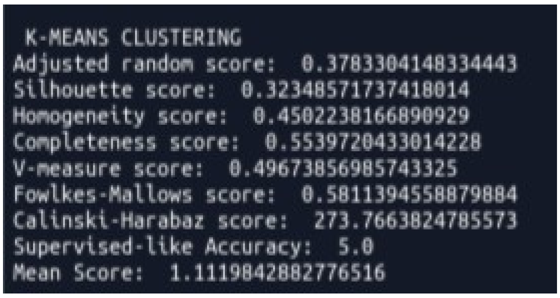
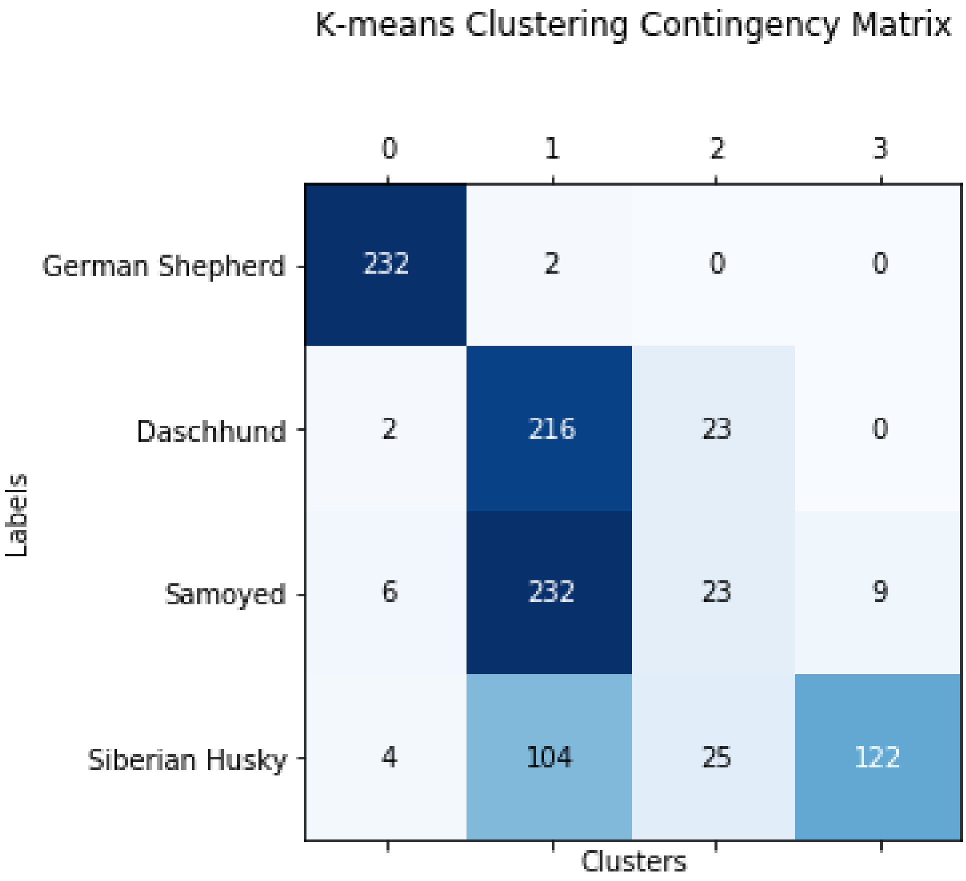
Tema 2 – Bonus: Dog Breeds
Dumitrescu Gabriel Horia

I)Clustering

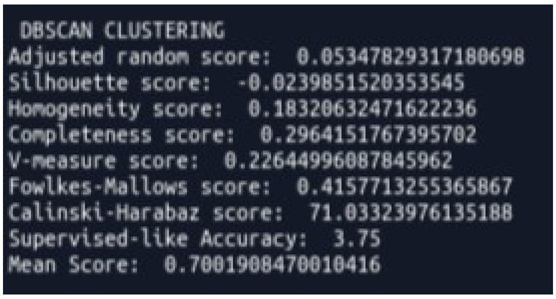
a)Kmeans



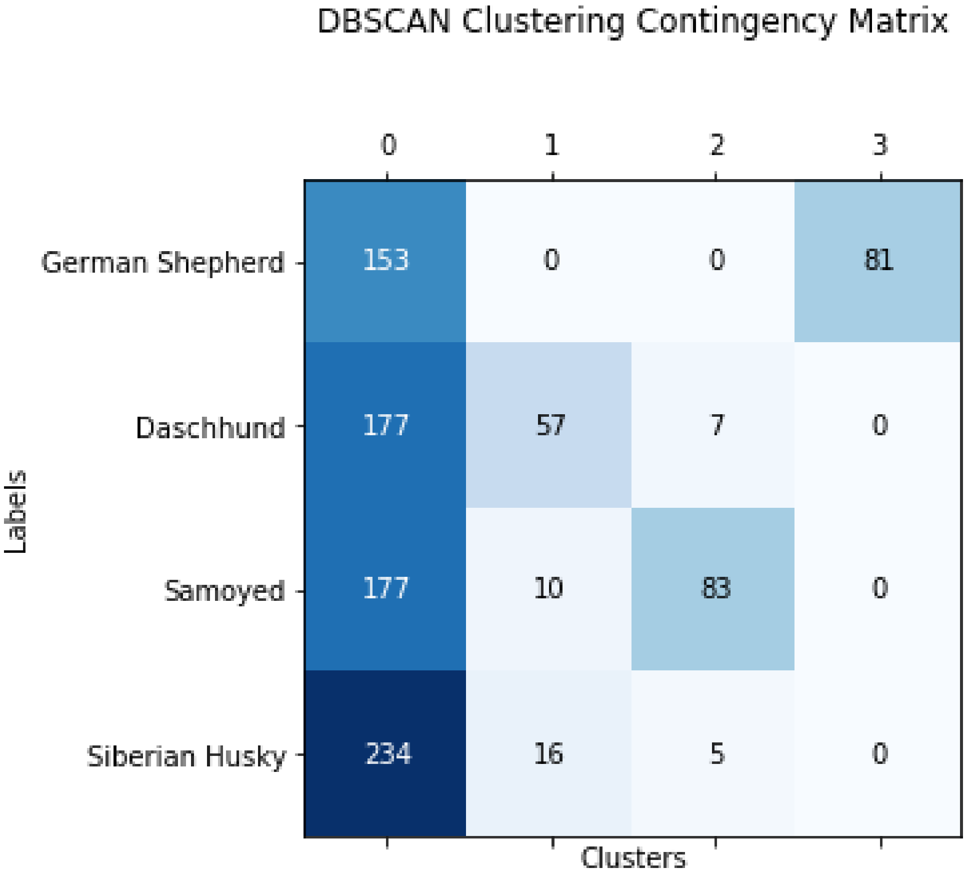
```
 K-MEANS CLUSTERING
Adjusted random score:  0.3783304148334443
Silhouette score:  0.32348571737418014
Homogeneity score:  0.4502238166890929
Completeness score:  0.5539720433014228
V-measure score:  0.49673856985743325
Fowlkes-Mallows score:  0.5811394558879884
Calinski-Harabaz score:  273.7663824785573
Supervised-like Accuracy:  5.0
Mean Score:  1.1119842882776516
```



b)DBSCAN



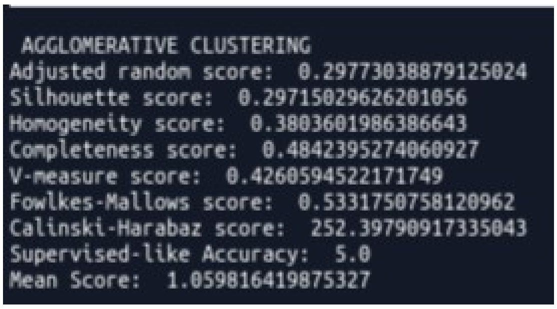
```
 DBSCAN CLUSTERING
Adjusted random score:  0.05347829317180698
Silhouette score:  -0.0239851520353545
Homogeneity score:  0.18320632471622236
Completeness score:  0.2964151767395702
V-measure score:  0.22644996087845962
Fowlkes-Mallows score:  0.4157713255365867
Calinski-Harabaz score:  71.03323976135188
Supervised-like Accuracy:  3.75
Mean Score:  0.7001908470010416
```



DBSCAN Clustering Contingency Matrix

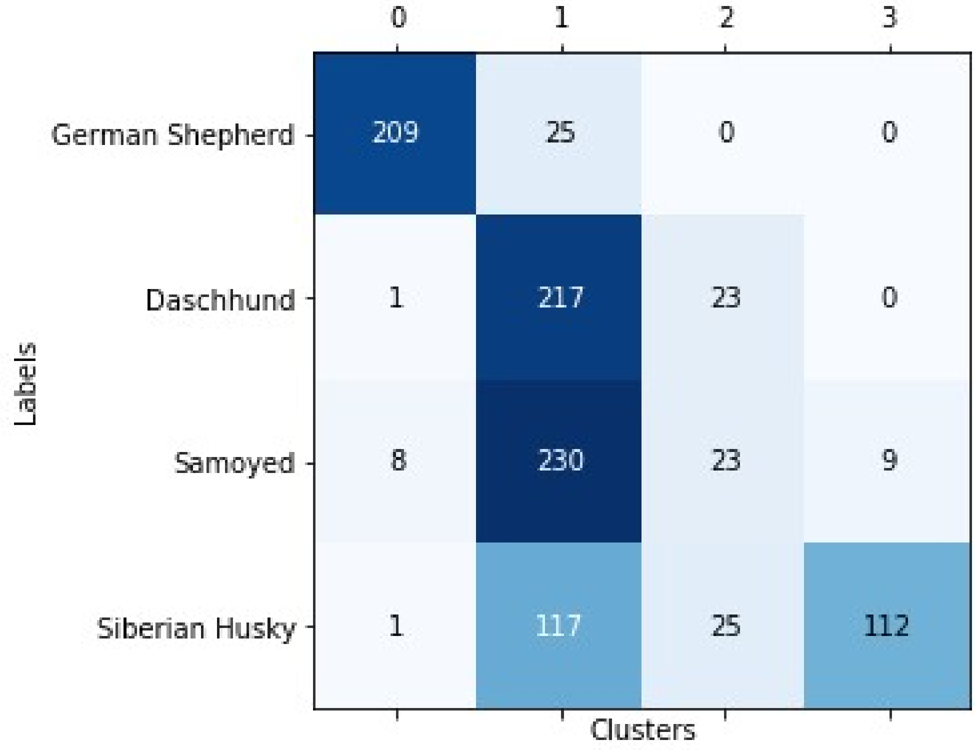
	0	1	2	3
German Shepherd	153	0	0	81
Daschhund	177	57	7	0
Samoyed	177	10	83	0
Siberian Husky	234	16	5	0

c)Agglomerative



```
 AGGLOMERATIVE CLUSTERING
Adjusted random score:  0.29773038879125024
Silhouette score:  0.29715029626201056
Homogeneity score:  0.3803601986386643
Completeness score:  0.4842395274060927
V-measure score:  0.4260594522171749
Fowlkes-Mallows score:  0.5331750758120962
Calinski-Harabaz score:  252.39790917335043
Supervised-like Accuracy:  5.0
Mean Score:  1.059816419875327
```

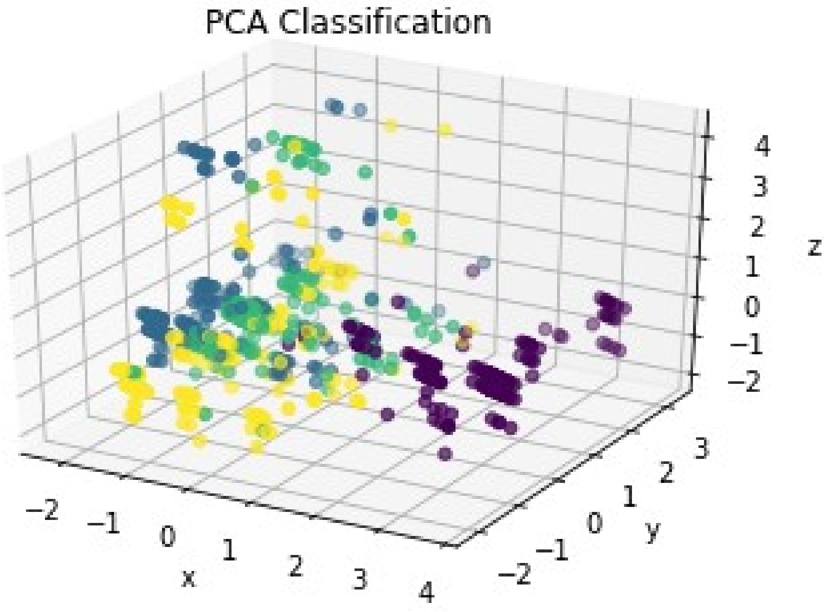
Agglomerative Clustering Contingency Matrix

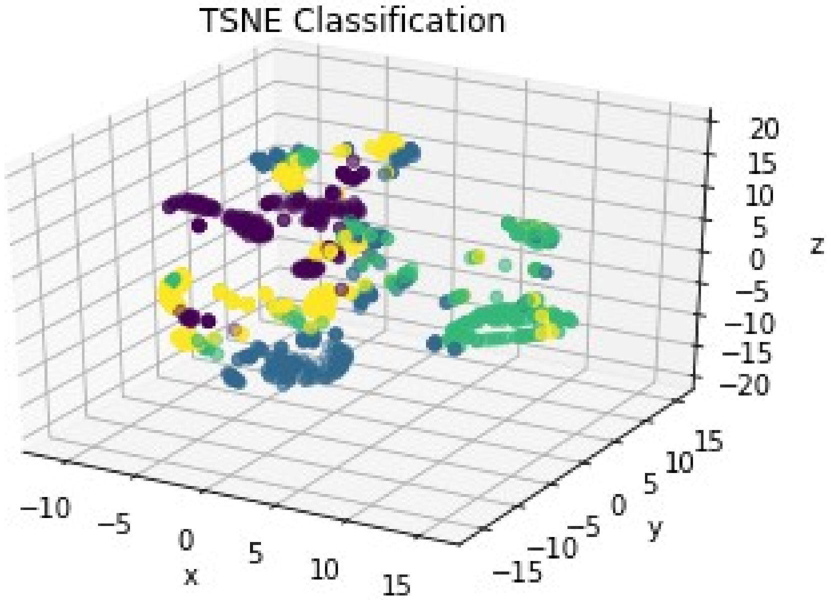


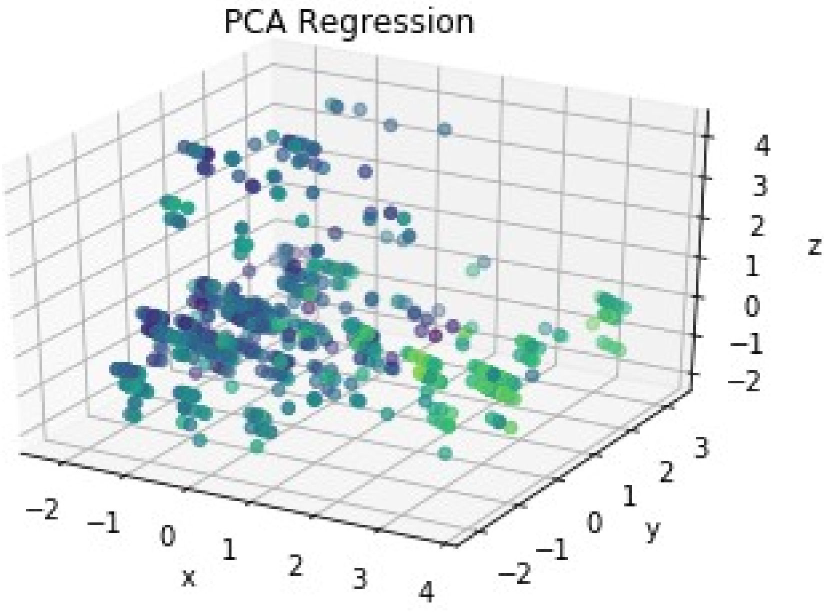
	0	1	2	3
German Shepherd	209	25	0	0
Daschhund	1	217	23	0
Samoyed	8	230	23	9
Siberian Husky	1	117	25	112

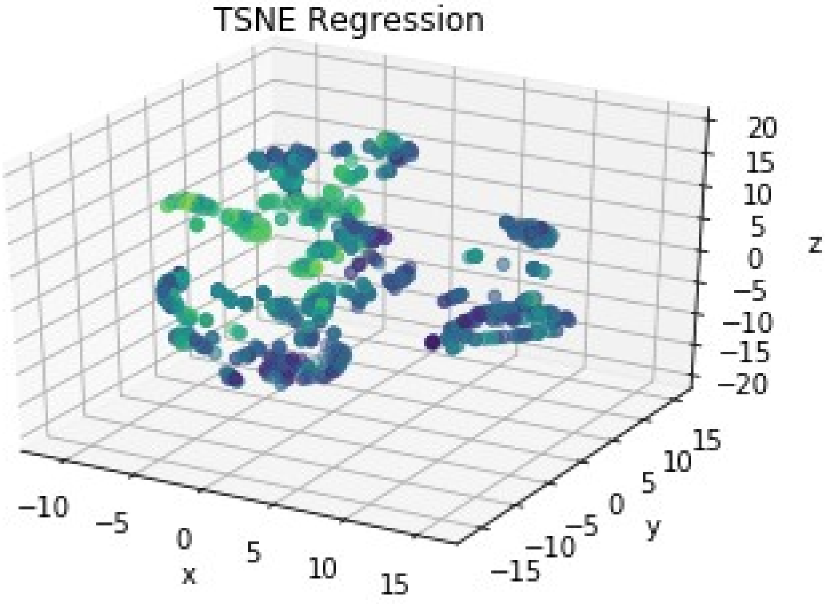
We can observe all clusterings improved, but this time, kmeans is the best one. Except German Shepherd, everything is hard to discriminate.

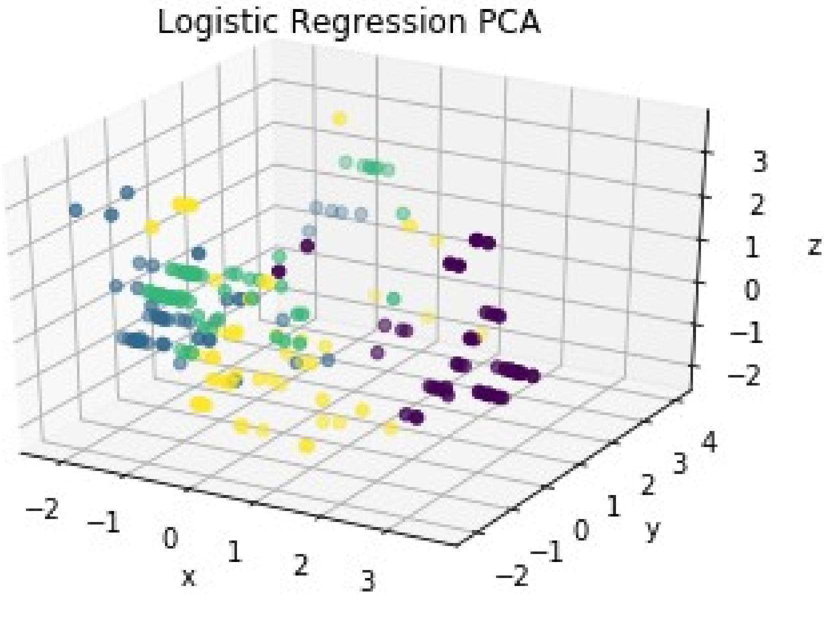
II) Data Visualization

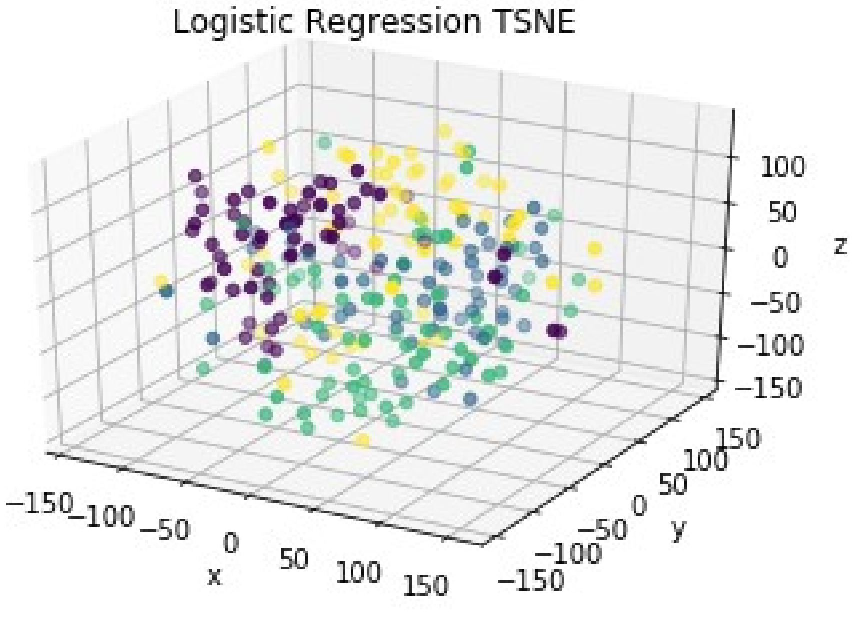


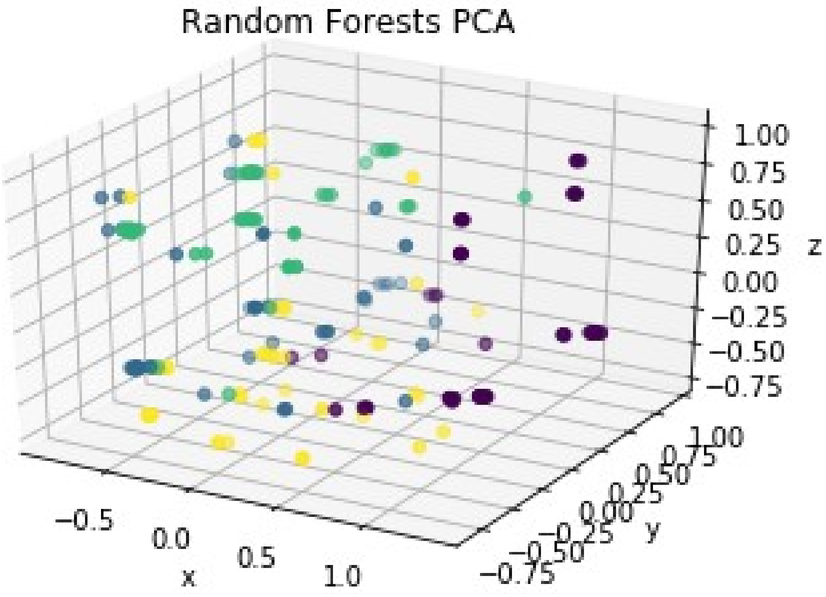


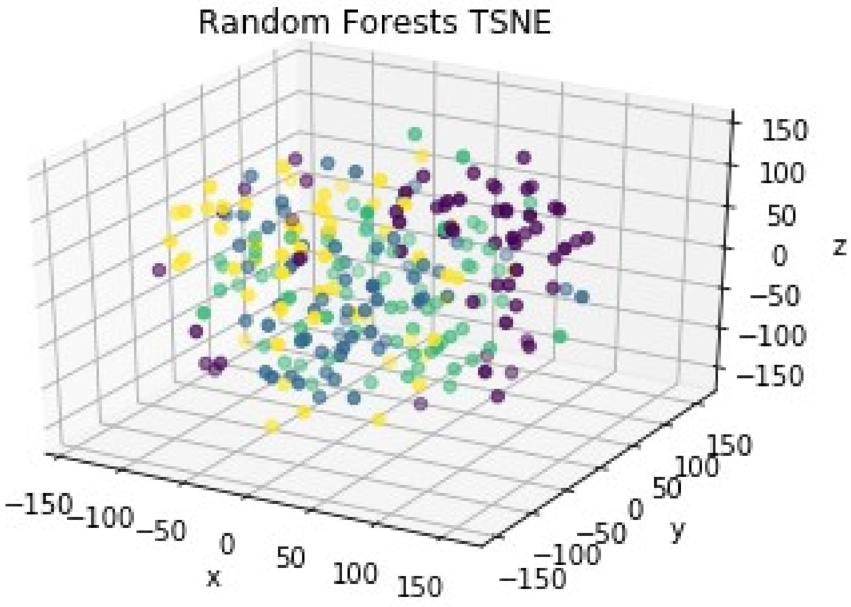

PCA Regression
x
−2
−1
0
1
2
3
4
y
−2
−1
0
1
2
3
z
−2
−1
0
1
2
3
4

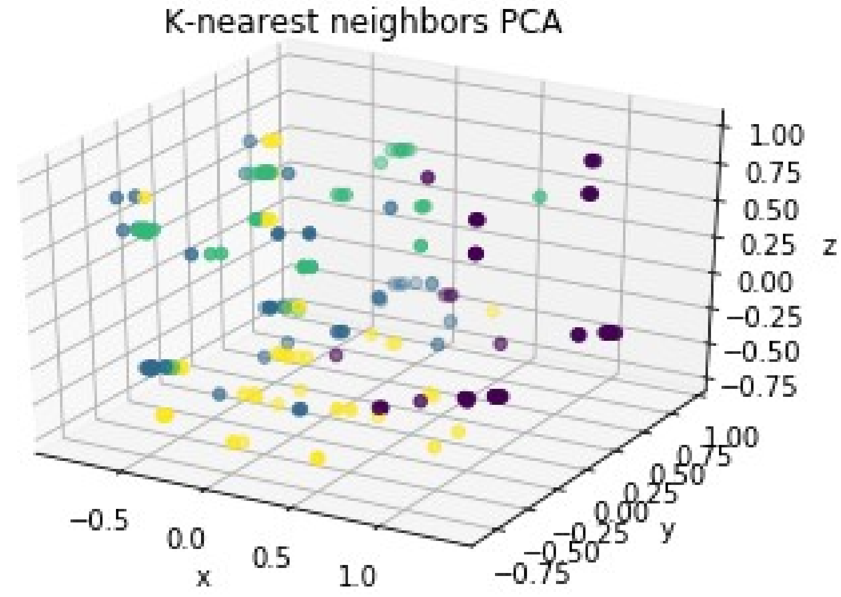

TSNE Regression
x
−10
−5
0
5
10
15
y
−15
−10
−5
0
5
10
15
z
−20
−15
−10
−5
0
5
10
15
20

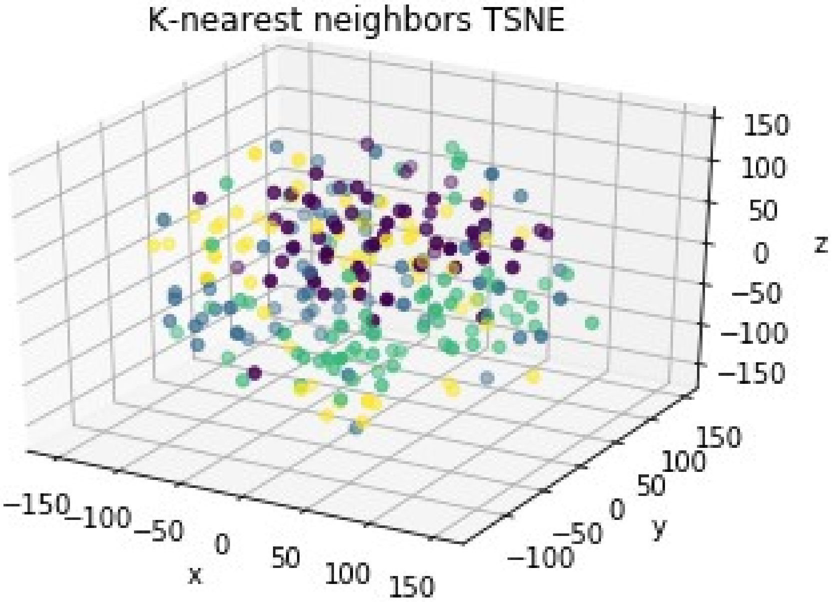

Logistic Regression PCA
x
−2
−1
0
1
2
3
y
−2
−1
0
1
2
3
4
z
−2
−1
0
1
2
3

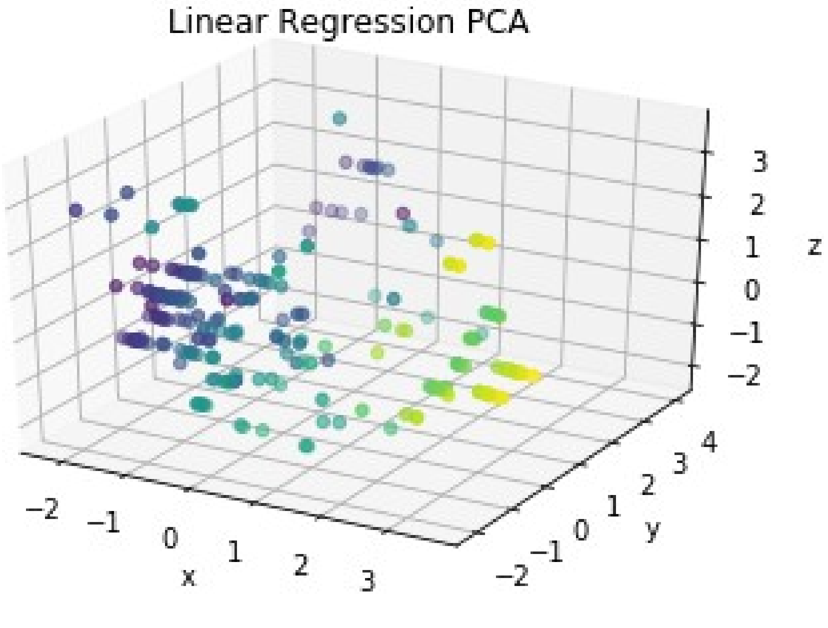

Logistic Regression TSNE
x
−150
−100
−50
0
50
100
150
y
−150
−100
−50
0
50
100
150
z
−150
−100
−50
0
50
100


Random Forests PCA
1.00
0.75
0.50
0.25
z
0.00
−0.25
−0.50
−0.75
1.00
0.75
0.50
0.25
y
0.00
−0.25
−0.50
−0.75
−0.5
0.0
0.5
1.0
x

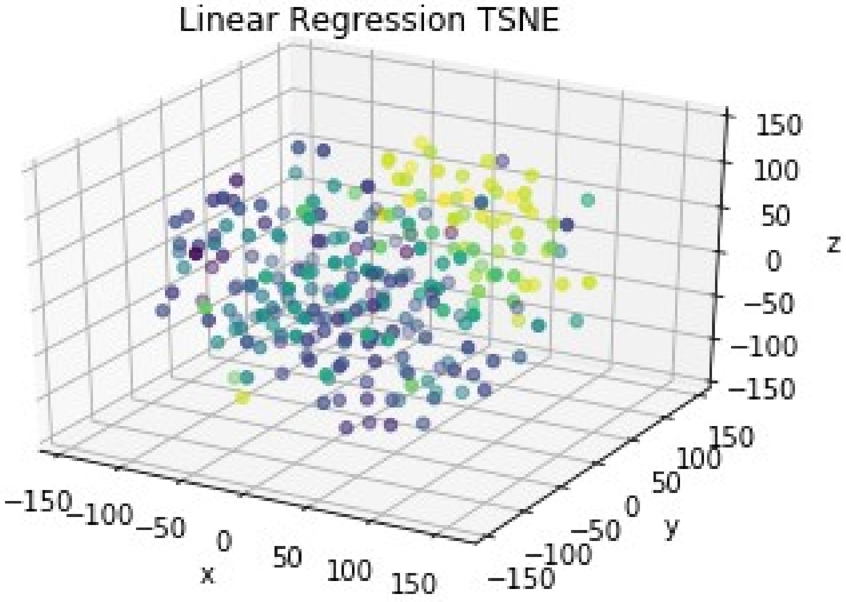

Random Forests TSNE
150
100
50
0
z
−50
−100
−150
150
100
50
0
y
−50
−100
−150
−150
−100
−50
0
50
100
150
x


K-nearest neighbors PCA
x
−0.5
0.0
0.5
1.0
y
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
z
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

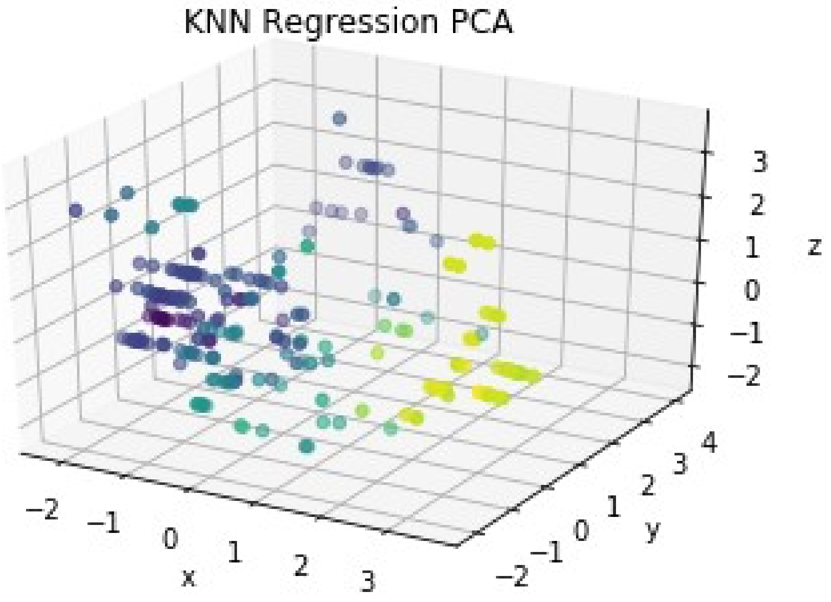

K-nearest neighbors TSNE
x
−150
−100
−50
0
50
100
150
y
−100
−50
0
50
100
150
z
150
100
50
0
−50
−100
−150

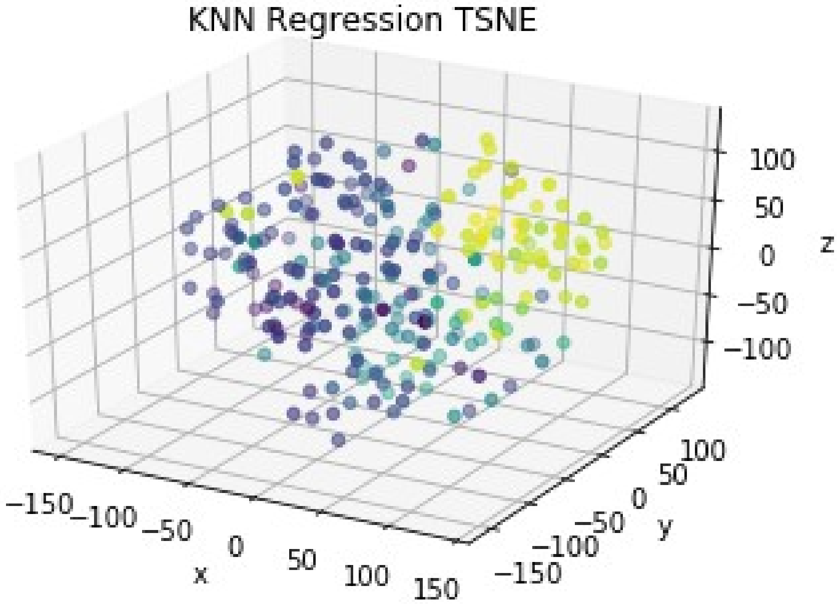


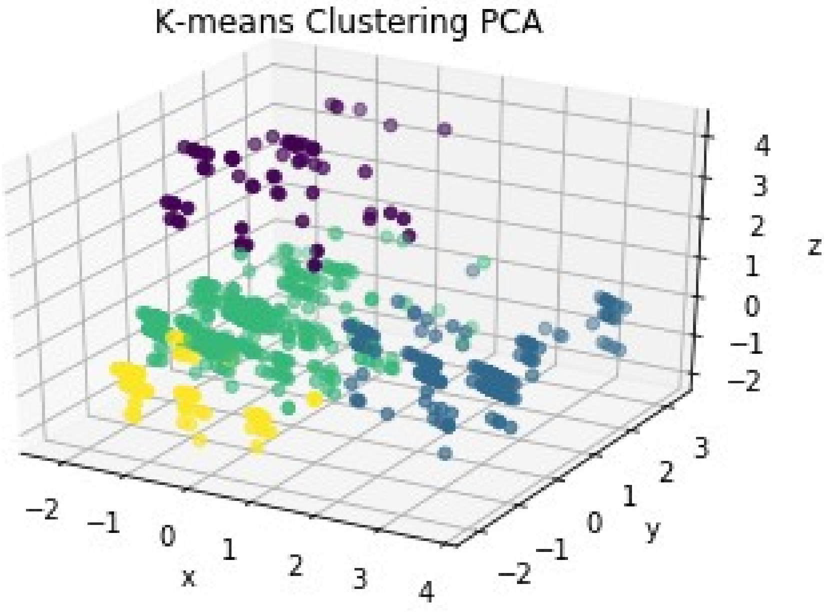
Linear Regression PCA

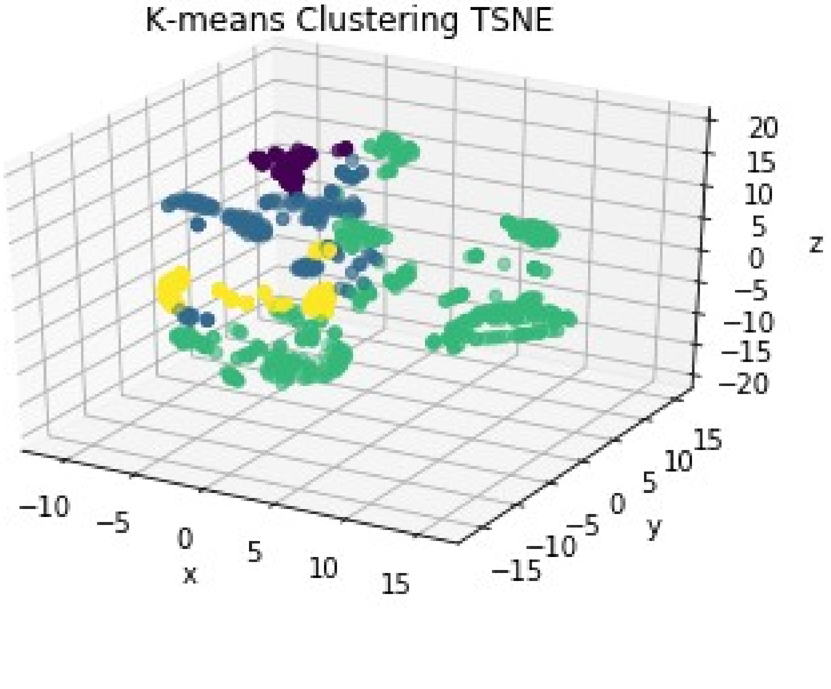


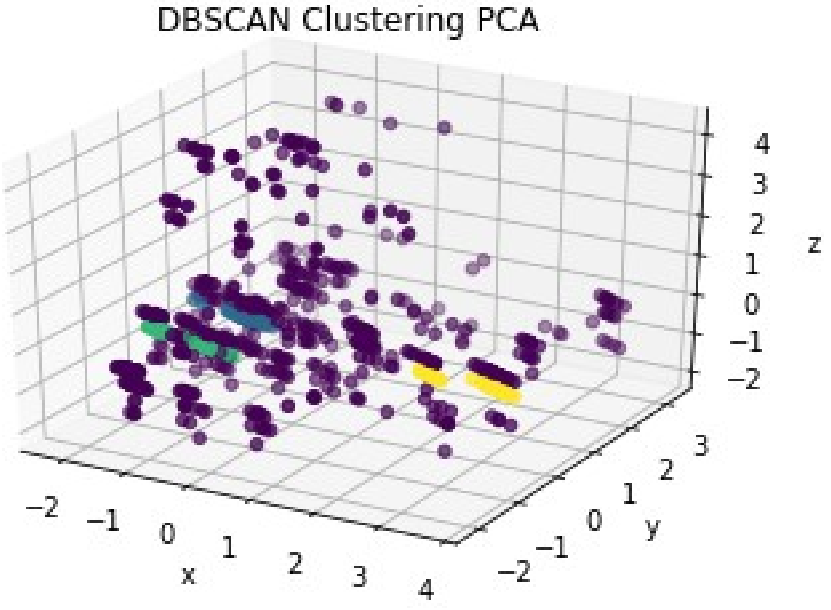
Linear Regression TSNE

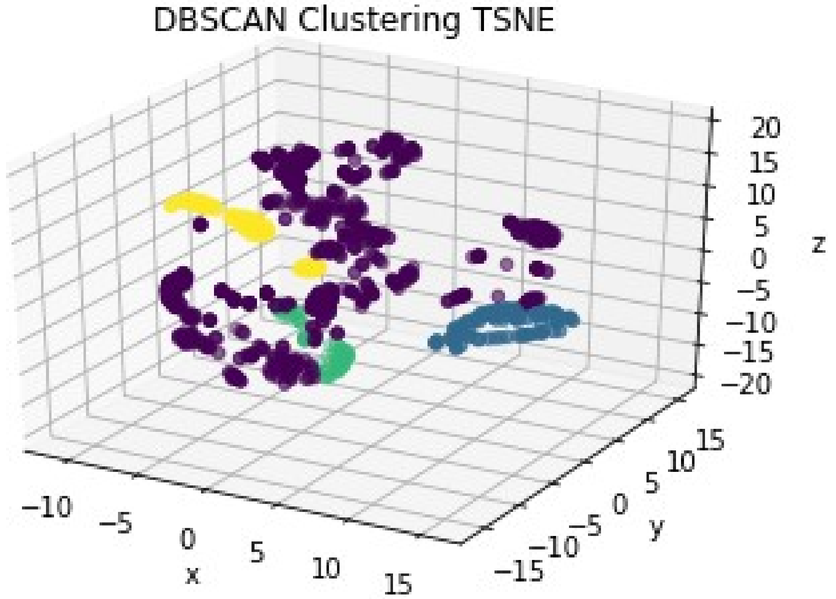

KNN Regression PCA
x
−2
−1
0
1
2
3
y
−2
−1
0
1
2
3
4
z
−2
−1
0
1
2
3

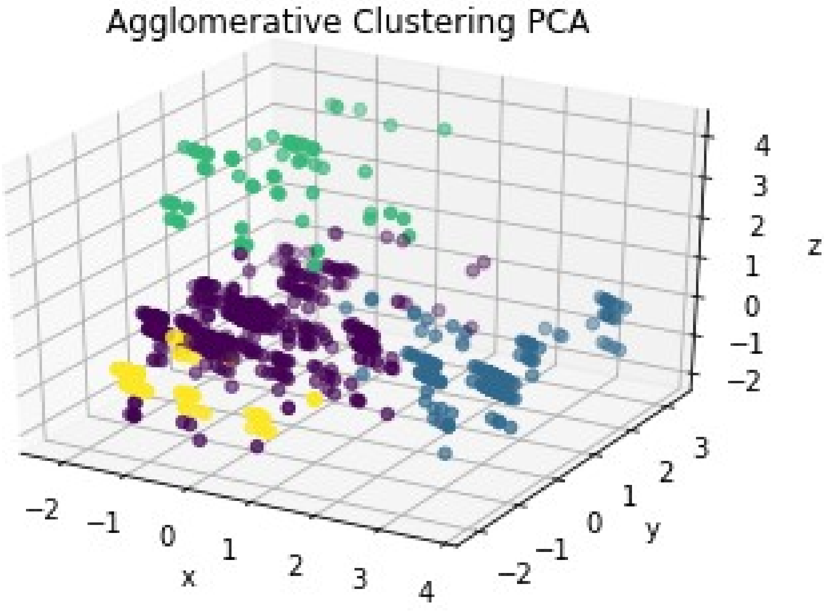

KNN Regression TSNE
x
−150
−100
−50
0
50
100
150
y
−150
−100
−50
0
50
100
z
−100
−50
0
50
100

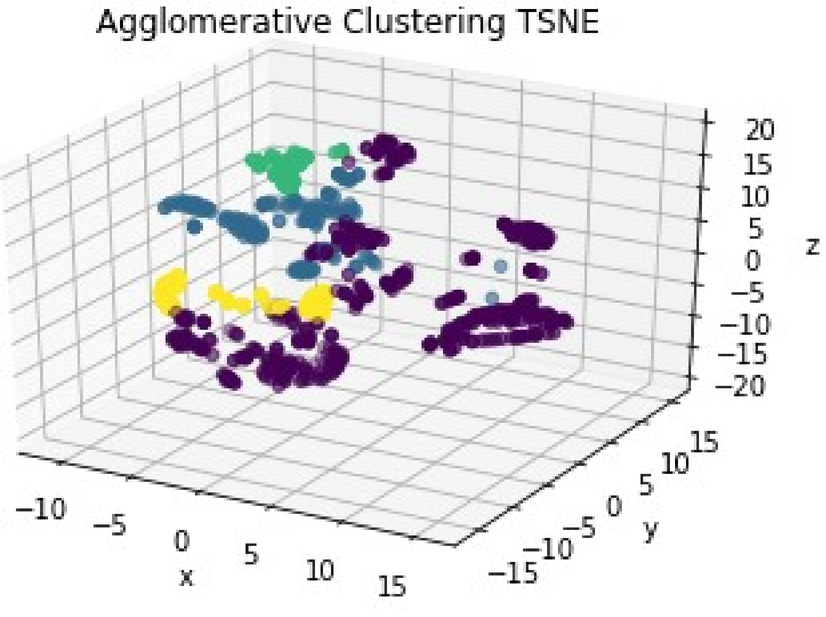

K-means Clustering PCA
4
3
2
1
0
−1
−2
z
3
2
1
0
−1
−2
y
−2
−1
0
1
2
3
4
x


K-means Clustering TSNE
20
15
10
5
0
−5
−10
−15
−20
z
15
10
5
0
−5
−10
−15
y
−10
−5
0
5
10
15
x


DBSCAN Clustering PCA
−2
−1
0
1
2
3
4
x
−2
−1
0
1
2
3
y
−2
−1
0
1
2
3
4
z


DBSCAN Clustering TSNE
−10
−5
0
5
10
15
x
−15
−10
−5
0
5
10
15
y
−20
−15
−10
−5
0
5
10
15
20
z





Needed to run my code:
https://pypi.org/project/imbalanced-learn/
https://pypi.org/project/gender-guesser/