The identifier that I used on miner2 was Zinjero. At the time of writing this report my highest rank is 39 with an accuracy score of 0.86.

Part 1:

After loading in both the test and data sets into separate data frames, I started doing some data preprocessing on them. The first thing I did was normalize certain features. These features that I normalized were age, capital-gain, capital-loss, and hours-per-week. The reason I had to do this is because these features had a vast range of values, especially capital-gain and capital-loss, that would be impractical to represent in a categorical manner. After doing this, I separated all the categorical features on whether they were nominal or ordinal. I concluded that the nominal features were workclass, marital-status, occupation, relationship, race, sex, and native-country. Also, I concluded that the ordinal features in the dataset were age, education, education-num, capital-gain, capital-loss, and hours-per-week. I then used OneHotEncoder to encode the nominal features and OrdinalEncoder to encode the ordinal features. Next, I split the preprocessed train data into a train set and a validation set.

After finishing preprocessing, I decided that I wanted to test out several classifiers on this dataset. These classifiers were KNN, Random Forest, Decision Tree, Linear SVM, Logistic Regression, and Ada Boost. To find out what the optimal hyperparameter for each of these classifiers would be, I did cross validation for each classifier. Specifically, for each classifier and its corresponding hyperparameter, I calculated the accuracy and f1 score of the classifier as the value of the hyperparameter increased. The results for each classifier can be seen in the tables below.

KNN results:

Number of neighbors	accuracy-score	f1-score
2	0.8257331490864425	0.5331139448786507
3	0.8163672654690619	0.6047587574355584
4	0.8320282511899278	0.5779320987654321
5	0.8307999385843697	0.6259334691106585
6	0.8400122831260556	0.6097378277153559
7	0.8327959465684016	0.6251290877796901
8	0.8380162751420236	0.6122748989342154
9	0.836480884385076	0.6326319420489824
10	0.8400122831260556	0.614929785661493
11	0.8354061108552127	0.6280360860513532
12	0.8433901427913404	0.6244477172312224
13	0.8387839705204975	0.6326102169349196
14	0.8381698142177184	0.6127847171197648
15	0.8384768923691079	0.6349757113115891
16	0.8395516658989712	0.6212395795578108
17	0.8381698142177184	0.6345353675450762
18	0.8383233532934131	0.6227158724471514
19	0.839705204974666	0.6392536281962681
20	0.8409335175802242	0.6276060388209921
21	0.8409335175802242	0.6420179682100898
22	0.8432366037156457	0.632337054375225
23	0.8415476738830032	0.6426592797783933
24	0.844311377245509	0.6357758620689654
25	0.8432366037156457	0.6458550121401319
26	0.8420082911100876	0.6310505557547508
27	0.8403193612774451	0.6386379430159833
28	0.8409335175802242	0.6299999999999999
29	0.8421618301857823	0.6418118466898954
30	0.8424689083371718	0.6319942611190819

Random Forest Classifier Results:

Value of Max Depth	accuracy-score	f1-score
2	0.7802855826807923	0.07498383968972204
3	0.8082296944572394	0.32740980075390413
4	0.8281897742975587	0.4797768479776848
5	0.8354061108552127	0.513611615245009
6	0.8395516658989712	0.5625784847216408
7	0.8435436818670352	0.5777041027766265
8	0.8492246276677414	0.6062550120288693
9	0.8490710885920467	0.6137524557956778
10	0.8524489482573315	0.6293868106440416
11	0.8533701827115001	0.6308465403942791
12	0.8562874251497006	0.6430205949656751
13	0.8558268079226163	0.6439135381114904
14	0.8592046675879011	0.652256351915055
15	0.8558268079226163	0.6473901614720241
16	0.8562874251497006	0.6512667660208644
17	0.8576692768309535	0.6547486033519553
18	0.8610471364962383	0.666912035333088
19	0.857976354982343	0.6603011384502387
20	0.8615077537233226	0.6715222141296432
21	0.8588975894365116	0.6654532216963961
22	0.8581298940580377	0.664488017429194
23	0.8607400583448488	0.6733885487936622
24	0.8581298940580377	0.6685796269727404
25	0.8565945033010901	0.6642703091301222
26	0.8550591125441425	0.6614060258249641
27	0.8541378780899739	0.6619217081850534
28	0.8533701827115001	0.664323374340949
29	0.8498387839705205	0.661122661122661
30	0.8522954091816367	0.6655076495132128

Decision Tree results: 0.85 Accuracy on miner

Value of Max Depth	f1-score	accuracy-score
2	0.5115692048801009	0.8217411331183786
3	0.542997542997543	0.828650391524643
4	0.5908396946564886	0.8354061108552127
5	0.6117000392618767	0.8481498541378781
6	0.6232741617357003	0.8533701827115001
7	0.6208236208236207	0.85014586212191
8	0.624367458154924	0.8518347919545525
9	0.618096357226792	0.8502994011976048
10	0.6399394856278366	0.8538307999385844
11	0.6478034251675354	0.8547520343927529
12	0.6527929901423877	0.853984339014279 2
13	0.658273381294964	0.8541378780899739
14	0.6490210297316896	0.8513741747274681
15	0.6522662889518415	0.8492246276677414
16	0.6655780535597648	0.8427759864885613
17	0.6623207301173403	0.8409335175802242
18	0.6455134766291368	0.8404729003531399
19	0.6471176274575142	0.837402118839244 5
20	0.6452905811623246	0.8369415016121603
21	0.6347505858721125	0.8324888684170121
22	0.6329715061058345	0.833870720098265
23	0.6376518218623481	0.8350990327038231
24	0.6334114496149984	0.831874712114233
25	0.6352074966532798	0.8326424074927069
26	0.6300908784920901	0.831260555811454
27	0.6304857621440536	0.8306463995086749
28	0.6282440175261207	0.8306463995086749
29	0.6263921700978737	0.8300322432058959
30	0.6313672922252012	0.8311070167357593

Linear SVM Results: 0.80 Accuracy on miner

Value of C in SVM	f1-score	accuracy-score
1e-05	0.5270997103847745	0.6490096729617688
0.0001	0.5885817852288174	0.721173038538308
0.001	0.6242409521496235	0.7624750499001997
0.01	0.6425682507583417	0.7828957469676032
0.1	0.6415478615071284	0.7838169814217718
1.0	0.6404058490002984	0.814985413787809
10.0	0.536704730831974	0.8255796100107478
100.0	0.6255259467040672	0.7950253339474896
1000.0	0.032786885245901634	0.7735298633502227

Max f1-score when C = 0.01

Max accuracy-score when C = 10.0

Linear Regression Results: 0.84 Accuracy on miner

Value of C in LR	f1-score	accuracy-score
1e-05	0.5664710385302373	0.7391371103945954
0.0001	0.6045418950665623	0.7673883003224321
0.001	0.6272310045894951	0.7755258713342545
0.01	0.6362020817466362	0.7799785045294028
0.1	0.642362002567394	0.7861200675571933
1.0	0.6430223592906709	0.7867342238599724
10.0	0.6429675425038639	0.7871948410870566
100.0	0.6420702024084038	0.7855059112544143
1000.0	0.6443987667009251	0.7875019192384461

Max f1-score when C = 1000.0

Max accuracy-score when C = 1000.0

AdaBoost Results: 0.85 Accuracy on miner

Value of n_estimators in AdaBoost	accuracy-score	f1-score
1	0.7716873944418855	0.0
2	0.8217411331183786	0.5115692048801009
3	0.8294180869031168	0.5802795617680393
4	0.8307999385843697	0.5809885931558935
5	0.8311070167357593	0.5778971603990791
6	0.8369415016121603	0.6052044609665428
7	0.8413941348073085	0.6016197454685692
8	0.8423153692614771	0.6014745828482732
9	0.8432366037156457	0.603495145631068
10	0.841701212958698	0.6107965269913174
11	0.8458467680024566	0.6135488837567359
12	0.8447719944725933	0.6322299017824663
13	0.8466609243052357	0.6458923512747876
14	0.8478427759864886	0.642109064644276
15	0.8490710885920467	0.6460208858480374
16	0.8470750860080147	0.6356986100950986
17	0.8499923230462153	0.6433004746257759
18	0.85014586212191	0.6521739130434783
19	0.8512206356517734	0.6538049303322615
20	0.8510670965760786	0.6525787965616047
21	0.849531705819131	0.6462093862815884
22	0.8489175495163519	0.6395604395604396
23	0.8502994011976048	0.6437705516989404
24	0.8522954091816367	0.6476190476190476
25	0.8526024873330262	0.6496350364963503
26	0.8519883310302472	0.6486880466472303
27	0.8535237217871948	0.6484893146647015
28	0.8529095654844158	0.6498538011695906
29	0.8535237217871948	0.650805270863836

Using the values of f1 and accuracy in the each of the tables above, I was able to conclude that:

- the best number of neighbors value for the KNN classifier is 30 because it had f1 score of 0.632 and an accuracy score of 0.843
- the best max depth value for the random forest classifier is 20 because it had f1 score of 0.672 and an accuracy score of 0.862
- the best max depth value for the decision tree classifier is 11 because it had f1 score of 0.648 and an accuracy score of 0.855
- the best C value for the Linear SVM classifier is 10 because it had f1 score of 0.537 and an accuracy score of 0.826
- the best C value for the Logistic Regression was 1000 because it had f1 score of 0.644 and an accuracy score of 0.788
- and the best n_estimators in the Ada Boost Classifier is 27 because it had f1 score of 0.648 and an accuracy score of 0.854

Accuracy of each classifier when tested on miner:

- KNN: 0.84
- Random Forest: 0.86
- Decision Tree: 0.85
- Linear SVM: 0.80
- Logistic Regression: 0.84
- Ada Boost: 0.85

Part 2:

To compare how biased each of these classifiers were for race and sex, I first calculated the average demographic disparity for race and sex, the average equality of opportunity disparity for race and sex, and the average equality of odds for race and sex for each of the classifiers. The results of these fairness metrics for each classifier can be seen in the tables below.

Fairness metrics for Random Forest classifier:

```
Average demographic disparity for race: 0.04004828435518567
Average demographic disparity for sex: 0.0031701479332217897
Average equality of opportunity disparity for race: 0.016032699189609458
Average equality of opportunity disparity for sex: 0.0030357741518720616
Average equality of odds disparity for race: 0.04378533866482481
Average equality of odds disparity for sex: 0.009356497956192644
```

Fairness metrics for KNN classifier:

```
Average demographic disparity for race: 0.030277986157024102
Average demographic disparity for sex: 0.0028387908169568155
Average equality of opportunity disparity for race: 0.018020611463527042
Average equality of opportunity disparity for sex: 0.0010528927613044081
Average equality of odds disparity for race: 0.0424654593938653
Average equality of odds disparity for sex: 0.0073133219871494015
```

Fairness metrics for Decision Tree classifier:

```
Average demographic disparity for race: 0.02605589902179357
Average demographic disparity for sex: 0.0008588557583216749
Average equality of opportunity disparity for race: 0.016431988877648268
Average equality of opportunity disparity for sex: 0.003776460954472635
Average equality of odds disparity for race: 0.06086703465527814
Average equality of odds disparity for sex: 0.0038461206594480046
```

Fairness metrics for Linear SVM classifier:

```
Average demographic disparity for race: 0.010592522394458358
Average demographic disparity for sex: 0.005404730251736711
Average equality of opportunity disparity for race: 0.009144548150891763
Average equality of opportunity disparity for sex: 0.00348793087736924
Average equality of odds disparity for race: 0.03382537752905505
Average equality of odds disparity for sex: 0.0053788446211869845
```

Fairness metrics for Logistic Regression classifier:

```
Average demographic disparity for race: 0.03678521410928922
Average demographic disparity for sex: 0.011425033425083525
Average equality of opportunity disparity for race: 0.015425605463908166
Average equality of opportunity disparity for sex: 0.00126239800693273
Average equality of odds disparity for race: 0.05926669544953276
Average equality of odds disparity for sex: 0.0016919311122339797
```

Fairness metrics for Ada Boost classifier:

```
Average demographic disparity for race: 0.026876677459930325
Average demographic disparity for sex: 0.008373370126004115
Average equality of opportunity disparity for race: 0.01575093369296567
Average equality of opportunity disparity for sex: 0.000550017036112127
Average equality of odds disparity for race: 0.05294021354185936
Average equality of odds disparity for sex: 0.0049845622729352
```

Using the tables above, I was able to conclude that:

- Linear SVM had the lowest average demographic disparity for race
- Decision Tree had the lowest average demographic disparity for sex
- Linear SVM had the lowest average equality of opportunity disparity for race
- Ada Boost had the lowest average equality of opportunity disparity for sex
- Linear SVM had the lowest average equality of odds disparity for race
- Logistic Regression had the lowest average equality of odds disparity for sex

Using these conclusions, I am persuaded to believe that Linear SVM was the most unbiased classifier for race and sex out of the other 5 classifiers.

The most likely cause of the biases seen in each of the classifiers is because of the inclusion of race and sex. Also, the inclusion of features that are related to race and sex could have caused the biases seen in the classifiers.

Part 3:

The classifier that I will be referring to from here onwards will be the Random Forest classifier shown above because it the highest accuracy.

To find out what features where correlated with race and sex, I created a correlation heatmap.

Correlation heatmap:

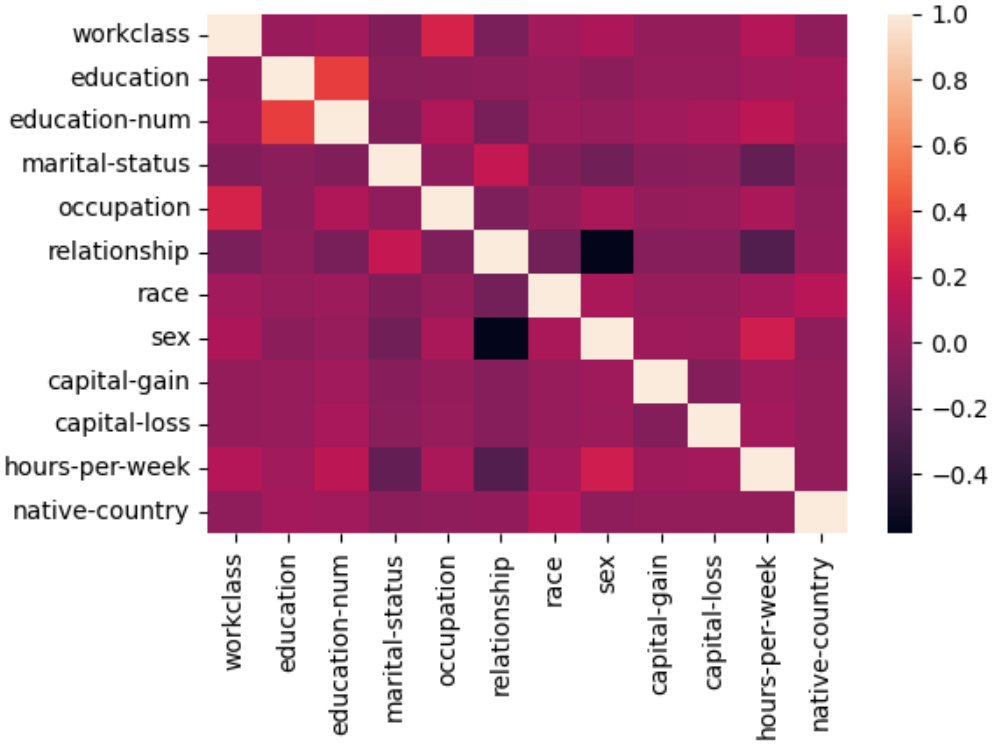

Table that shows how race and sex correlate with the other features:

	race	sex
workclass	0.049742	0.095981
education	0.014131	-0.027356
education-num	0.031838	0.012280
marital-status	-0.068013	-0.129314
occupation	0.006763	0.080296
relationship	-0.116055	-0.582454
race	1.000000	0.087204
sex	0.087204	1.000000
capital-gain	0.017123	0.039856
capital-loss	0.016933	0.034283
hours-per-week	0.054760	0.226817
native-country	0.137852	-0.008119

For the purposes of this assignment, two features are significantly correlated if their values in the table above is greater than 0.10. Using the correlation heatmap and the table above, I was able to find out that the native-country was significantly correlated with race. Also, I was able to find out that workclass, occupation, and hours-per-week were significantly correlated with sex.

After doing this I tested the classifier's fairness and accuracy when no feature was removed, when only race and sex were removed, and when the features significantly correlated with sex and race were removed in addition. The results of these tests can be seen below. Note: these tests were only done on the validation set.

<u>When no feature is removed:</u> Accuracy score on miner = 0.86

```
Accuracy Score: 0.8615077537233226
F1 Score: 0.6715222141296432

Fairness metrics for forest classifier
Average demographic disparity for race: 0.04004828435518567
Average demographic disparity for sex: 0.0031701479332217897
Average equality of opportunity disparity for race: 0.016032699189609458
Average equality of opportunity disparity for sex: 0.0030357741518720616
Average equality of odds disparity for race: 0.04378533866482481
Average equality of odds disparity for sex: 0.009356497956192644
```

<u>When only sex and race are removed:</u> Accuracy score on miner = 0.86

```
Accuracy Score: 0.8593582066635959
F1 Score: 0.665204678362573

Fairness metrics for forest classifier
Average demographic disparity for race: 0.040361874438751374
Average demographic disparity for sex: 0.0005228686918770376
Average equality of opportunity disparity for race: 0.014722656155319947
Average equality of opportunity disparity for sex: 0.0024923863626080123
Average equality of odds disparity for race: 0.05183447591878911
Average equality of odds disparity for sex: 0.01671980236005388
```

<u>When the features correlated with sex and race are also removed:</u> Accuracy score on miner = 0.85

```
Accuracy Score: 0.8526024873330262
F1 Score: 0.6299151888974557

Fairness metrics for forest classifier
Average demographic disparity for race: 0.00910834230023258
Average demographic disparity for sex: 0.004193472149060823
Average equality of opportunity disparity for race: 0.025671713433546593
Average equality of opportunity disparity for sex: 0.00743538431006692
Average equality of odds disparity for race: 0.05657449577997642
Average equality of odds disparity for sex: 0.0015806018556420165
```

Simply removing just race and sex from the data did not make much of a difference on the fairness of the classifier. It did however reduce the average demographic disparity for sex. It also resulted in a slight decrease in accuracy of the classifier.

Removing the features that are correlated with race and sex in addition also did not make much of a difference on the fairness of the classifier. It did however reduce the average demographic disparity for race. It also resulted in an even more decrease in accuracy of the classifier.

Using this data alone, I can't comfortably say that these mitigation strategies reduced unfair outcomes. I think that this is the case because my classifier was already fair even without removing any of the sensitive attributes. I think the more likely reason, however, is because I didn't use the test data set when testing the removal of sensitive attributes.