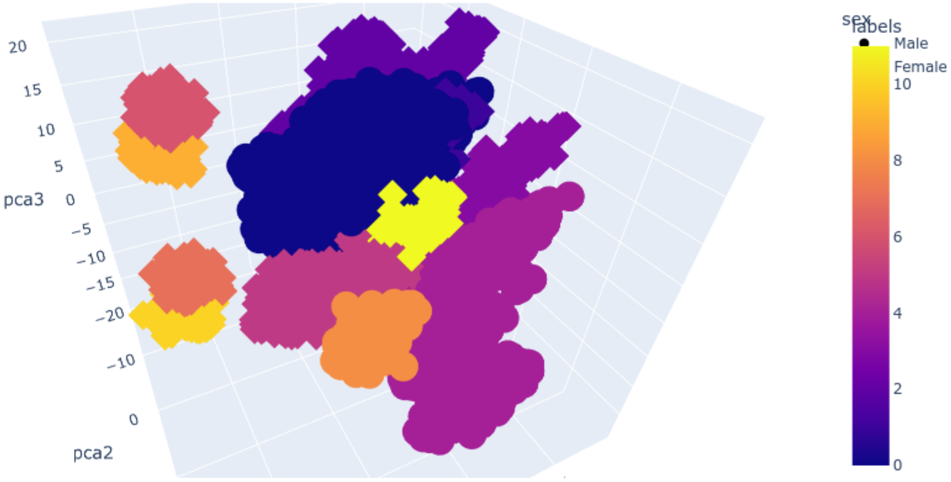
Figure 19: Clustering, colors=clusters, symbols=sex



Here we see figure 19 but coloured by the 12 clusters. We can clearly see that 3 clusters(clusters 0-blue,4-pink,8-orange) are male and the remaining 9 are female.

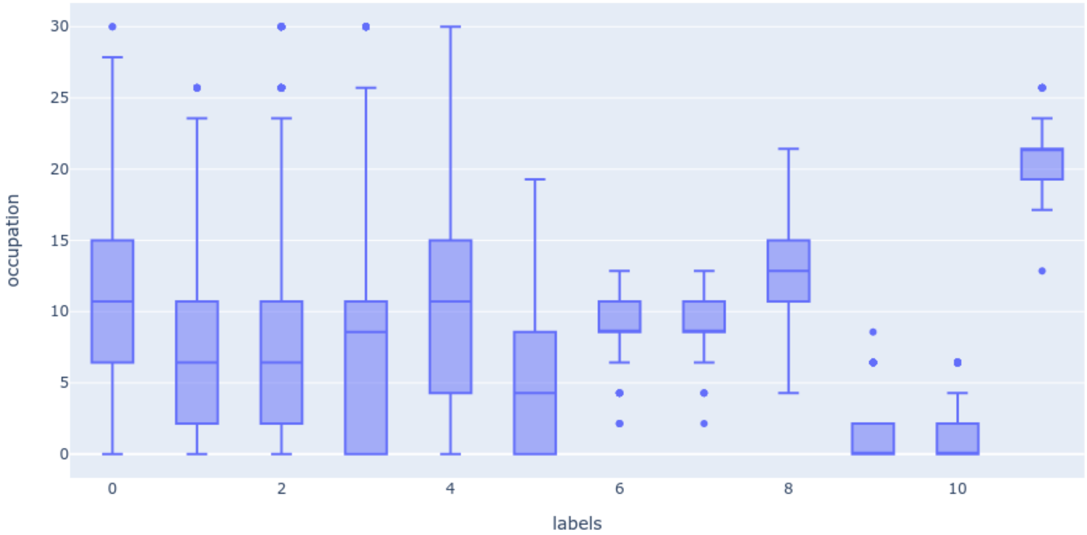
When looking at PCAs 2,3,4 instead (figure 20) we don't have such an easy identification what the clusters mean.

Figure 20: Clustering, colors=clusters, symbols=sex



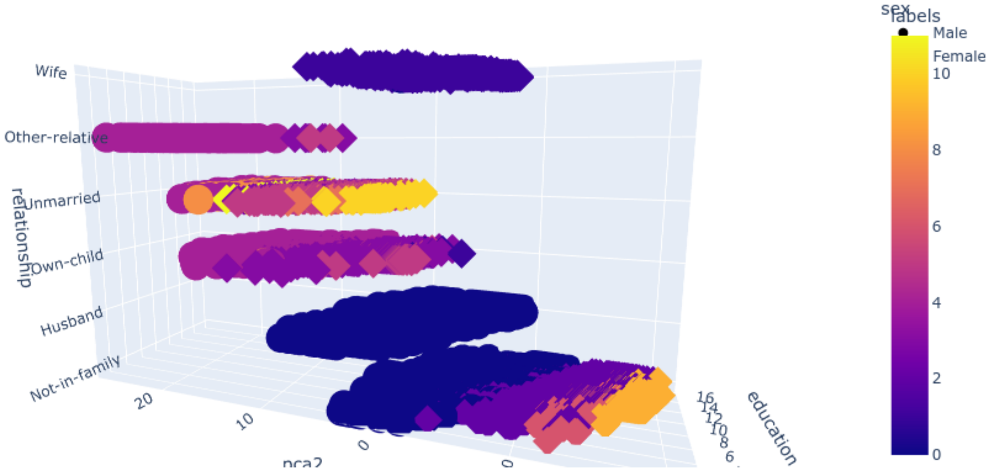
Lets plot the Occupations of the Clusters in Numerical Form (% of people making 50K$ or More) in Figure 25. We see that a few Cluster (0,4,8,11, which are all 3 Male Clusters but only 1 Female Cluster) are working in occupations with higher chance for 50k+ Income than the other clusters.

Figure 21: Clustering, Bargraph, Occupation



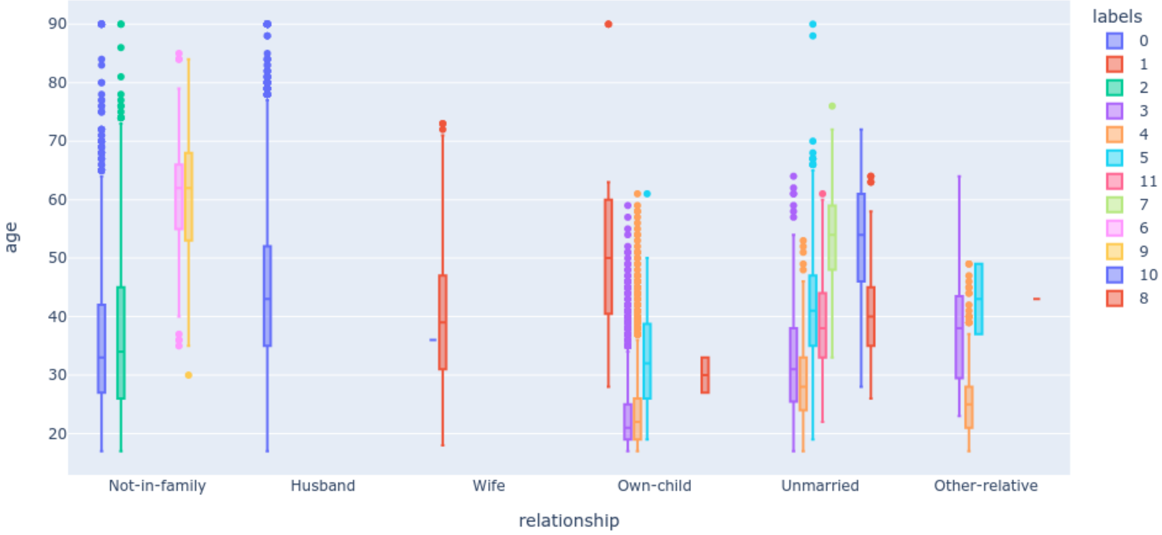
When plotting Relationship Status, Education and pca2 we can see a very interesting result in figure 22. The clusters correspond to relationship very well, as nearly each relationship slice has its own unique clusters.

Figure 22: Clustering, colors=clusters, symbols=sex



Let's investigate further by looking at a corresponding bar graph in figure 23:

Figure 23: Clustering, Bargraph, Age and Relationships

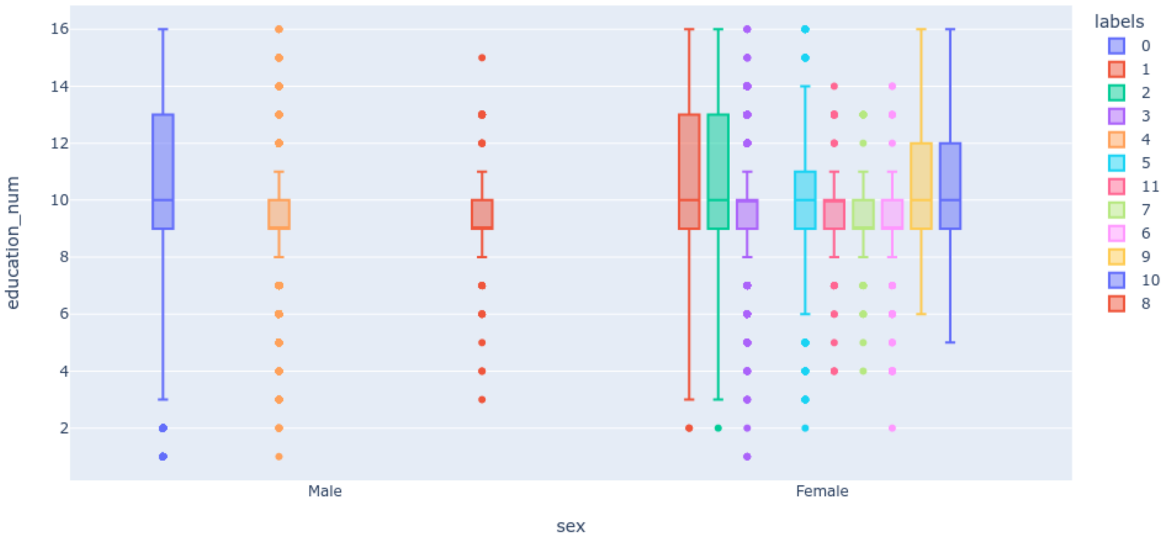


This is a lot of Information. Let's summarize what the cluster mean by combining this Barplot with the knowledge about sex from above:

- Cluster 0 are younger single Man and adult married Man.
- Cluster 1 are adult married Woman and older “Own Child” Woman.
- Cluster 2 are younger “Not in Family” Woman.
- Cluster 3 and 5 are younger "own Child", "middle aged" (≈ 30) unmarried and older "Other Relative" Woman.
- Cluster 4 are younger "own Child", "middle aged" (≈ 30) unmarried and younger "Other Relative" Man.
- Cluster 6,7,9,10 are older unmarried and not in family Woman.
- Cluster 8 are unmarried older and younger "own Child" Man.
- Cluster 11 are the unmarried rich Woman who outperform all Man clusters in Income.

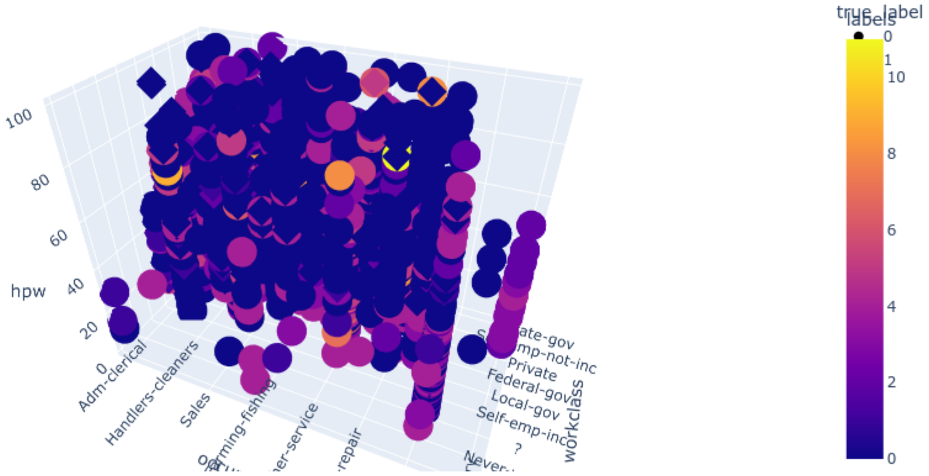
Lets also check Education in figure 24. We see that Cluster 0 (for Man) and Clusters 1,2,9,10 for Woman have a significantly higher Education level then the other Clusters have.

Figure 24: Clustering, Bargraph, Sex and Education



We have also checked many other attributes for their connection to the 12 clusters and there are some subtle connections to be found (for example Cluster 4 seams make up the majority of the work class "Never Worked" which if one includes the young age of cluster 4 suggests that its male Students and young unmarried Fathers), but most clusters become indiscernible blobs when plotted against many attributes. As a representative example of the mess we are dealing with, here Figure 25

Figure 25: Clustering, Plot, Hours per Weak, Workclass, occupation



So as a Summary. The DBSCAN clustered the Dataset up primarily by Sex and Relationship status with the other attributes only having minor effects. To a degree at which point it is hard to say whether an observation is even a direct part of the clustering or incidental. For Example, Cluster 0 works slightly more than other clusters. But this might just be a pure side effect of Cluster 0 being many middle-aged husbands, who have to support a family, and not a direct thing the clustering observed. With DBSCAN giving such messy and complicated results here, we advise using K-Means for this Dataset instead.

Protected Attributes

– Notes:

By normalizing, we are multiplying the binary male female difference by n. Now Males and females are always at least 30 apart, while other attributes like education, occupation, age etc are far more granular. Even the PCA can not overcome this so sex dominates the first component and creates 2 clear groups. A solution is to maybe normalize sex weighted with n/5 or something

martial-status and relationship are highly correlated. A Husband is married, a single is unmarried, etc. As we put both of those in both normalized with n we effectively weigh relationship 2x more then any other attribute. A solution is to weigh both with n/2 so together they add up to 1n.

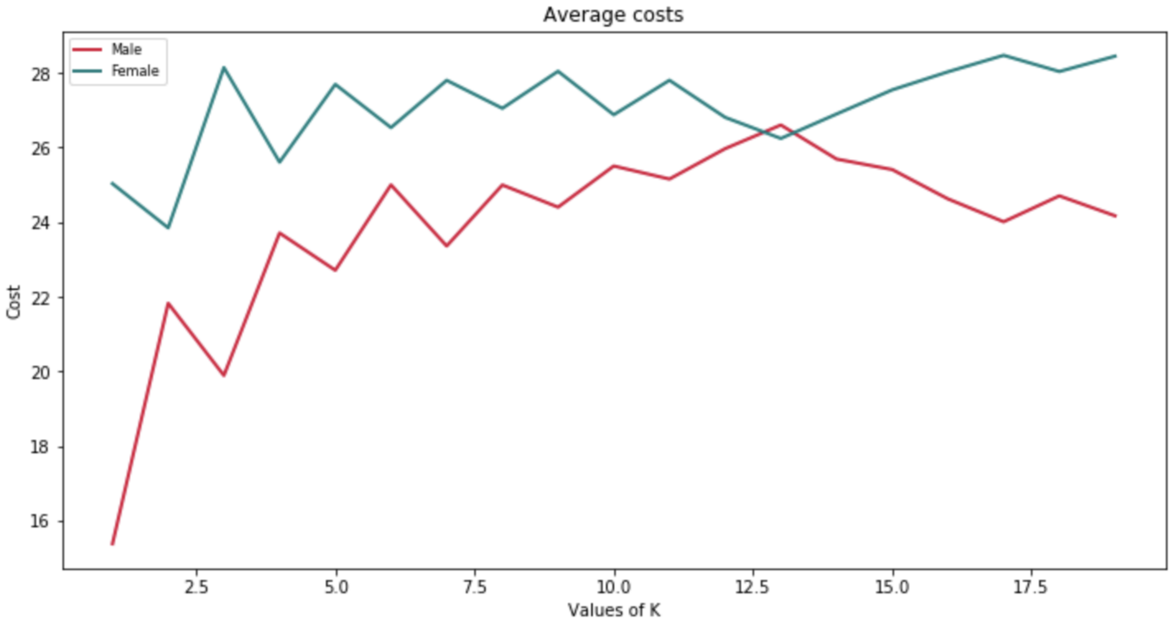
Those clear divisions encourage the clustering algorithms to divide our data along sex and relationship.

K-Means is especially discriminatory on our data as we are first dividing the group up into male/female. then we divide the males up by occupation(reasonable) and the woman up by whether they are married or not (quite sexist).

We could try the algorithms with those fixes applied but that would likely result in even worse clusters with worse silhouettes etc as the Data would be even more clumped and it would be even harder to divide the Data.

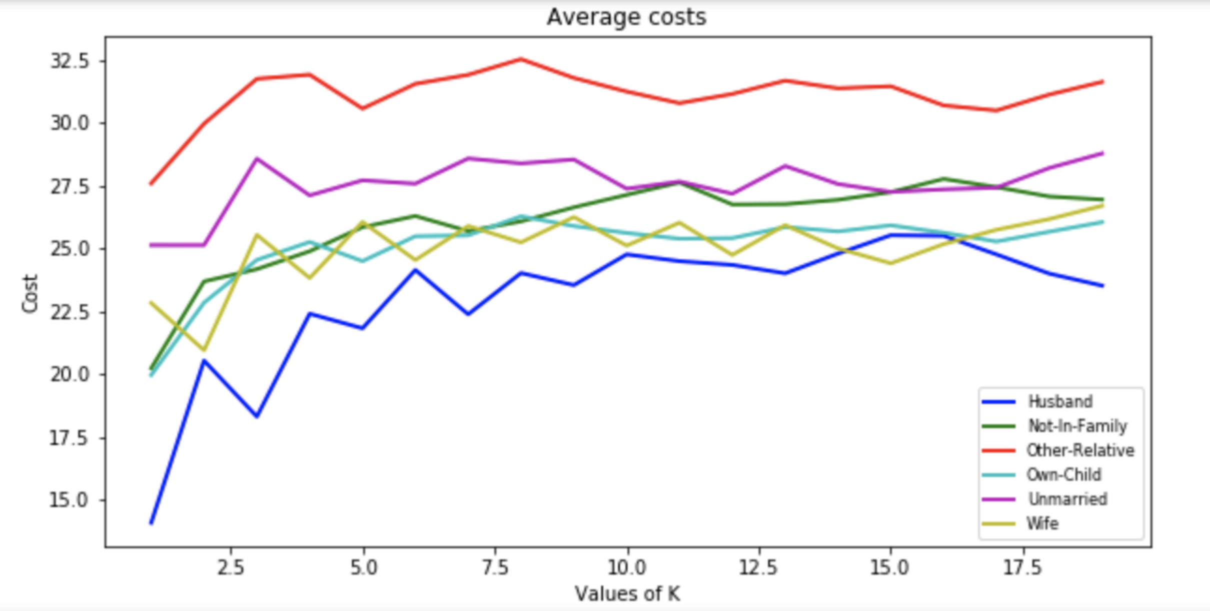
Other than these observations, we can gain an insight of (un)fairness by calculating and comparing respective average cost functions of features.

Figure 26: Sex and Cost Function



As we can see in the plot above, average cost of clustering for ’Female’ is greater than average cost of clustering for ’Male’ in any ’K’ value. We can count this observation as a negative outcome on ’Female’ group.

Figure 27: Relationship and Cost Function



As we can also observe from the plot above, while 'husband' has an advantage among the whole population, 'other-relative' has a disadvantage.