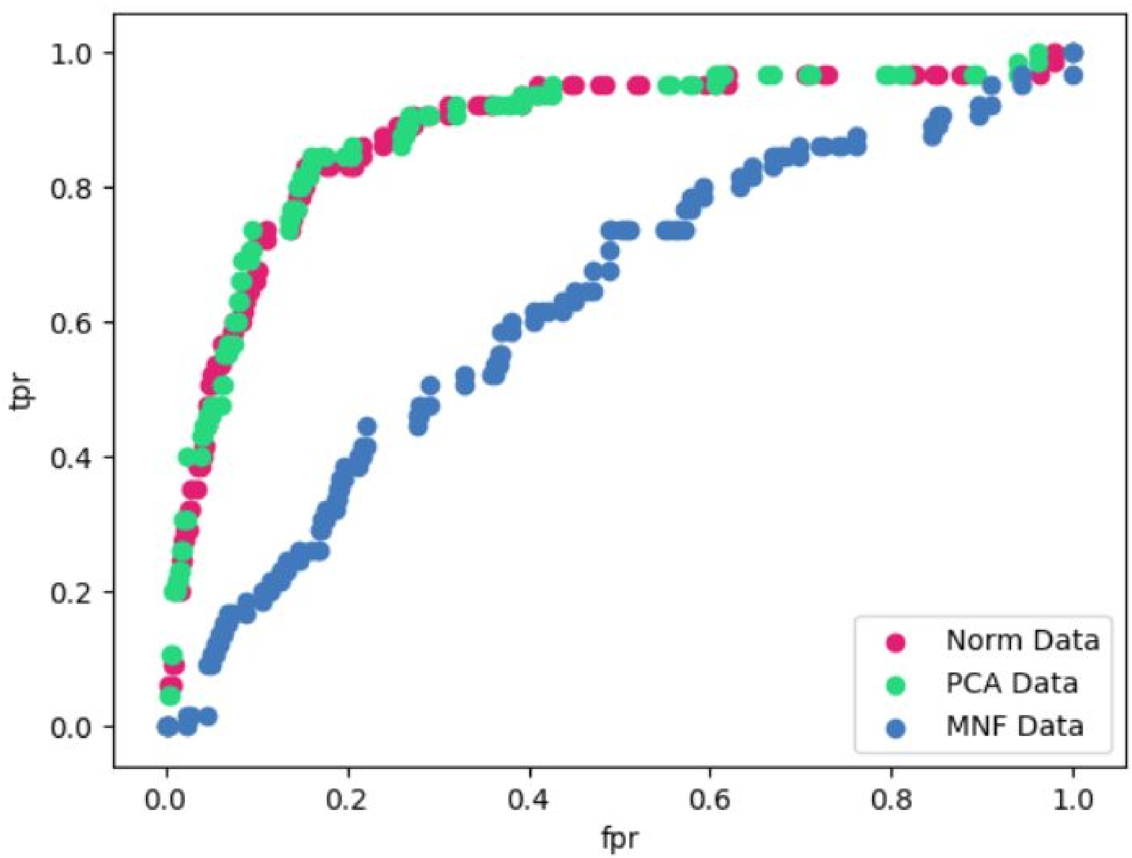
1A: See Code
1B: See Code and the way I figured the most important features for the first 3 components is when I ran PCA component analysis on the dataset averaged the weights of the first 3 principles components and saw which features had the most positive weights.
1C See code After I ran NMF on the dataset I calculated the reconstruction error for a range of components and then I realized that the lower number of components is better for the model. So look through the range ann saw that 9 components was a good value to use

1D



2A: See Code
2B

size	maxFeat	trainAcc	testAcc
1	1	0.948168	0.88125
1	2	0.948168	0.885417
1	3	0.930295	0.88125

1	4	0.918677	0.88125
1	5	0.912422	0.883333
1	6	0.899017	0.877083
1	7	0.893655	0.872917
1	8	0.887399	0.86875
1	9	0.888293	0.870833
1	10	0.879357	0.872917
1	11	0.883825	0.872917
1	12	0.874888	0.866667
1	13	0.872207	0.866667
1	14	0.872207	0.864583
1	15	0.873101	0.866667
1	16	0.873101	0.866667
1	17	0.87042	0.864583
1	18	0.873995	0.866667
1	19	0.875782	0.870833
2	1	0.950849	0.883333
2	2	0.949062	0.89375
2	3	0.930295	0.90625
2	4	0.921358	0.877083
2	5	0.899911	0.877083
2	6	0.895442	0.877083
2	7	0.886506	0.870833
2	8	0.895442	0.86875
2	9	0.891868	0.86875
2	10	0.882931	0.875
2	11	0.88025	0.866667
2	12	0.883825	0.866667
2	13	0.875782	0.870833
2	14	0.876676	0.870833
2	15	0.879357	0.86875
2	16	0.87042	0.866667
2	17	0.871314	0.866667
2	18	0.873101	0.864583
2	19	0.866845	0.864583
3	1	0.946381	0.875
3	2	0.951743	0.897917
3	3	0.941912	0.89375
3	4	0.925827	0.879167
3	5	0.924039	0.883333
3	6	0.905273	0.885417
3	7	0.901698	0.870833
3	8	0.891868	0.875

3	9	0.884718	0.875
3	10	0.891868	0.86875
3	11	0.889187	0.875
3	12	0.873995	0.866667
3	13	0.879357	0.872917
3	14	0.879357	0.866667
3	15	0.883825	0.86875
3	16	0.873995	0.864583
3	17	0.869526	0.866667
3	18	0.875782	0.86875
3	19	0.872207	0.864583
4	1	0.947274	0.858333
4	2	0.949955	0.8875
4	3	0.940125	0.904167
4	4	0.924039	0.885417
4	5	0.90706	0.885417
4	6	0.908847	0.88125
4	7	0.906166	0.877083
4	8	0.891868	0.875
4	9	0.882931	0.872917
4	10	0.89008	0.86875
4	11	0.888293	0.877083
4	12	0.893655	0.875
4	13	0.884718	0.872917
4	14	0.873101	0.86875
4	15	0.882931	0.872917
4	16	0.875782	0.86875
4	17	0.879357	0.870833
4	18	0.874888	0.866667
4	19	0.873995	0.866667
5	1	0.93387	0.86875
5	2	0.956211	0.904167
5	3	0.931189	0.88125
5	4	0.919571	0.879167
5	5	0.922252	0.8875
5	6	0.915103	0.877083
5	7	0.905273	0.875
5	8	0.895442	0.866667
5	9	0.901698	0.879167
5	10	0.893655	0.875
5	11	0.893655	0.877083
5	12	0.885612	0.877083
5	13	0.888293	0.870833

5	14	0.874888	0.866667
5	15	0.879357	0.870833
5	16	0.876676	0.86875
5	17	0.873101	0.864583
5	18	0.875782	0.866667
5	19	0.869526	0.8625
6	1	0.942806	0.864583
6	2	0.946381	0.8875
6	3	0.947274	0.897917
6	4	0.915103	0.8875
6	5	0.915996	0.883333
6	6	0.909741	0.86875
6	7	0.899017	0.875
6	8	0.899017	0.872917
6	9	0.899911	0.877083
6	10	0.889187	0.866667
6	11	0.890974	0.870833
6	12	0.881144	0.86875
6	13	0.882931	0.866667
6	14	0.890974	0.870833
6	15	0.884718	0.875
6	16	0.881144	0.870833
6	17	0.876676	0.866667
6	18	0.873995	0.866667
6	19	0.878463	0.870833
7	1	0.942806	0.860417
7	2	0.928508	0.891667
7	3	0.934763	0.875
7	4	0.932082	0.885417
7	5	0.917784	0.885417
7	6	0.911528	0.8875
7	7	0.904379	0.879167
7	8	0.901698	0.875
7	9	0.898123	0.877083
7	10	0.890974	0.870833
7	11	0.889187	0.875
7	12	0.89008	0.86875
7	13	0.88025	0.879167
7	14	0.890974	0.870833
7	15	0.884718	0.875
7	16	0.876676	0.866667
7	17	0.875782	0.866667
7	18	0.876676	0.872917

7	19	0.878463	0.86875
8	1	0.944593	0.864583
8	2	0.955317	0.897917
8	3	0.941912	0.877083
8	4	0.91689	0.88125
8	5	0.917784	0.88125
8	6	0.906166	0.879167
8	7	0.903485	0.879167
8	8	0.902592	0.872917
8	9	0.896336	0.883333
8	10	0.902592	0.879167
8	11	0.887399	0.875
8	12	0.885612	0.870833
8	13	0.883825	0.870833
8	14	0.882931	0.875
8	15	0.874888	0.866667
8	16	0.874888	0.86875
8	17	0.871314	0.86875
8	18	0.879357	0.86875
8	19	0.873101	0.86875
9	1	0.957998	0.854167
9	2	0.948168	0.864583
9	3	0.937444	0.89375
9	4	0.915996	0.879167
9	5	0.921358	0.875
9	6	0.914209	0.875
9	7	0.899017	0.870833
9	8	0.896336	0.870833
9	9	0.892761	0.875
9	10	0.884718	0.875
9	11	0.883825	0.870833
9	12	0.88025	0.870833
9	13	0.88025	0.86875
9	14	0.883825	0.872917
9	15	0.878463	0.866667
9	16	0.887399	0.86875
9	17	0.882038	0.866667
9	18	0.879357	0.86875
10	1	0.9437	0.866667
10	2	0.954424	0.9
10	3	0.932976	0.883333
10	4	0.922252	0.872917
10	5	0.923146	0.895833

10	6	0.908847	0.875
10	7	0.912422	0.88125
10	8	0.899017	0.86875
10	9	0.894549	0.872917
10	10	0.885612	0.86875
10	11	0.881144	0.870833
10	12	0.882931	0.866667
10	13	0.888293	0.875
10	14	0.886506	0.86875
10	15	0.882038	0.86875
10	16	0.874888	0.86875
10	17	0.881144	0.86875
10	18	0.876676	0.86875
10	19	0.873995	0.870833

2B-2C: The table above is the result of me running a program to find the optimal parameters I noticed that if the depth level is none, it produces better results compared to any other depth level. When I ran my optimal parameters the oob error was about 13 and the accuracy was about 91.