Introduction into Artificial Intelligence

Comp 3308

Assignment 2

470366820

470168442

Table of Content

[**Introduction**](#_clpi1yo959im)4

[Aim](#_8a2kes9r4qep) 4

[Importance](#_ue0x5htqlcru) 4

[**Data**](#_mgr8qilrs5dd)5

[Description of Dataset:](#_3qi7urh061am) 5

[CFS(Correlation Feature Selection)](#_y4gilsgjr45i) 6

[**Results and Discussion**](#_g53huas4twfg)7

[Accuracy of Classifiers](#_83rly5bzheqd) 7

[Analysis of Classifiers](#_o127nvdx8ehd) 7

[ZeroR:](#_u4wvaw24s94o) 7

[1R:](#_732qjnaz1pe7) 8

[NB (Naive Bayes):](#_oho8eeb84tlr) 9

[DT (Decision Tree)](#_rhkw2f9kehjx) 10

[MLP (Mult-Layer Perceptron)](#_17lb7gurq0zb) 10

[SVM (Support Vector Machine)](#_p4hvxxgaukkn) 11

[RF](#_xvy9folkaxpj) 12

[Implementation of Classifier](#_2gs0dopqnpd1) 13

[**Conclusion**](#_lijrlimiqhfs)14

[**Future work**](#_fyi95mvjtpzi)14

[**Reflection**](#_6fk5txe13h2r)15

[**References**](#_dgchodt3pat0)16

[**Appendix**](#_qnaazvud5shp)17

[CFS Attributes](#_1aq29ysiasgt) 17

[ZeroR](#_dp4lzj3zcxqi) 18

[ZeroR (CFS)](#_8mbvnltl3z1r) 19

[1R](#_lnbs71cgpqlf) 20

[1R (CFS)](#_l9skbq41dyby) 21

[1NN](#_ngwd14uj3g) 22

[1NN (CFS)](#_5s1jdjgwp5a1) 23

[5NN(Weka)](#_fi5jknbwtb8x) 24

[5NN(CFS)(Weka)](#_obb7zed9tuvv) 25

[NB(Weka)](#_oldceapiceqg) 26

[NB (CFS) (Weka)](#_4cg0rakxg4p4) 27

[DT](#_qjlq4hukixdt) 28

[DT (CFS)](#_qbi6479i61nz) 28

[MLP](#_rtvdket1bmsi) 29

[MLP (CFS)](#_ldiovsr3tu3w) 29

[SVM](#_19ge0qqtvu53) 30

[SVM (CFS)](#_k9hp6i9tcjal) 31

[RF](#_nf6b5gtwp01y) 32

[RF (CFS)](#_w0cpydpetupr) 33

[1NN (Implemented)](#_vn7h7h3bmfxi) 34

[1NN (CFS) (Implemented)](#_yqyisfwgqgq2) 35

[5NN(Implemented)](#_13qbr03xe9c2) 36

[5NN (CFS)(Implemented)](#_qkt9f9hnw4b) 37

[NB (Implemented)](#_9qu8xyrrfmm8) 38

[NB (CFS)(Implemented)](#_t1rl2e7xf0yj) 39

[Analysis of ZeroR](#_yrp4lpdhw5em) 39

[Analysis of 1R](#_ao5h8rm5kuad) 40

[Analysis of NB](#_3pem5erxbhrt) 40

[Analysis of MLP](#_7l5otrvzq3ah) 42

[Analysis of 1NN](#_4ijbqaauutvz) 40

[Analysis of 5NN](#_gsr5ze9bkkdf) 41

[Analysis of DT(J48)](#_yqjwjkir9ysx) 41

[Analysis of RF](#_mwp33ru62n2f) 42

[Analysis of SVM (SMO)](#_7l5otrvzq3ah) 42

# 

# **Introduction**

## Aim

The main aim of this study is to investigate the effects and accuracy of class classification by implementing the K-Nearest Neighbor (KNN) and Naive Bayes (NB) algorithms on the modified Pima Indians Diabetes dataset.

An objective of this assignment was to examine the effects of 10-fold stratified cross validation to estimate the accuracy of not only the K-Nearest Neighbor and Naive Bayes algorithms but various other classifiers such as MLP, SVM and RF on this data set using Weka. The second objective is to explore the correlation of the data using Correlation-based Feature Selection to reduce the dataset to the best subset and how that would affect the accuracy of the various classifiers.

By doing so we are able to understand the limitations of certain classifiers, their complexity and select the best classifier with the highest accuracy for this dataset.

## Importance

Conducting an analysis on this dataset is highly important as AI researchers are able to experiment and understand the effects of changing parameters of the various classifiers and their complexities. It would allow researchers to understand the limitations of the dataset provided and may result in another collection of data samples in the hopes to create a highly accurate classifier. With such a classifier, it would positively impact the medical industry as the human errors when diagnosing patients for diabetes is lowered and may be used in other industries as well.

# **Data**

## **Description of Dataset:**

The pima Indians database was originally collected by the National Institute of Diabetes and Digestive and Kidney Diseases on the 9th of May 1990 and was donated by the following individual:

**Vincent Sigillito**

**(vgs@aplcen.apl.jhu.edu)**

**Research Center, RMI Group Leader**

**Applied Physics Laboratory**

**The Johns Hopkins University**

**Johns Hopkins Road**

**Laurel, MD 20707**

**(301) 953-6231**

The data has been used before by a team of researchers who applied the ADAP learning algorithm, an early neural network model to predict the onset of diabetes mellitus or commonly known as diabetes. The diagnosis of the patient in the dataset is done in accordance with the following criteria: If the 2-hour post load plasma glucose was at least 200mg/dl, they were diagnosed as yes, otherwise no. With 576 training instances, their algorithm resulted in an accuracy of 76% using the remaining 192 instances as testing data.

For the purpose of this assignment the data has been modified as of March 2015 with the following changes:

* Replaced missing values with averages
* Class changed to nominal values
* The following constraints were put in place when selecting the data:
  1. All patients here are females
  2. Age is of at least 21 years old
  3. Having a Pima Indian Heritage

The dataset has 768 instances with the following attributes and class

**Numeric Valued Attribute:**

1. Number of times pregnant
2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3. Diastolic blood pressure (mm Hg)
4. Triceps skin fold thickness (mm)
5. 2-Hour serum insulin (mu U/ml)
6. Body mass index (weight in kg/(height in m)^2)
7. Diabetes pedigree function
8. Age (years)

**Class (Binary classification):**

1. Class variable ("yes" or "no")
   1. **Yes:** The patient is diagnosed with diabetes (Positive)
   2. **No:** The patient is not diagnosed with diabetes (Negative)

Out of the 768 instances, 500 patients have been diagnosed as negative and the remaining 268 patients have been diagnosed as positive.

## **CFS (Correlation Feature Selection)**

A classifiers’ performance is only as good as the set of data that is fed. A dataset such as the one above has numerous attributes, some being relevant in making predictions while the others may just add noise resulting in low accuracy (Hall, M. A. (1999)). In order to select relevant features for the model we have selected CFS which allows us to select features which have low correlation with each other while being highly correlated with the class.

**List of features selected:**

1. Plasma glucose concentration a 2 hours in an oral glucose tolerance test
2. 2-Hour serum insulin (mu U/ml)
3. Body mass index (weight in kg/ (height in m)^2)
4. Diabetes pedigree function
5. Age (years)

We will use these attribute subsets selected by CFS as well as all the features and run various classifiers on them. Below are the accuracy results.

# Results and Discussion

# 

## Accuracy of Classifiers

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Classifier** | **ZeroR** | **1R** | **1NN** | **5NN** | **NB** | **DT** | **MLP** | **SVM** | **RF** |
| **No feature selection (%)** | 65.10 | 70.83 | 67.84 | 74.48 | 75.13 | 71.74 | 75.39 | 76.30 | 74.87 |
| **CFS (%)** | 65.10 | 70.83 | 69.01 | 74.48 | 76.30 | 73.31 | 75.78 | 76.69 | 75.91 |

**Table 1: Accuracy results using Weka (r.d to 2 decimal places)**

**(Refer to Appendix)**

## Analysis of Classifiers

### **ZeroR:**

ZeroR is one of the simplest classifiers which selects the majority class while focusing only on the class and ignoring all features in the dataset. As it does not have much power to predict, its result is a good baseline to benchmark other classification methods against (Sayad, D. S. (n.d.)). With the majority class being “no” with 500 instances out of the 768 instances we attained an accuracy of 65.10% regardless of whether features were selected. It is a useful classifier as we can compare the performance of other classifiers with this benchmark and expect their accuracy to be greater.

However, when analyzing it again, with reference to the appendix section Analysis of ZeroR, using ZeroR as a base classifier, assuming distribution of results are from a Gaussian distribution, we applied a Paired T-Test (corrected) with 95% confidence interval to the other algorithms. From this analysis we observed that ZeroR has the lowest percentage with 65.11%, all the other algorithms have a larger percentage than it and the difference is significant as shown with the “v” symbol beside each algorithm result. This justifies the idea to use ZeroR as a base classifier due to its lowest accuracy.

### 

### **1R:**

Unlike zeroR, 1R selects at most 1 feature in making predictions. The way it works is by assigning a rule for each feature, selects the rule with the lowest total error to be the one rule (Sayad, D. S. (n.d.) ). Despite this classifier being simple and computationally cheap, it is able to provide a great accuracy of 70.83% for both CFS and when no feature selection was done. This would mean that the feature that was selected by 1R happens to be also in the subset of features selected by CFS. The accuracy of this classifier is 5.73% more than the zeroR classifier. This can mainly be attributed to the selection of Plasma glucose concentration of 2 hours in an oral glucose tolerance test attribute which has the highest correlation to the class value.

However, when analyzing it again, with reference to the appendix section Analysis of 1R, using 1R as a base classifier, assuming distribution of results are from a Gaussian distribution, we applied a Paired T-Test (corrected) with 95% confidence interval to the other algorithms. From this analysis we observed ZeroR’s percentage value was significantly different from 1R with 65.11% as compared to 70.95%. This shows 1R is a better model for this dataset as compared to ZeroR. Naive Bayes, Multi Layered Perceptron, Random Forest and Support Vector Machine had results larger than the base algorithm (1R) and their results are statistically different. They have result values of 74.81%, 75.02%, 76.08%, 76.28% respectively. The Decision Tree algorithm performs better than 1R and its results are not significantly different as compared to 1R with a higher accuracy probability of 74.77% and therefore a better model as compared to 1R with respect to this dataset.

**KNN (K- nearest Neighbour):**

KNN is also known as an instance based learning or lazy classifier. Unlike 1R, naive Bayes or SVM, lazy classifiers store training examples and do not build the classifier until new unlabeled examples need to be classified. As such, they are fast for training but are slower in the classification phase, as each new example is compared based on the Euclidean distance with each training example (Koprinska, I. (2020, April)).

In this case as k =1, the class of a single nearest training example is assigned to the new example. This gave us an accuracy of 67.84% when all the features were used and 69.01% when CFS was applied. This is about only 2.74% more likely to predict when all features were used and 3.91% more likely to predict when CFS was applied when compared with ZeroR. . With such a low which suggests that we should change the value of k.

When k = 5, the majority class of 5 nearest neighbours was assigned to the new example. With this change in parameter, the accuracy increased to 74.48% for both situations when all the features were applied and when a subset of features selected by CFS was used. This was a 6% increase in accuracy when compared with 1NN and a 9.35% increase in accuracy when compared with ZeroR.

Although k = 5 gave us a higher accuracy, increasing k does not guarantee that accuracy would increase. It is recommended to set k equal to the square root of the number of training examples however, that remains to be tested (Koprinska, I. (2020, April)). It is also important to note that this classifier is memory and computationally intensive and becomes impractical to use for large datasets. Another issue with this form of classifier is that high dimensionalities (many features) causes most of the training examples to lie on the boundaries causing the accuracy to reduce. To combat this, we have implemented CFS which reduces the dimensionality and in turn increases the accuracy of the classifier as seen in table1.

### **NB (Naive Bayes):**

NB algorithm is also known as a classification technique based on Bayes’s Theorem with the assumption of independence among predictors. It is quick and easy to predict the class of a test data set and generally performs well with multi class prediction (Koprinska, I. (2020, April)). The NB algorithm generally performs better when the assumption of independence holds between the data. It generally performs better with categorical input/ test variables compared to numerical. Due to the need to assume the independence of predictors, it generally does not fare well in real life scenarios. With respect to this dataset, it generally fairs pretty well with an accuracy percentage of 75.13% with all features present and 76.30%. with CFS. This would mean that with CFS, the noise in the data was reduced and therefore it shows an increase in an 1.557 % increase with CFS. The accuracy of this classifier is 10.03% more than the zeroR classifier with full features.

However, when analyzing it again, with reference to the appendix section Analysis of NB, using NB as a base classifier, assuming distribution of results are from a Gaussian distribution, we applied a Paired T-Test (corrected) with 95% confidence interval to the other algorithms. From this analysis we observed that ZeroR’s, 1R’s, IBk’s percentage accuracy were significantly different and lower than NB with 65.11%, 70.95%, 69.33% respectively. This shows NB is a better model for this dataset as compared to ZeroR’s, 1R, IBk. The Multi Layered Perceptron, Random Forest, Decision Trees and Support Vector Machine algorithms had results close to the base algorithm (NB) and their results were not statistically different. They have result values of 75.02%, 74.77%, 76.08%, 76.28% respectively. The Multi Layered Perceptron, Decision Trees and Support Vector Machine algorithms performed better than NB with results not significantly different as compared to NB and with higher accuracy probability and therefore better models as compared to NB with respect to this dataset.

### 

### **DT (Decision Tree)**

A decision tree is a supervised learning algorithm with a tree like structure. The features of the dataset are represented by internal nodes with branches which represent decision rules. The leaves of the tree are the outcomes of the classifier (Koprinska, I. (2020, April)). Using this classifier, we were able to attain an accuracy of 71.74% by using all the features and 73.31% when applying CFS. Although it is much more accurate than ZeroR, it is not as accurate as the Naive Bayes classifier. This classifier was able to predict with a 6.64% higher accuracy as compared to ZeroR when using all the features and 8.21% higher accuracy as compared to ZeroR when CFS was applied.

The reason for the low accuracy could be attributed to how a DT algorithm works. If the data is fairly large, DT is prone to overfitting. This results in the tree resembling a look-up table rather than a tree that has extracted patterns to make useful predictions. Here with 768 instances, the data size is relatively medium sized which could account for some overfitting. Another reason for overfitting is due to noise which could have been added by mistake during the collection of the data. One way to avoid this is by pruning the tree which would in turn increase the accuracy of this classifier.

However, when analyzing it again, with reference to the appendix section Analysis of DT (J48), using DT (J48) as a base classifier, assuming distribution of results are from a Gaussian distribution, we applied a Paired T-Test (corrected) with 95% confidence interval to the other algorithms. From this analysis we observed that ZeroR’s, IBk’s percentage values were significantly different and lower than DT (J48) 74.77%, with 65.11%, 69.33% respectively. This shows DT (J48) is a better model for this dataset as compared to ZeroR’s and IBk. The 1R, NB, Multi Layered Perceptron, Random Forest and Support Vector Machine algorithms had results that were not statistically different to the base algorithm DT (J48). They have result values of 70.95%, 74.81%, 75.02%, 76.08% and 76.28% respectively. The NB, Multi Layered Perceptron, Random Forest and Support Vector Machine algorithms performed better than DT (J48) with results not significantly different as compared to DT (J48) and with higher accuracy probability and therefore better models as compared to DT (J48) with respect to this dataset.

### 

### **MLP (Mult-Layer Perceptron)**

This form of classifier widely used today is essentially a neural network with multiple layers with neurons which have weights and an activation function. By tweaking the number of neurons and layers, the activation function, learning rate, and other parameters, we are able to create a highly complex decision boundary that can classify the new examples with much more accuracy (Koprinska, I. (2020, April)). Using the MLP we were able to attain an accuracy of 75.39% using all the features in the data and 75.58% with CFS applied. Although this is more accurate than ZeroR, the accuracy is not impressive. It is only 10.29% increase in accuracy when all features are used when compared with zeroR and a 10.68% higher accuracy as compared to zeroR when CFS was applied.

There are few reasons as to why the accuracy was low. Firstly, the classifier was stuck in a local minimum and is unable to escape it to reach the global minimum. It is highly possible that in both cases when CFS applied or when all the features were used, the classifier was stuck in a local minimum and thus giving a low accuracy. To combat this issue the initialization point could have been adjusted by changing the initial weights. Another possible solution would be to change the learning rate. It is important to take note that through experimentation and adjusting the parameters would result in a higher accuracy.

However, when analyzing it again, with reference to the appendix section Analysis of MLP, using MLP as a base classifier, assuming distribution of results are from a Gaussian distribution, we applied a Paired T-Test (corrected) with 95% confidence interval to the other algorithms. From this analysis we observed that ZeroR’s, 1R’s, IBk’s percentage values were significantly different and lower than MLP 75.02%, with 65.11%, 70.95%, 69.33% respectively. This shows MLP is a better model for this dataset as compared to ZeroR, 1R and IBk. The NB, Decision Tree, Random Forest and Support Vector Machine algorithms had results that were not statistically different to the base algorithm MLP. They have result values of 74.81%, 74.77%, 76.08% and 76.28% respectively. The Random Forest and Support Vector Machine algorithms performed better than MLP with results not significantly different as compared to MLP and with higher accuracy probability and therefore better models as compared to MLP with respect to this dataset.

### **SVM (Support Vector Machine)**

SVM is another form of supervised machine learning model which is able to create both linear and nonlinear decision boundaries. Each decision boundary is created based on the highest possible distance to the training examples from the two classes (Koprinska, I. (2020, April)). Using this classifier, we were able to attain an accuracy of 76.30% when all features were used and an accuracy of 76.69% with the CFS dataset. This is the best performing classifier thus far with a 11.20% increase in accuracy when compared with ZeroR when all features were used and a 11.59% increase in accuracy when compared with ZeroR when CFS was applied. Here we can see that CFS does not play a significant role in increasing the accuracy by much. This could be due to the fact that the support vectors are present in both the CFS and the data with all the features.

There are many reasons as to why the SVM may be performing poorly. One of the reasons is due to the noise in the dataset. It is highly probable that noise was created when the data was collected. This noise could cause target classes to overlap, making it difficult for the SVM to create a decision boundary and thus reducing the accuracy.

However, when analyzing it again, with reference to the appendix section Analysis of SVM, using SVM as a base classifier, assuming distribution of results are from a Gaussian distribution, we applied a Paired T-Test (corrected) with 95% confidence interval to the other algorithms. From this analysis we observed that ZeroR’s, 1R’s, IBk’s percentage values were significantly different and lower than SVM 76.28%, with 65.11%, 70.95%, 69.33% respectively. This shows SVM is a better model for this dataset as compared to ZeroR, 1R and IBk. The NB, MLP, Decision Tree and Random Forest algorithms had results that were not statistically different to the base algorithm SVM. They have result values of 74.81%, 75.02%, 74.77%, and 76.08% respectively. It has the highest accuracy probability compared to other algorithms and therefore the best model to use with respect to this dataset.

### **RF**

Random Forest is an improvement with respect to the decision trees algorithm. It generates small decision trees from random subsets of the data. Each decision tree is then provided a biased classifier. It requires much more computational power compared to other algorithms due to the need to create many trees (Koprinska, I. (2020, April)). Using this classifier, we were able to attain an accuracy of 74.87% when all features were used and an accuracy of 75.91% with the CFS dataset. This is a good performing classifier with a 9.77% increase in accuracy when compared with ZeroR when all features were used and a 10.81% increase in accuracy when compared with ZeroR with the CFS dataset.

There are many reasons as to why the RF may be performing poorly. One of the reasons is due to the overfitting of the datasets with noisy classification. It is highly probable that noise was created when the data was collected. This noise could cause target classes to overlap, making it difficult for the RF to generate decision trees and make decisions based on the majority of the votes.

However, when analyzing it again, with reference to the appendix section Analysis of RF, using RF as a base classifier, assuming distribution of results are from a Gaussian distribution, we applied a Paired T-Test (corrected) with 95% confidence interval to the other algorithms. From this analysis we observed that ZeroR’s, 1R’s, IBk’s percentage values were significantly different and lower than RF 76.08%, with 65.11%, 70.95%, 69.33% respectively. This shows RF is a better model for this dataset as compared to ZeroR, 1R and IBk. The NB, MLP, Decision Tree and SVM algorithms had results that were not statistically different to the base algorithm RF. They have result values of 74.81%, 75.02%, 74.77%, and 76.28% respectively. The Support Vector Machine algorithm performed better than RF with results not significantly different as compared to RF and with higher accuracy probability and therefore better model as compared to RF with respect to this dataset.

## 

## Implementation of Classifier

|  |  |  |  |
| --- | --- | --- | --- |
| **Classifier** | **1NN** | **5NN** | **NB** |
| **No feature selection (%)** | 68.49 | 74.22 | 74.74 |
| **CFS (%)** | 67.19 | 75.40 | 76.30 |

**Table2: Accuracy result using our implementation (r.d to 2 decimal places)**

**(Refer to Appendix)**

In this assignment we were tasked to implement our own KNN and NB classifiers. After which, we were tasked to check the accuracy of the data using 10-fold stratified cross validation. The results are as follows for 1NN with an accuracy of 68.48% when all features were used and an accuracy 67.19% when the CFS dataset was applied. The difference between the results produced by Weka and our implementation is fairly low with a difference of 0.65% in accuracy for 1NN when all features were used and a difference of 1.82% in accuracy for 1NN when features from CFS were used. With regards to 5NN we got a difference of 0.26% between our implementation of no feature selection and Weka’s 5NN model. As for CFS, our implementation of the 5NN algorithm worked better than that of Weka’s outperforming it by 0.92%. The difference between the results produced by Weka and our implementation is fairly low with a difference of 0.65% for 1NN and 0.26% for 5NN when all features were used. There was a difference of 1.82% for 1NN and a difference of 0.92% for 5NN when features from CFS were used. Finally, with the NB classifier, when using all the features there was a difference of 0.39% in accuracy between Weka and our implementation. The accuracy difference was 0% in accuracy when the subset selected by CFS was applied. This implies that our implementation of the KNN and NB classifier was correct.

## CFS Analysis

Analyzing the results, we can confirm that by applying CFS to the Pima Indian dataset and running the classifiers generally improves the results, except for 1NN. Although there are multiple ways of selecting features, the one based on the correlation proved to improve results. This can be attributed to the fact that correlated features can add noise to the space and thus reduces accuracy (Hall, M. A. (1999)). By eliminating these features and keeping those that are highly correlated with the class would greatly improve the results. For further analysis of other classifiers, it is suggested to use a subset of data which has had CFS applied on.

# Conclusion

It can be concluded that certain classifiers work better with respect to this particular data set. There is a difference of more than 10% when comparing the worst model (ZeroR) and best (SVM). This suggests that the choice of classifiers plays a role in correct prediction and that for this dataset, SVM works well due having a decent margin of separation. However, in real life, a few classifiers are used together as known as an ensemble to better model the data and accurately predict new results.

With regards to this dataset, by having the above mentioned machine learning models being applied to it, there is an average of about 72% accuracy in predicting a new patient being diagnosed with or without diabetes when the numeric valued attributes such as Number of times pregnant, Plasma glucose concentration a 2 hours in an oral glucose tolerance test, Diastolic blood pressure (mm Hg), Triceps skin fold thickness (mm), 2-Hour serum insulin (mu U/ml), Body mass index (weight in kg/(height in m)^2), Diabetes pedigree function, Age (years) be supplied. It must also be noted that because the dataset was of only individuals of Pima Indian heritage, are all females and above the age of 21, the classifiers in the report would only be able to predict for a new example of similar attributes. In order to be able to predict a large diaspora of individuals, it would be a good idea to have more instances of data from people of various races, age groups, geographical locations, gender and blood types.

However, as this report would be used in the medical industry, even the SVM classifier with an accuracy of 76.69% is not advisable to be implemented. This is the same level of accuracy that was attained by a group of researchers using the ADAP algorithm. However, with such a low accuracy, and the possibility of misdiagnosing numerous patients, using any of the classifiers here could cause detrimental effects for patients.

As this analysis does not take into consideration on how the classifiers would behave with changes to their parameters and how the classifiers would behave with larger sets of data, it would be advisable to conduct future work.

# 

# Future work

There could be a future in correctly predicting the diagnosis of a patient with diabetes. With a larger dataset and optimized combinations of machine learning models, the accuracy of correctly predicting the diagnosis of a patient would increase. Fine tuning methods such as treatment of data through the transformation of outliers or ensemble methods such as bagging or boosting (Sunil R., 2015), noise robust/ adaptation models such as noise robust random forest (NRRF) model can be used to minimize noise to increase accuracy/ performance of machine learning models (Zhou., Ding., Li. , 2019).

However, additional attributes/ tests have to be included in the future dataset to properly diagnose an induvial. Attributes such as Hemoglobin A1c (HbA1c) Test, racial/ethnic groups, past illnesses such be taken into account as these would support the future prediction of an accurate diagnosis of diabetes. (American Diabetes Association,2002)(Lipska, K. J., Krumholz, H. M., 2017)

# 

# Reflection

This assignment has had a positive effect on me as I am able to explore various classifiers, apply a real dataset, analyze their results. Some of the important takeaways are that the field of AI is vast, with numerous classifiers, each with different complexities and numerous parameters to experiment with to gain a high level of accuracy. I have also learnt that a classifier is just as good as the data is and that we need to carefully select attributes and ensure that we are not adding noise when collecting it. As such having completed this assignment, we as researchers have realized that there is so much more to explore and experiment in the realm of artificial intelligence.

Our work in this assignment is just one of many papers, created by AI researchers compiling knowledge in the hopes of improving the efficiency and accuracy of machines to be used to assist humans in making sound decisions and predictions.

# References

American Diabetes Association Diabetes Care (2002) Jan; 25(suppl 1): s21-s24.

URL:<https://doi.org/10.2337/diacare.25.2007.S21>

Hall, M. A. (1999). Correlation-based Feature Selection for Machine Learning. Retrieved from <https://www.cs.waikato.ac.nz/~mhall/thesis.pdf>

Koprinska, I. (2020, April). Introduction to Machine learning K-Nearest Neighbors (week 5) PDF.

Koprinska, I. (2020, April). Introduction to Machine learning Naive Bayes (week 6) PDF.

Koprinska, I. (2020, April). Introduction to Machine learning Decision Trees (week 7) PDF.

Koprinska, I. (2020, April). Introduction to Machine learning, Introduction to Neural Networks Perceptrons (week 8) PDF.

Koprinska, I. (2020, April). Introduction to Machine learning, Multilayer Neural Networks and Backpropagation Algorithm (week 9) PDF.

Koprinska, I. (2020, April). Introduction to Machine learning, Introduction to Support Vector Machines(week 10) PDF.

Lipska, K. J., & Krumholz, H. M. (2017). Is Hemoglobin A1c the Right Outcome for Studies of Diabetes?. *JAMA*, *317*(10), 1017–1018.

URL: <https://doi.org/10.1001/jama.2017.0029>

Sayad, D. S. (n.d.). ZeroR. Retrieved from <https://www.saedsayad.com/zeror.htm>

Sayad, D. S. (n.d.). OneR. Retrieved from https://www.saedsayad.com/oner.htm

Sunil R. (2015) 8 Proven Ways for improving the “Accuracy” of a Machine Learning Model

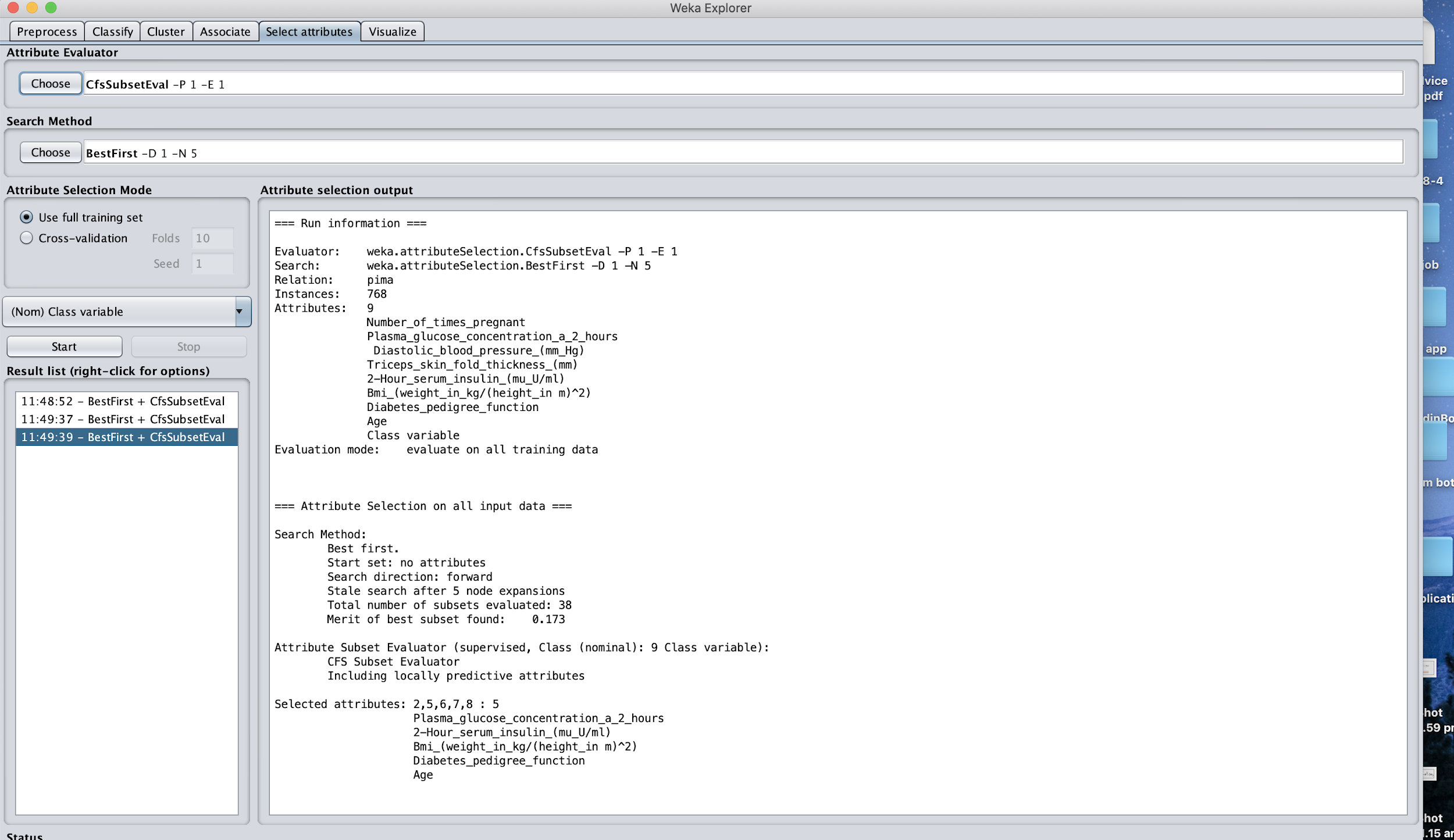
<https://www.analyticsvidhya.com/blog/2015/12/improve-machine-learning-results/>

X. Zhou, P. L. K. Ding and B. Li, (*2019)* "Improving Robustness of Random Forest Under Label Noise," *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa Village, HI, USA, pp. 950-958. doi: 10.1109/WACV.2019.00106

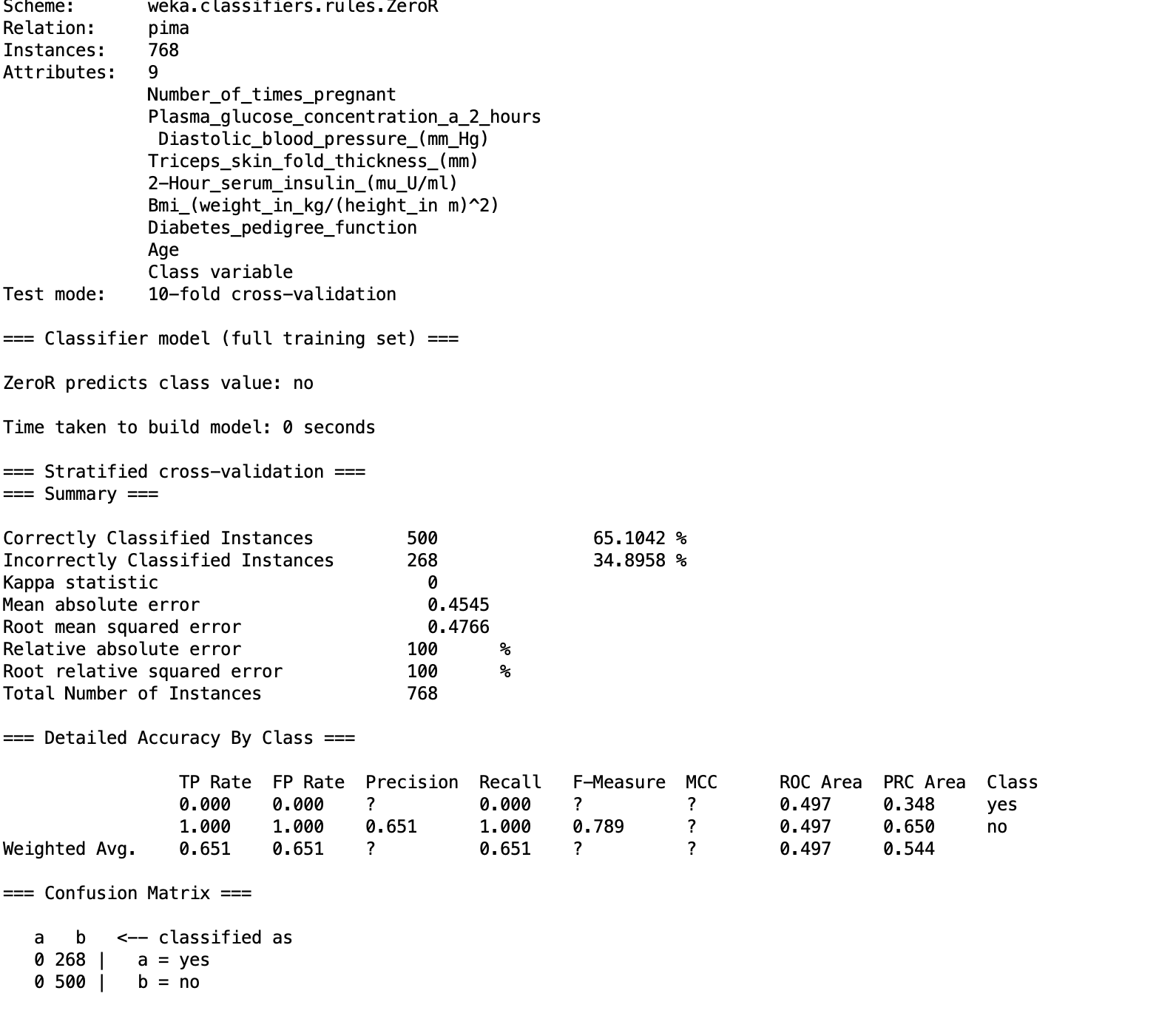
URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8658395&isnumber=8658235>

# Appendix

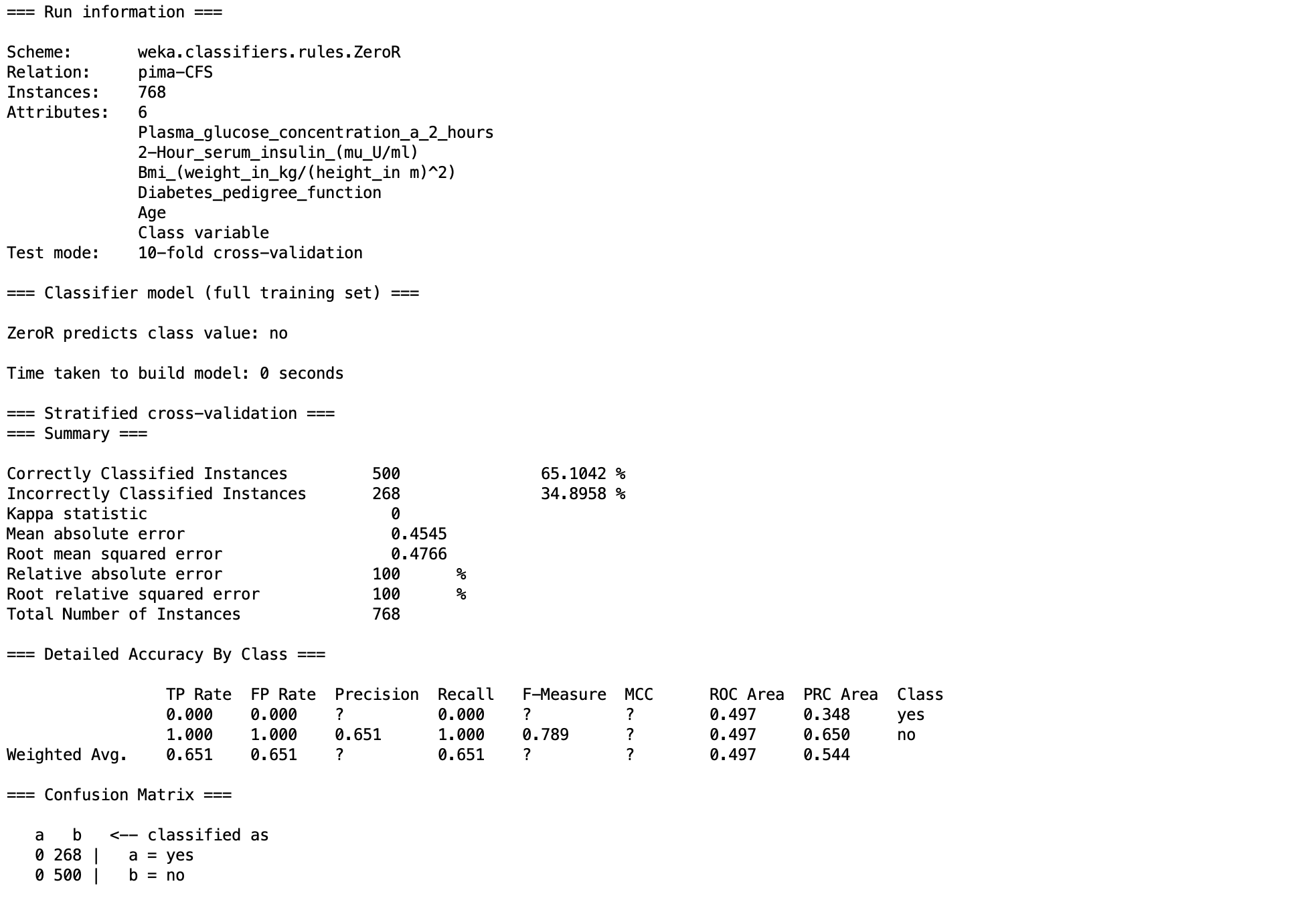
## CFS Attributes



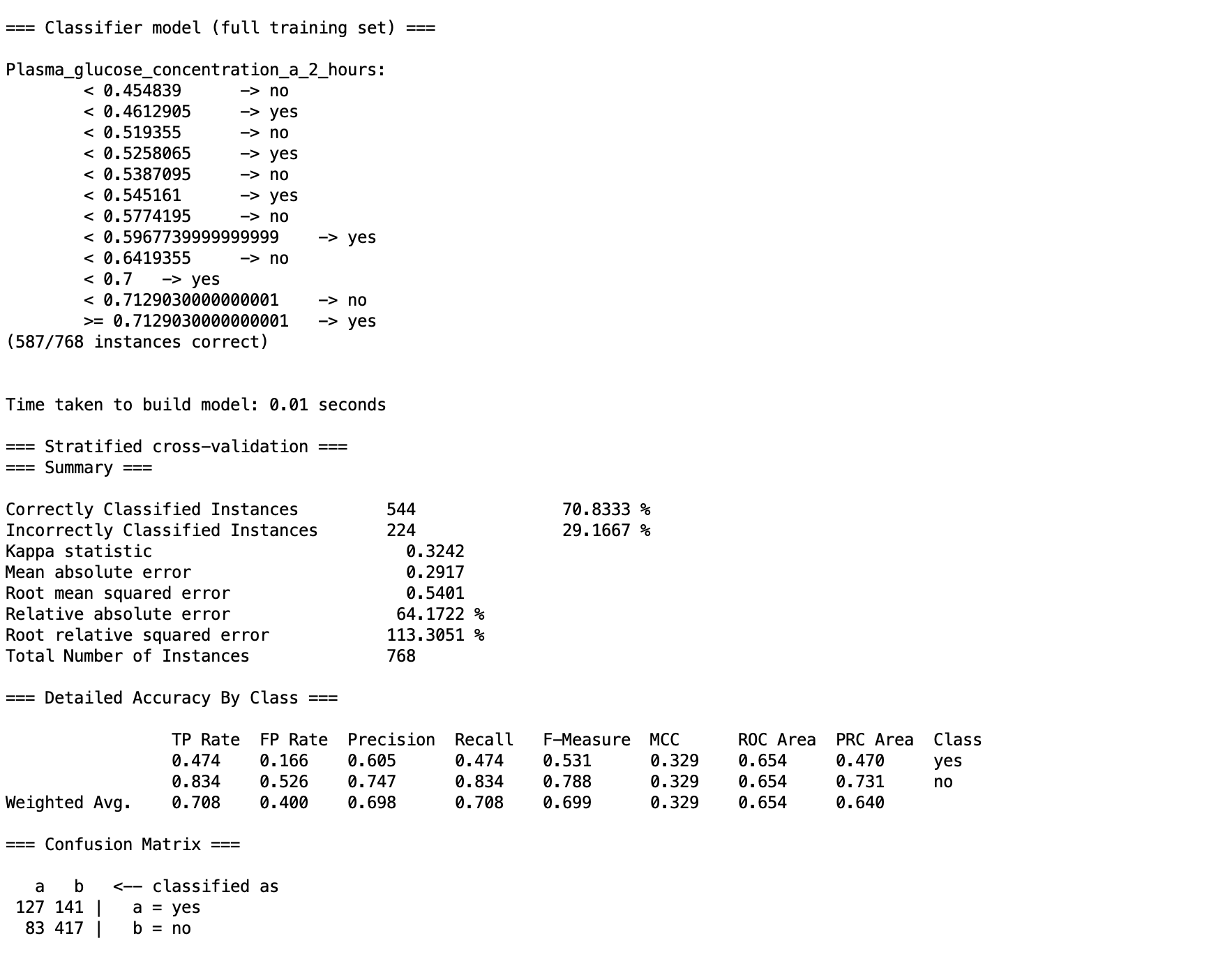
## ZeroR



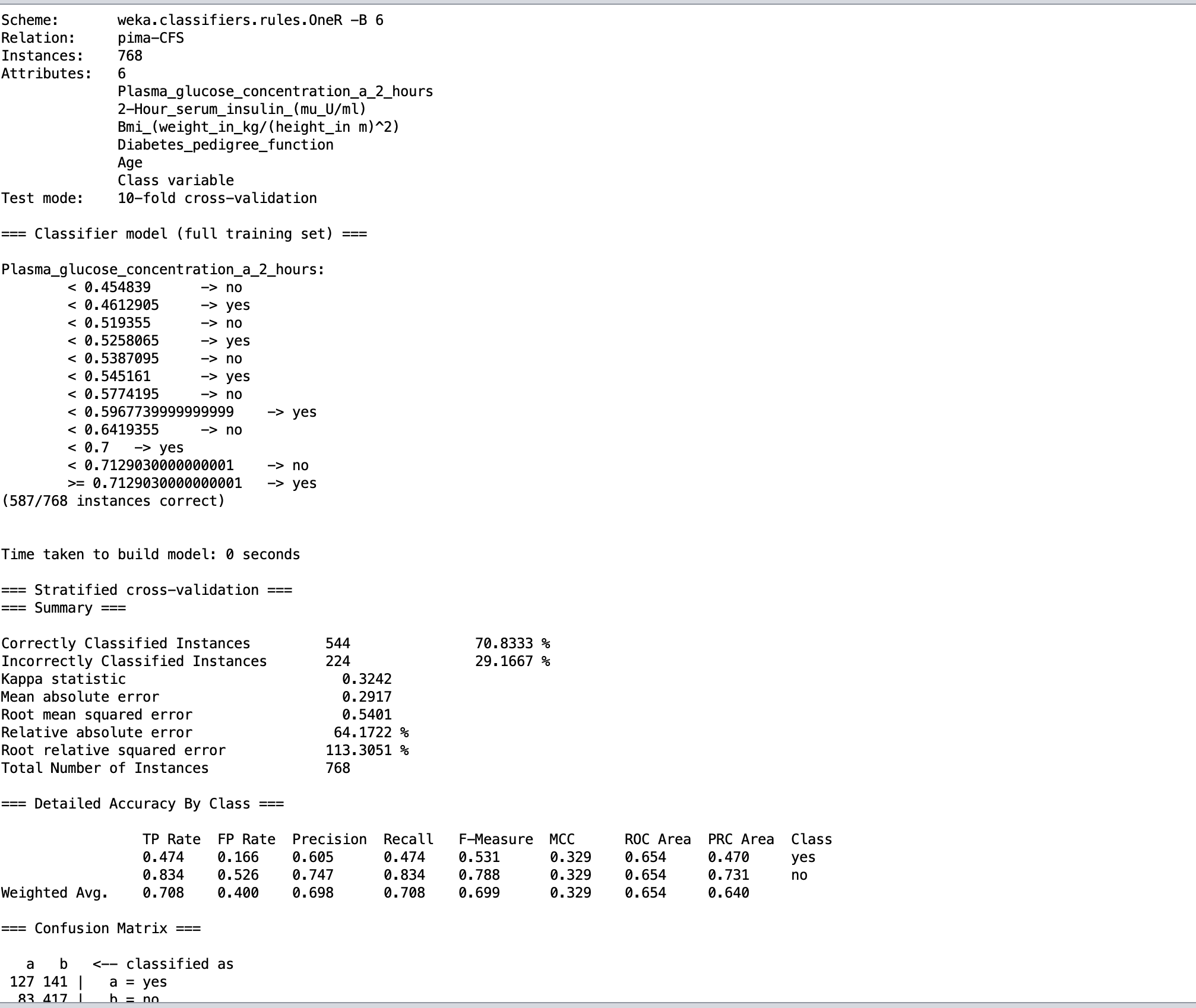
## ZeroR (CFS)



## 1R



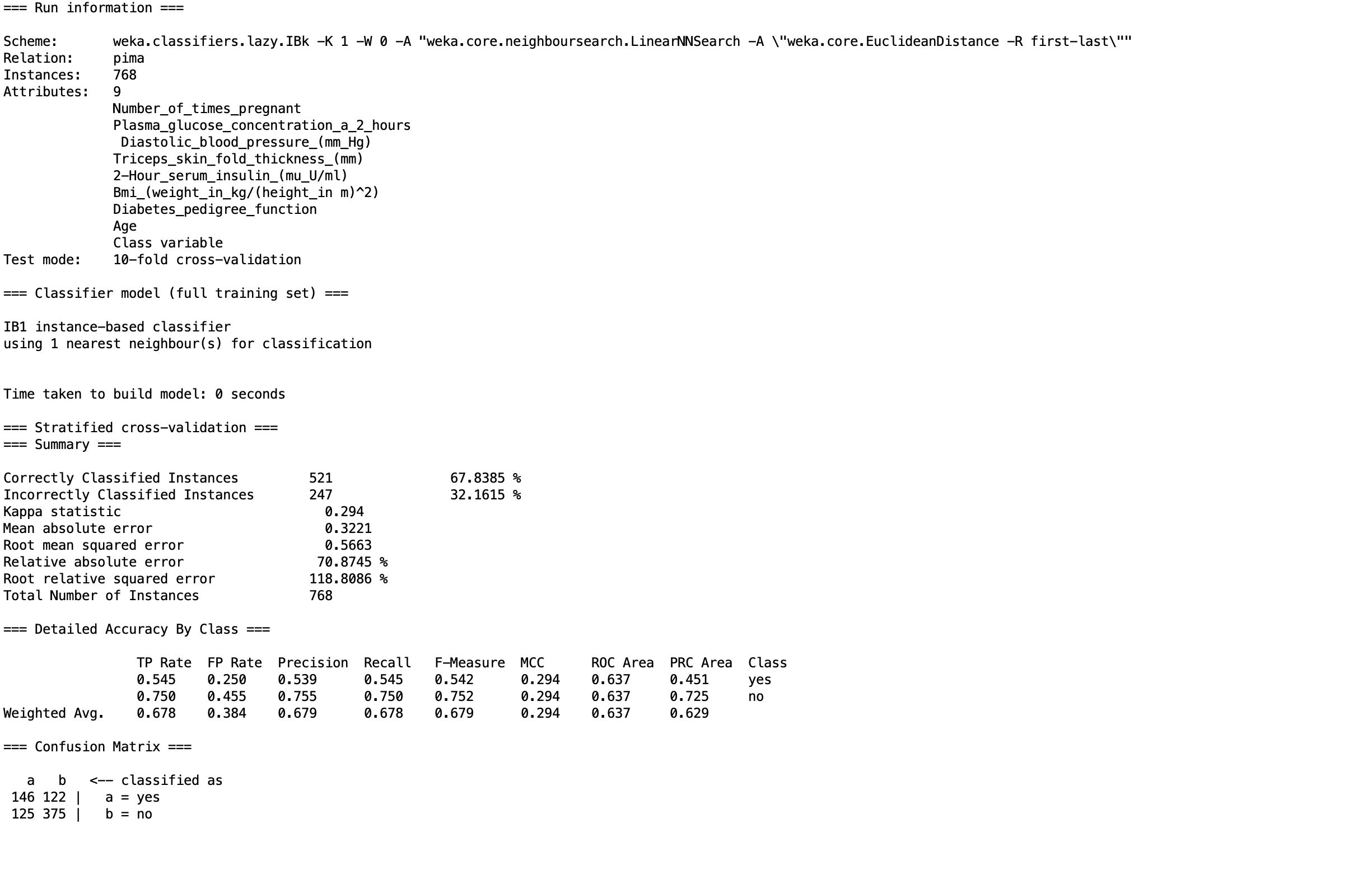
## 1R (CFS)



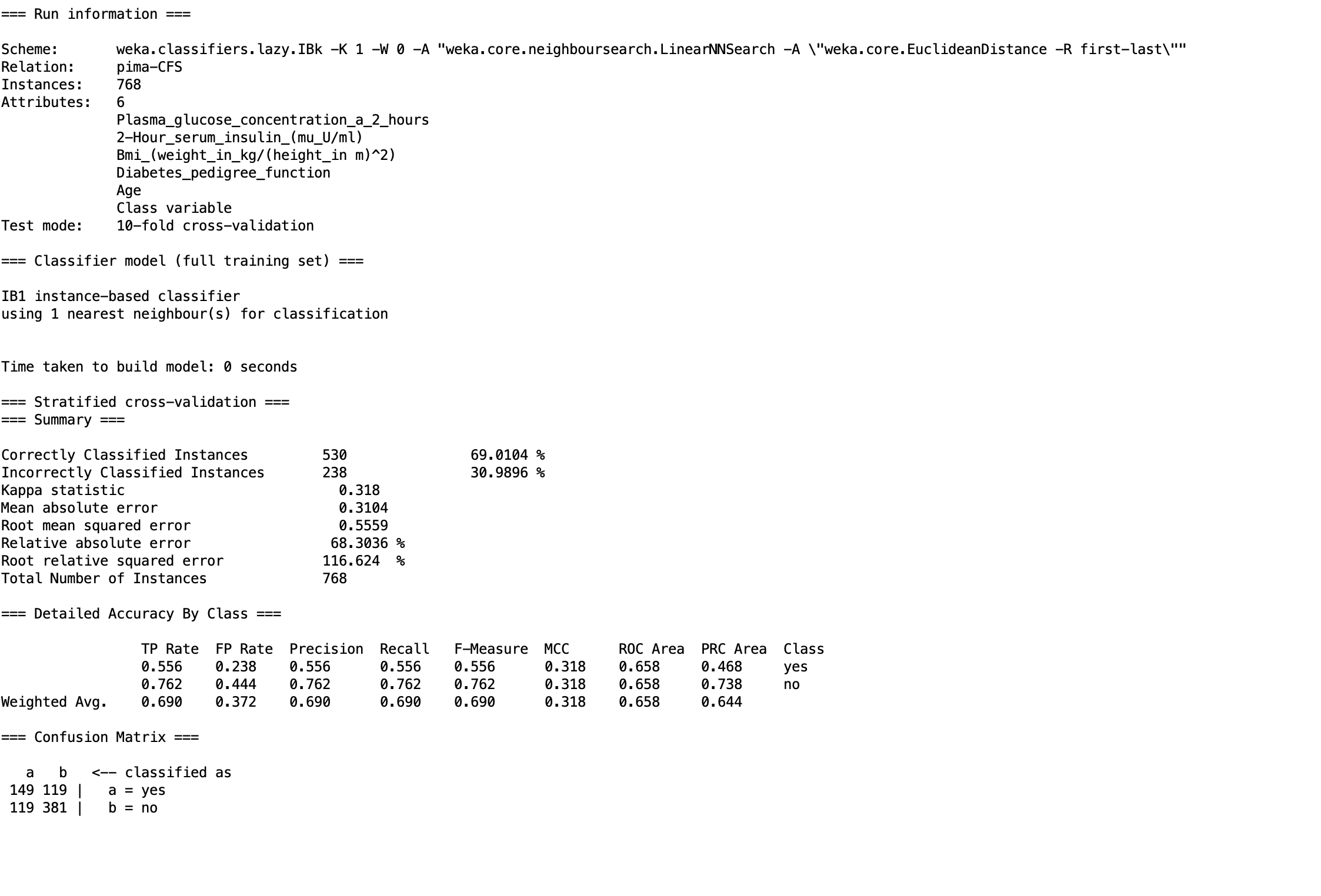
## 

## 

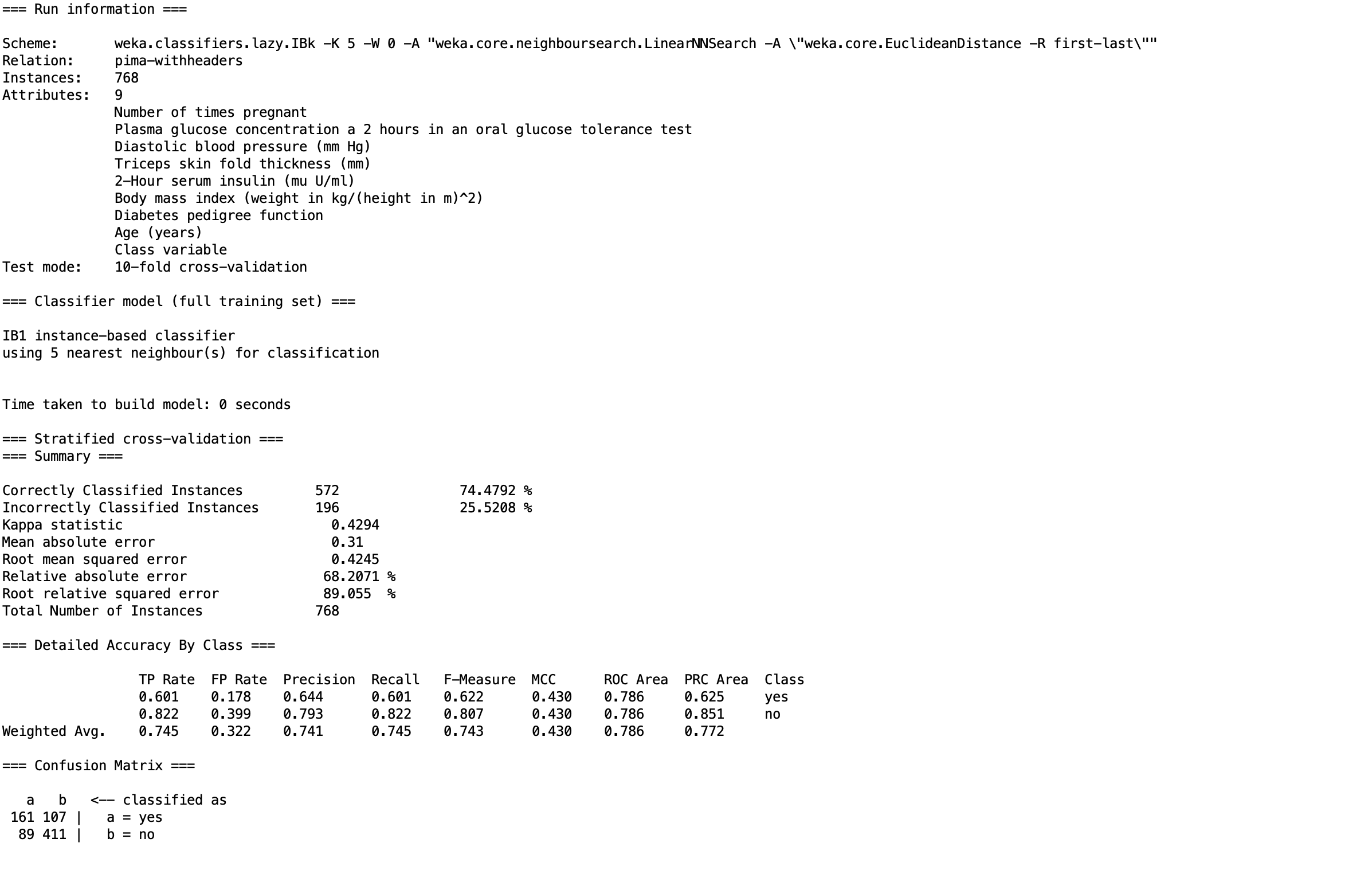
## 1NN



## 1NN (CFS)



## 5NN(Weka)

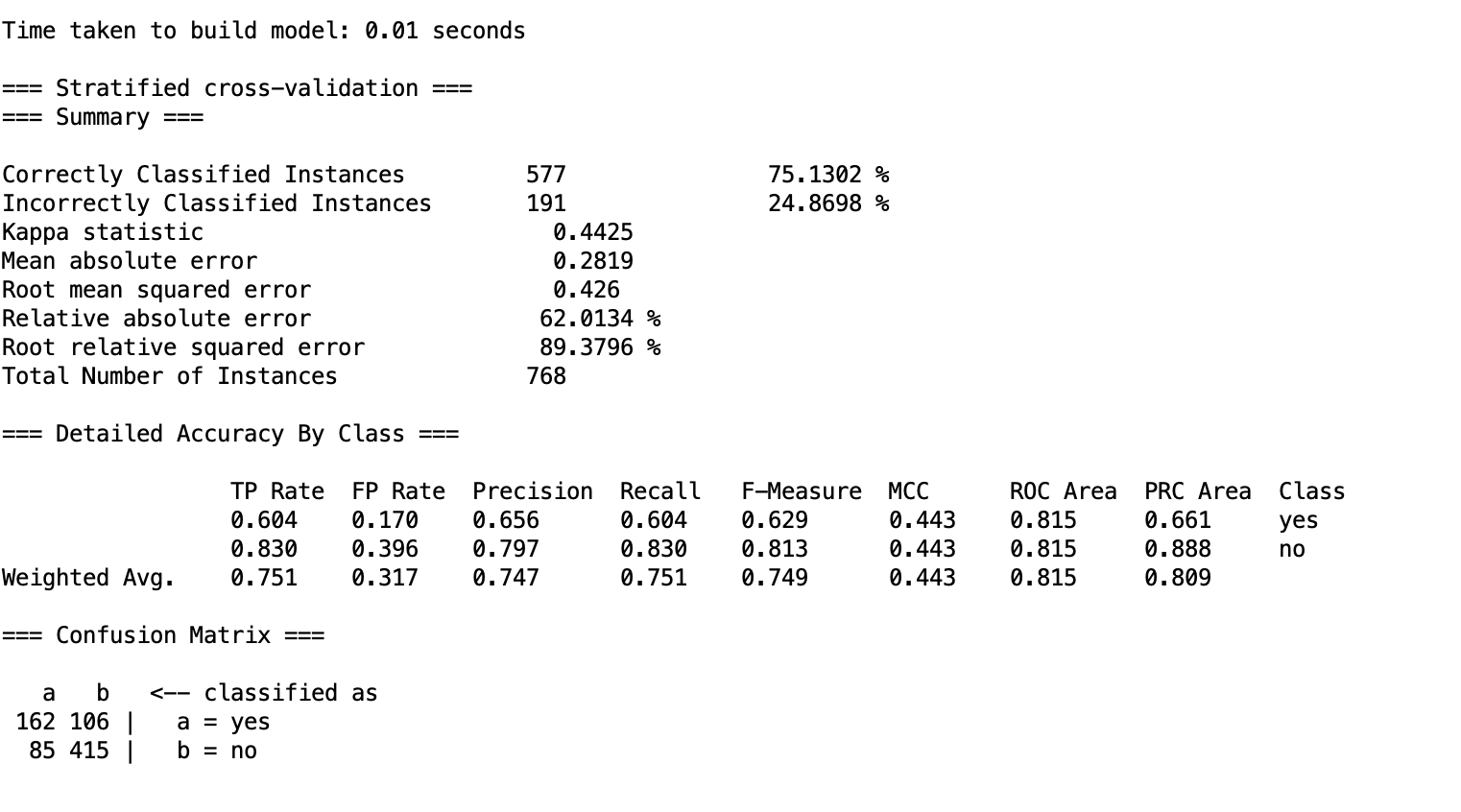


## 5NN(CFS)(Weka)

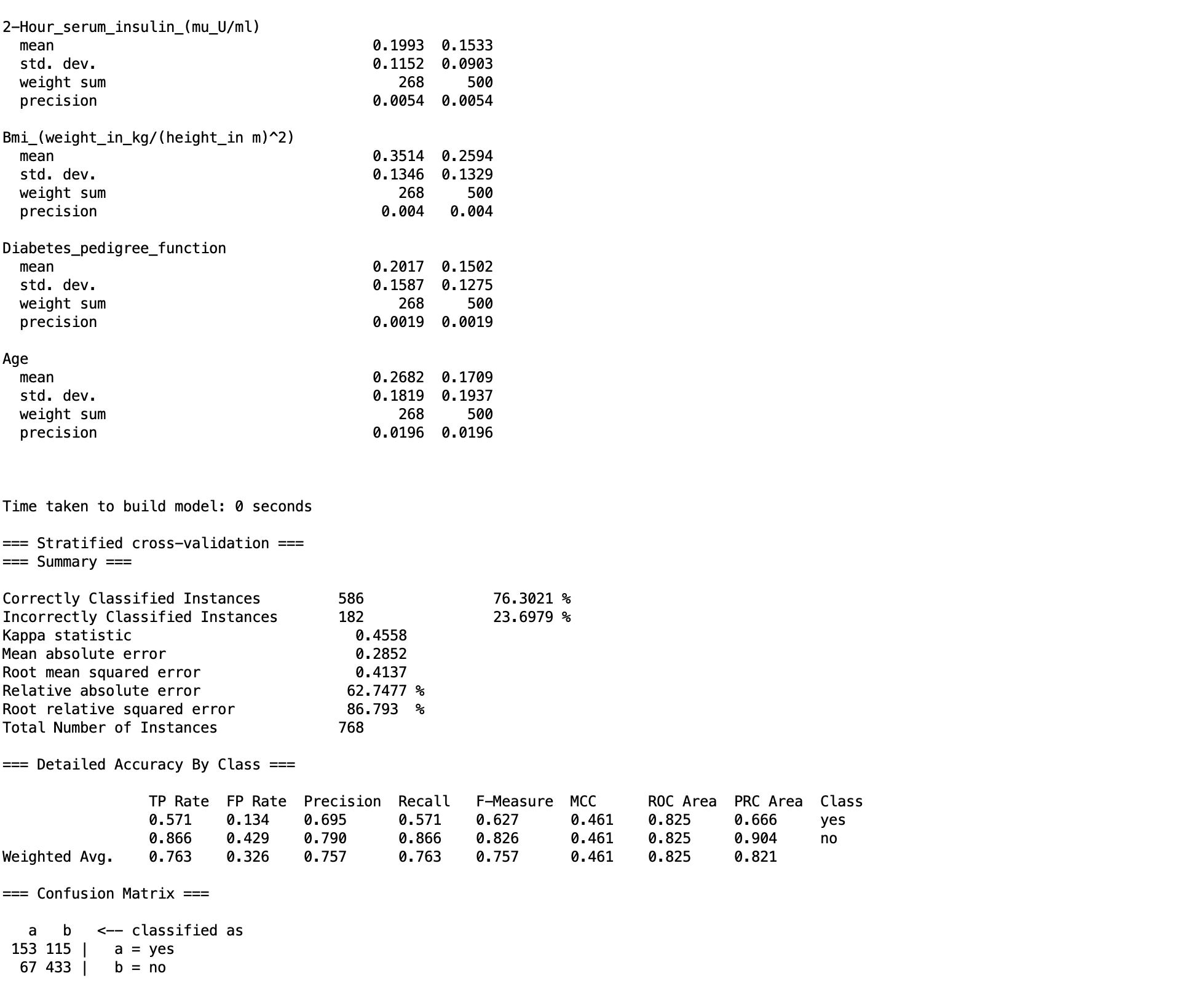
## 

## 

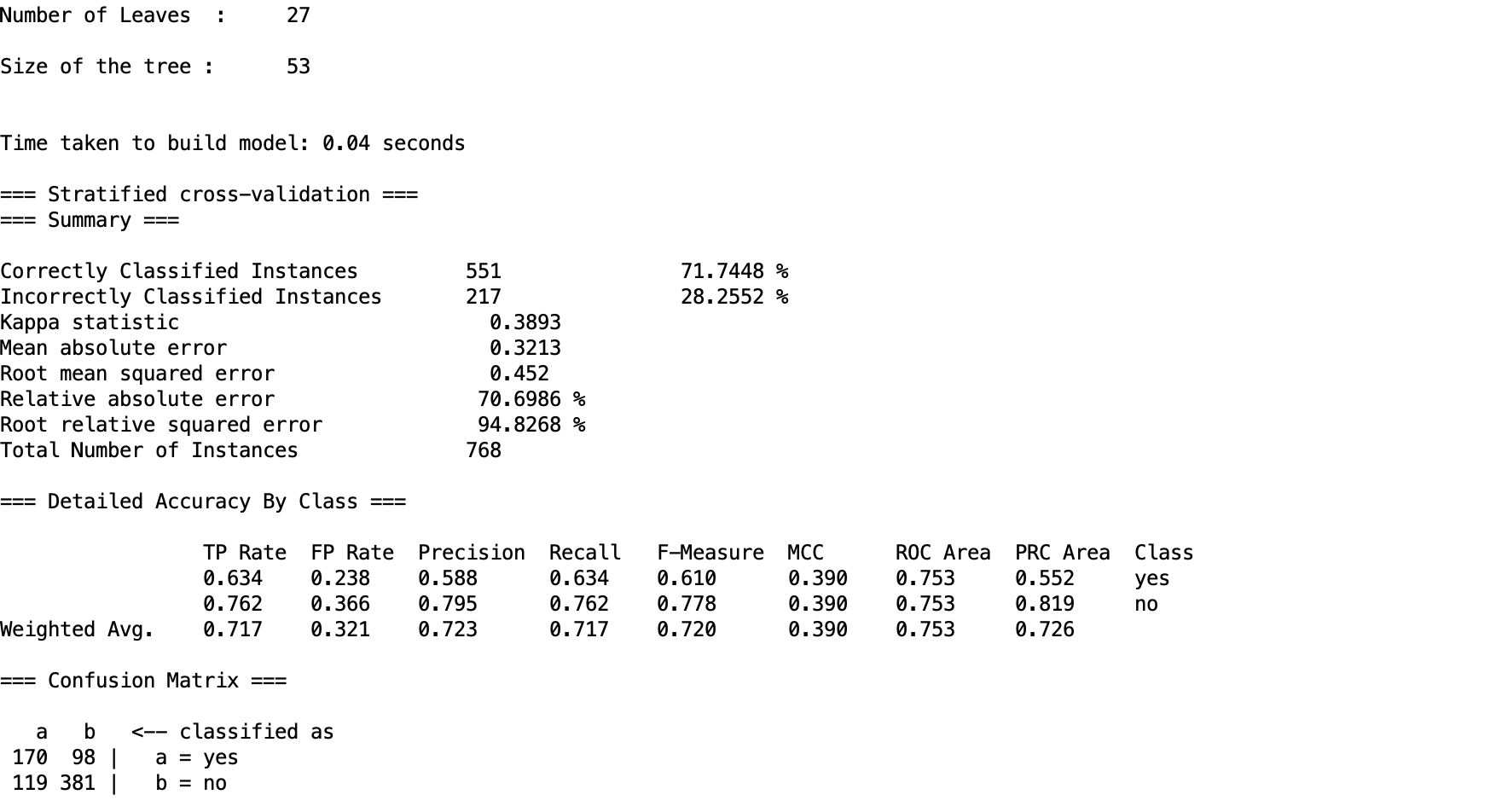
## NB(Weka)



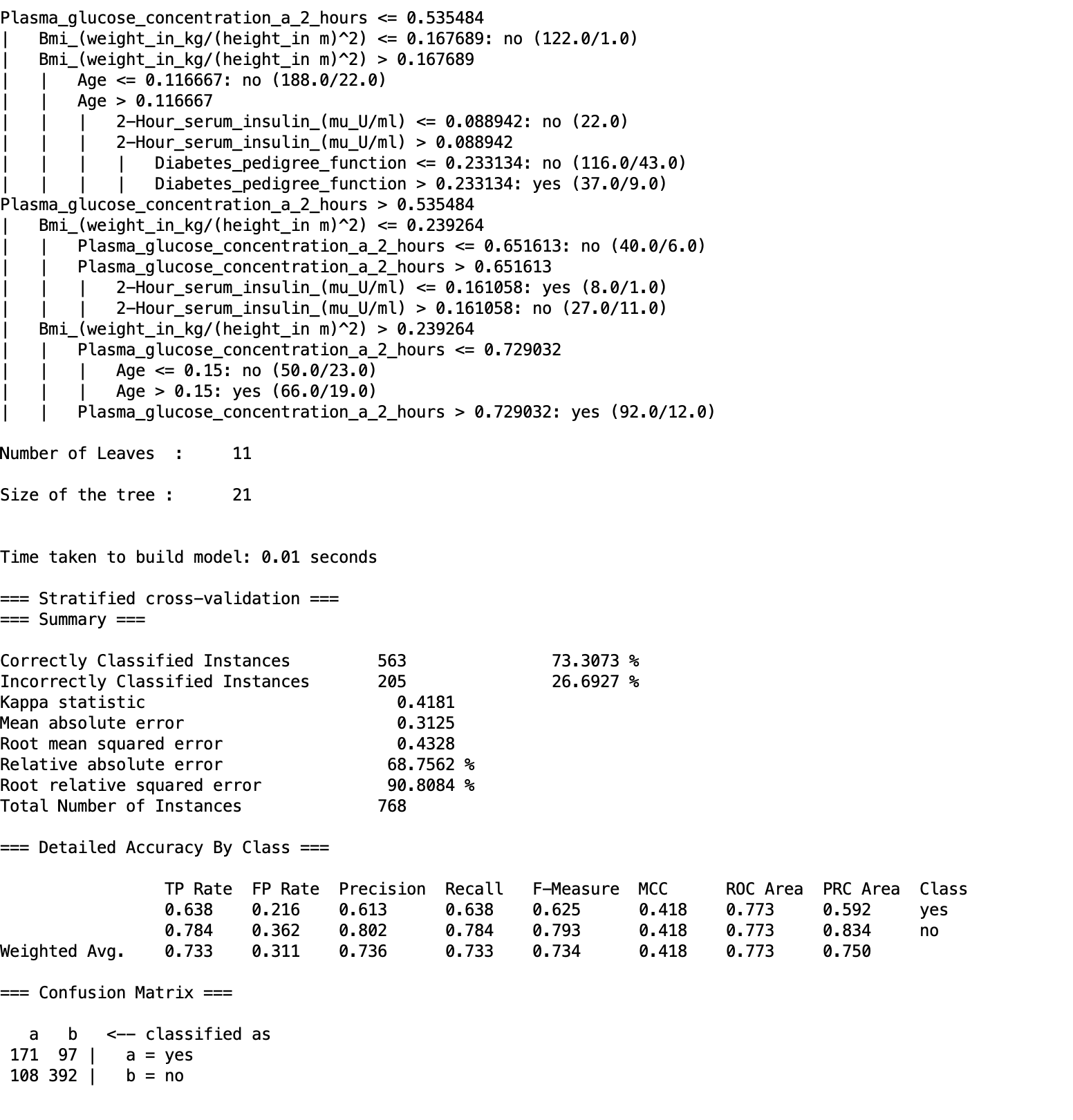
## NB (CFS) (Weka)



## DT



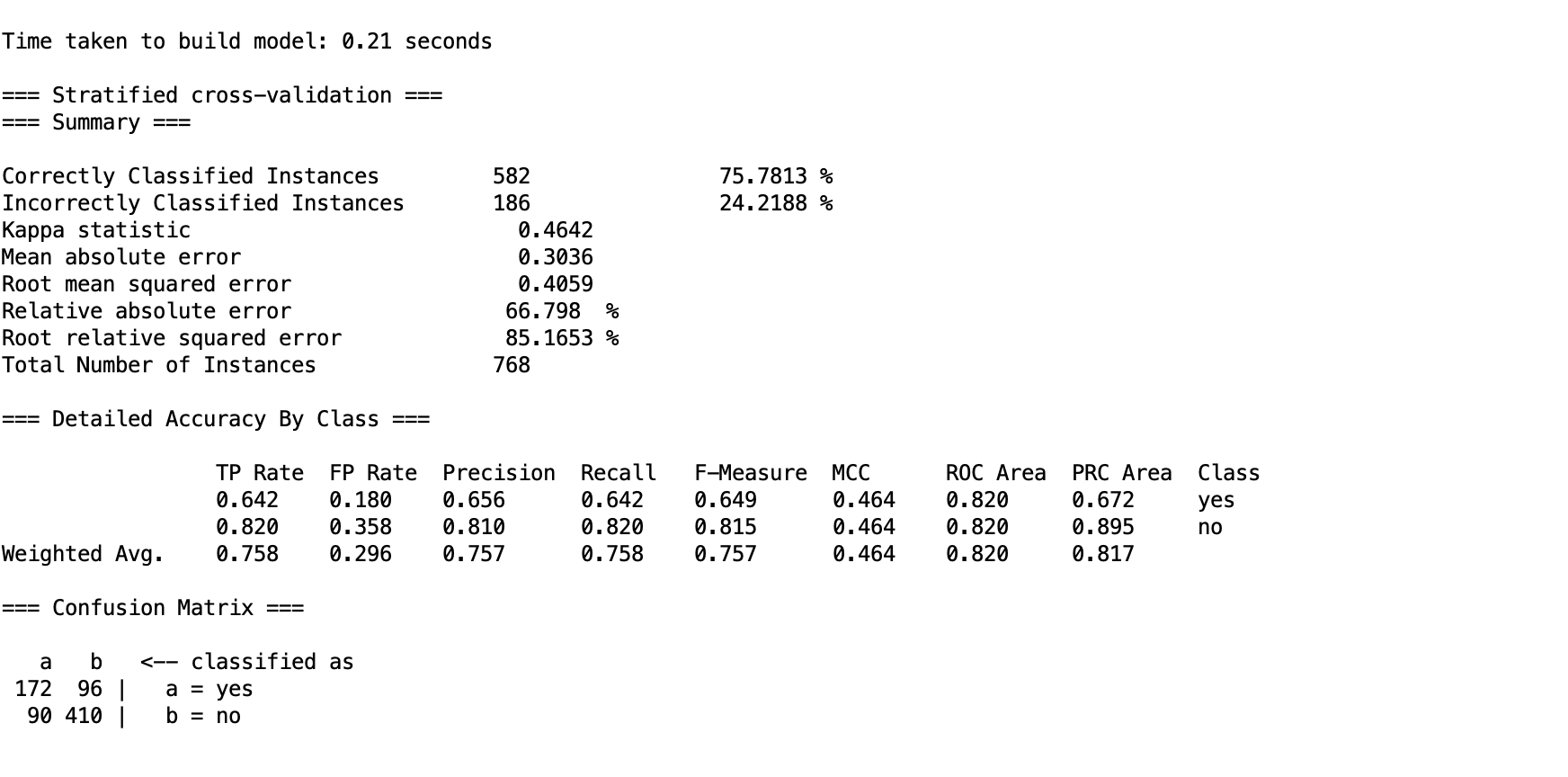
## DT (CFS)



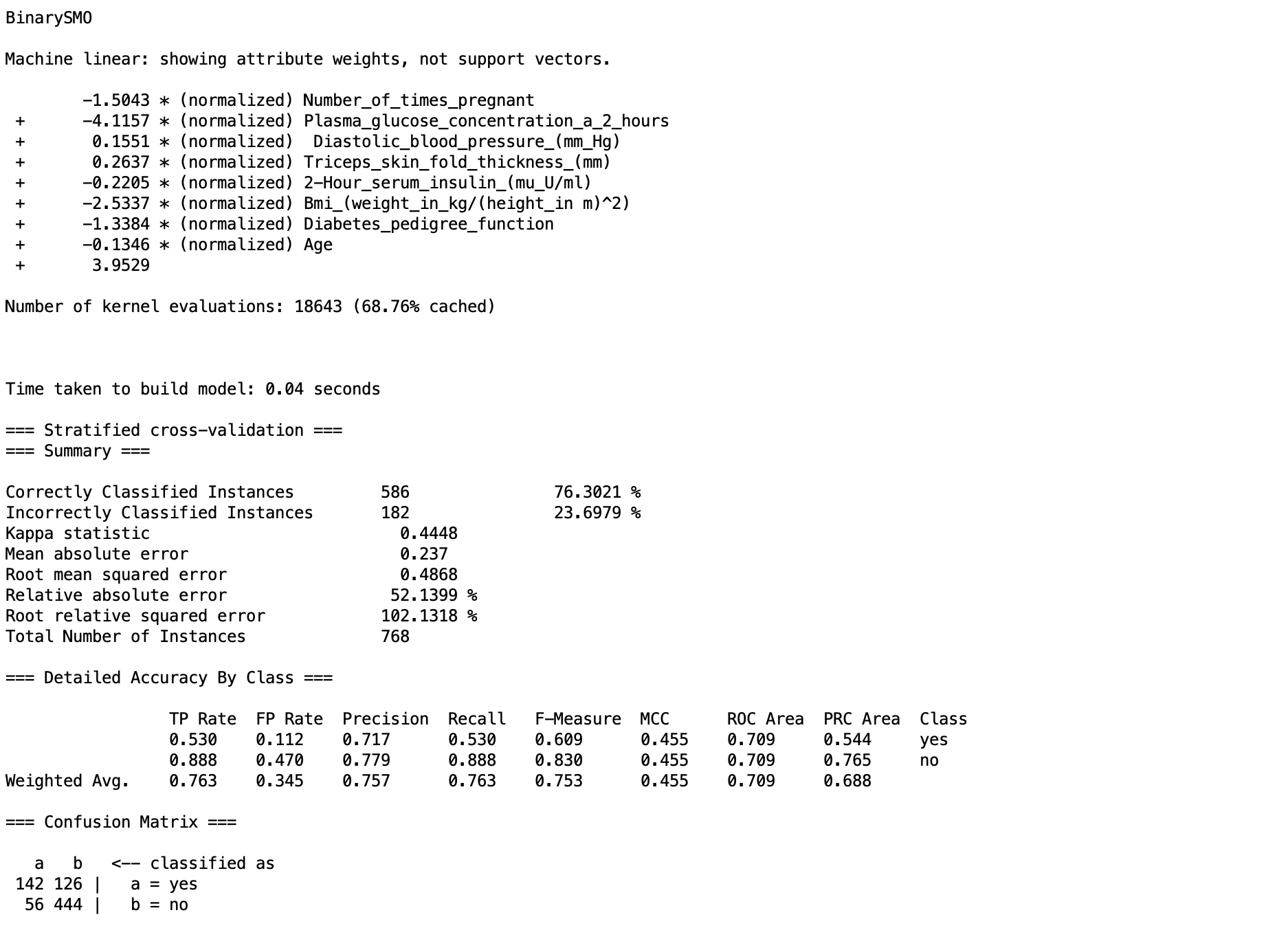
## MLP

## 

## MLP (CFS)

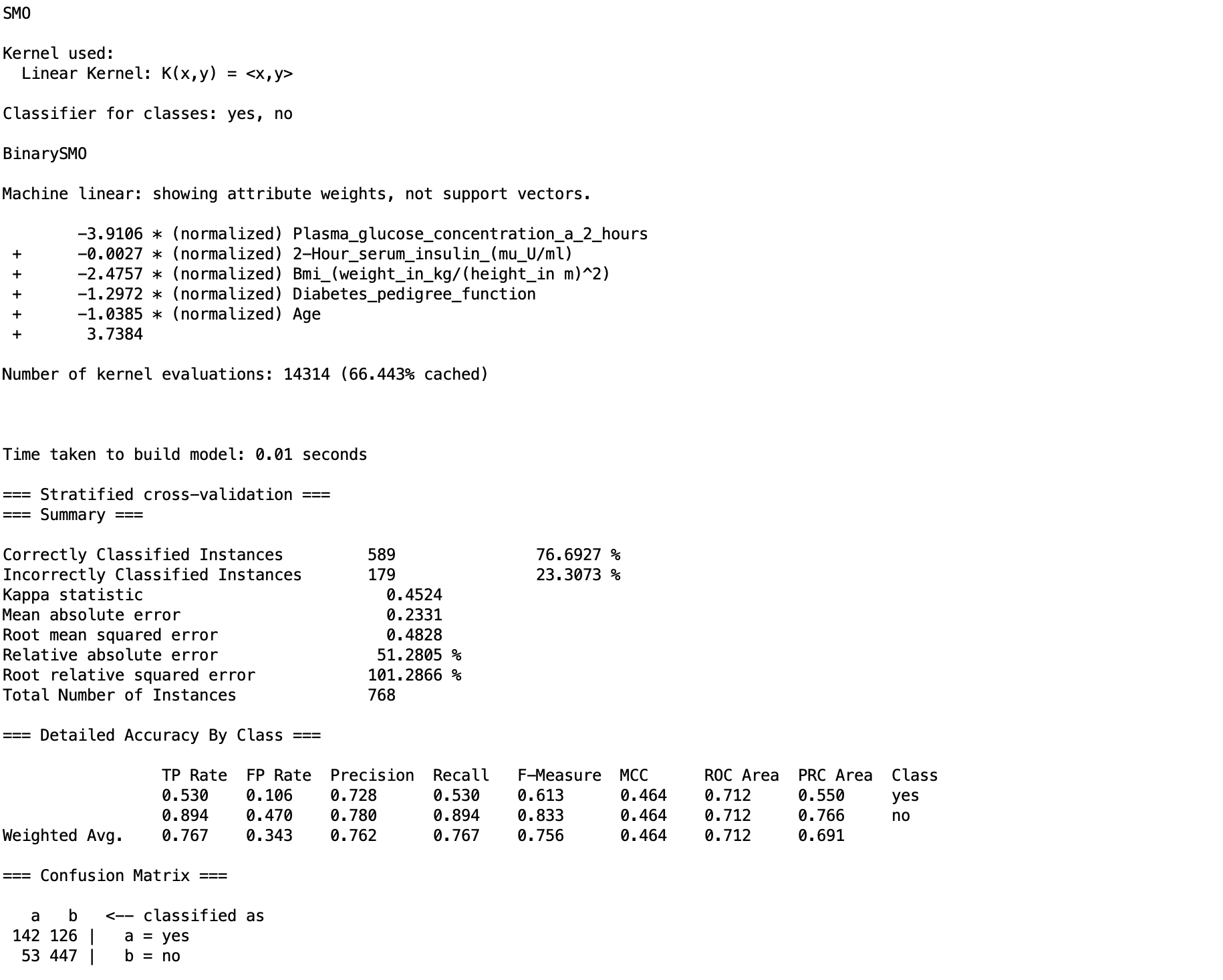


## SVM

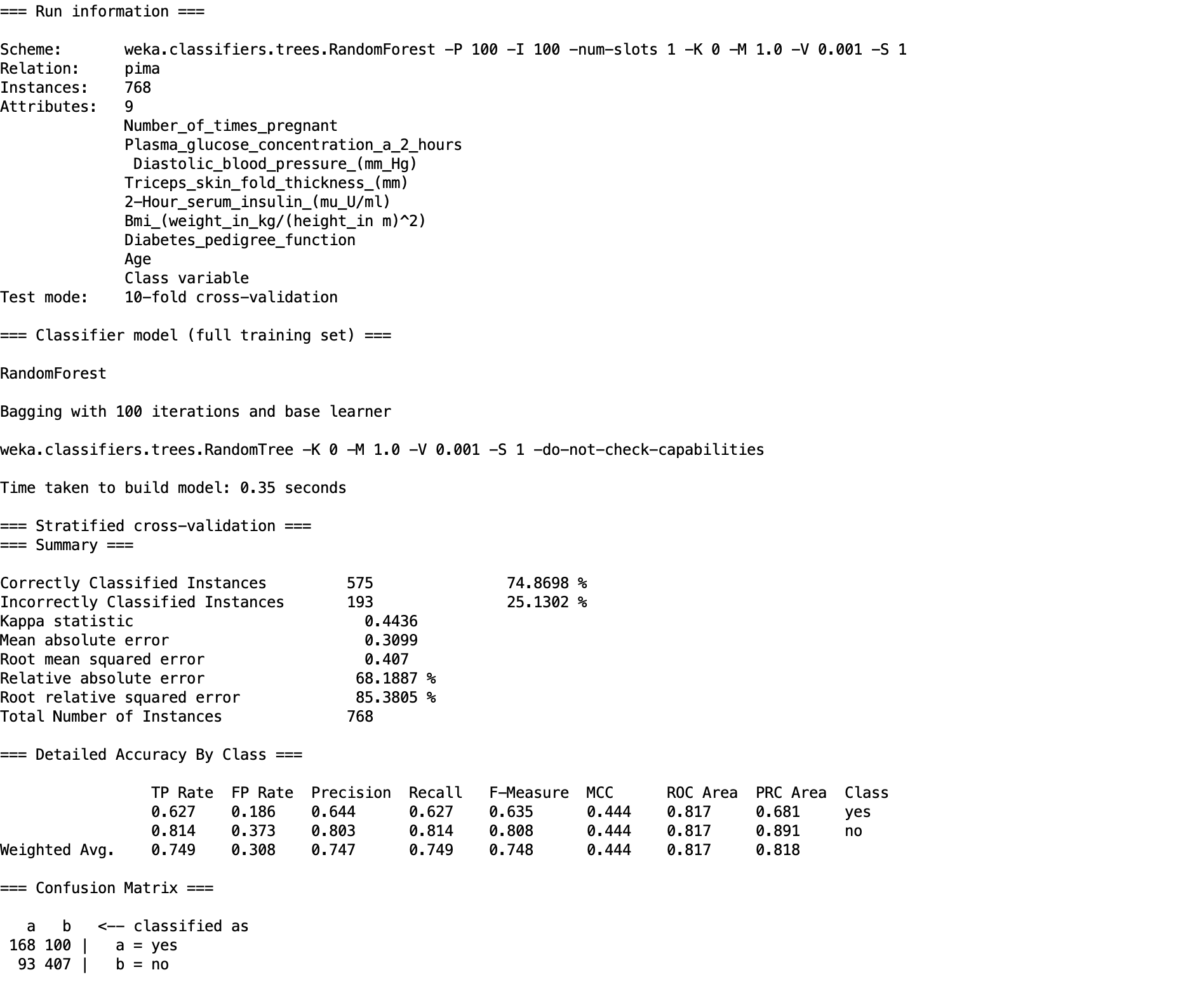
****

## 

## SVM (CFS)

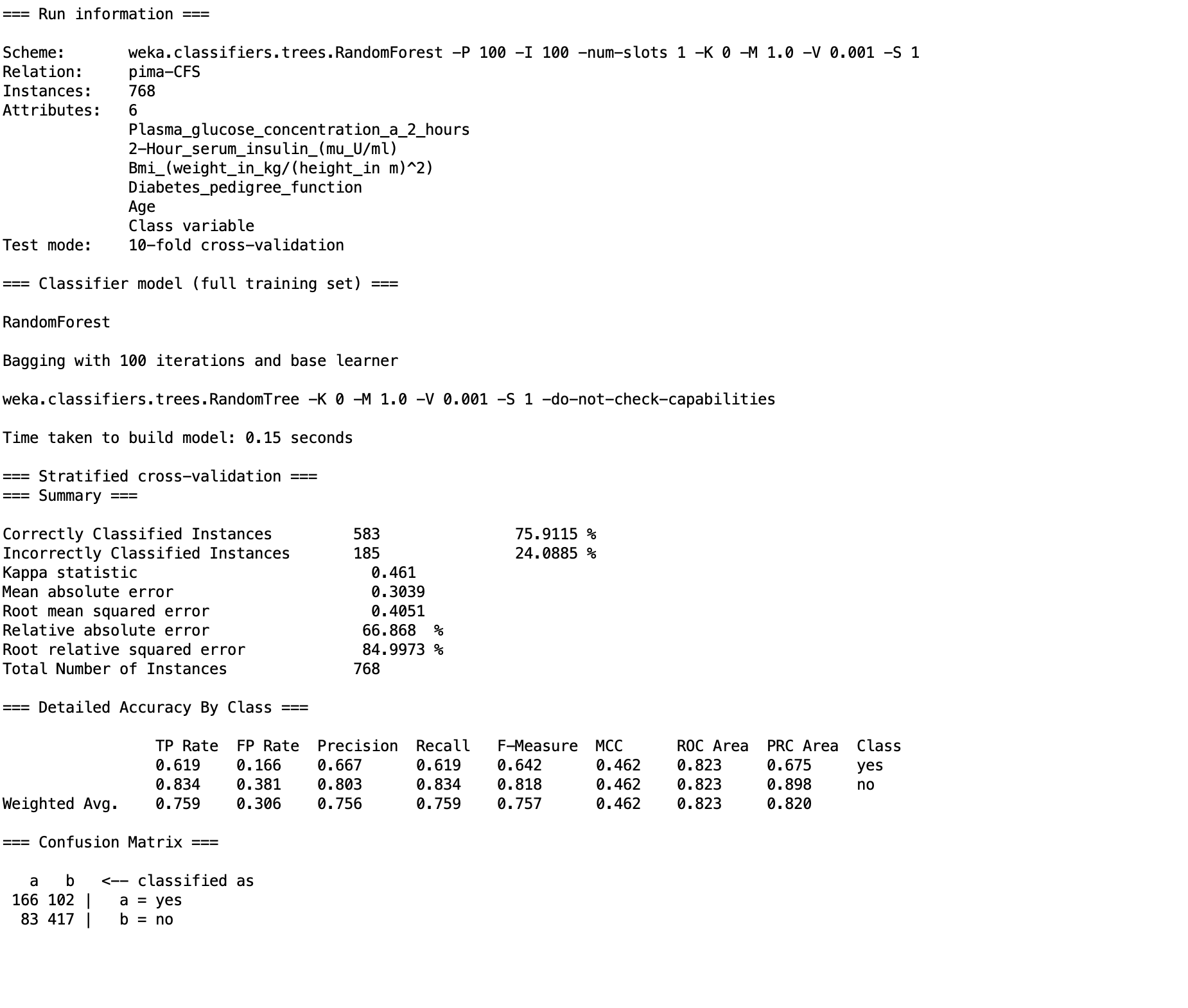


## RF

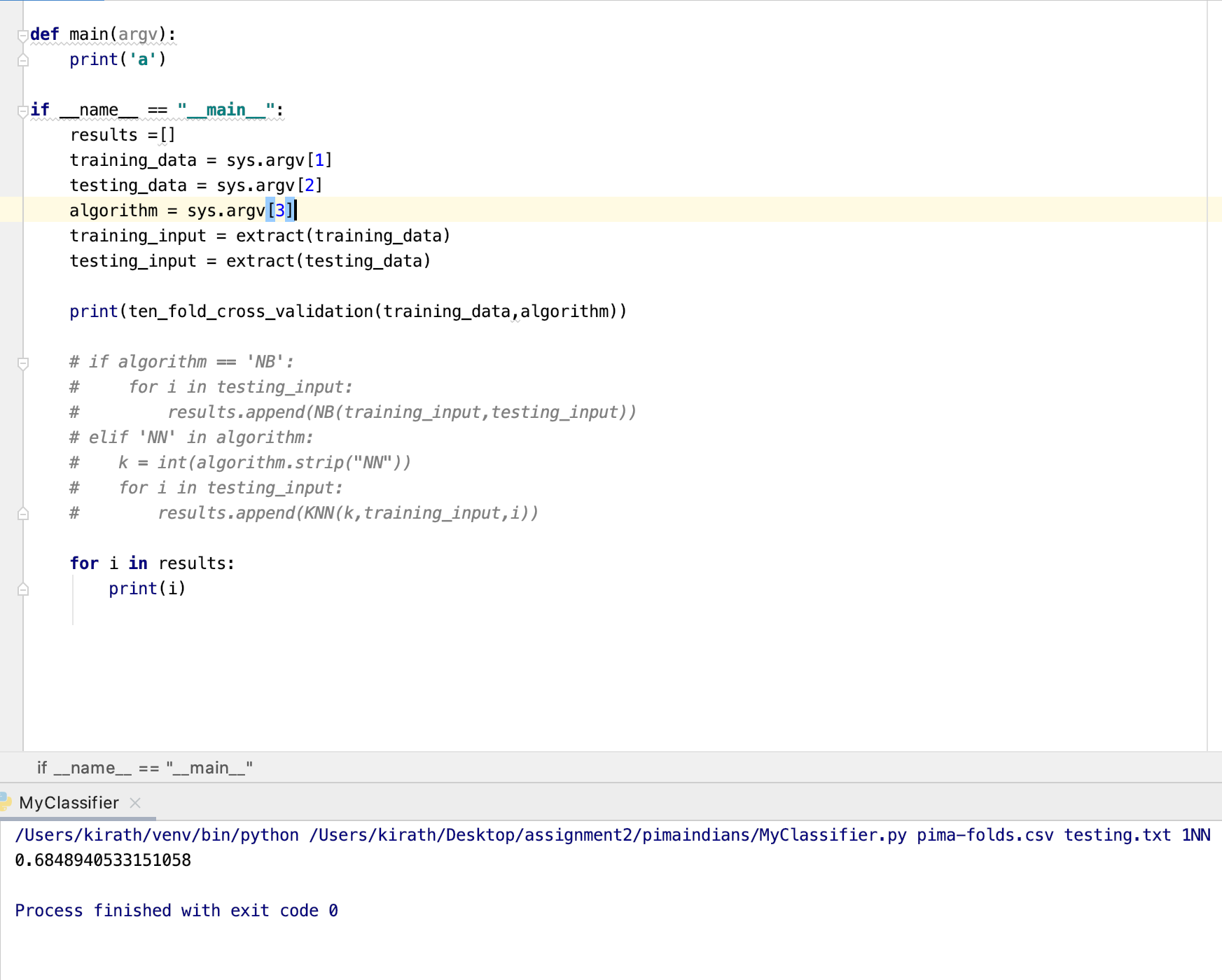


## 

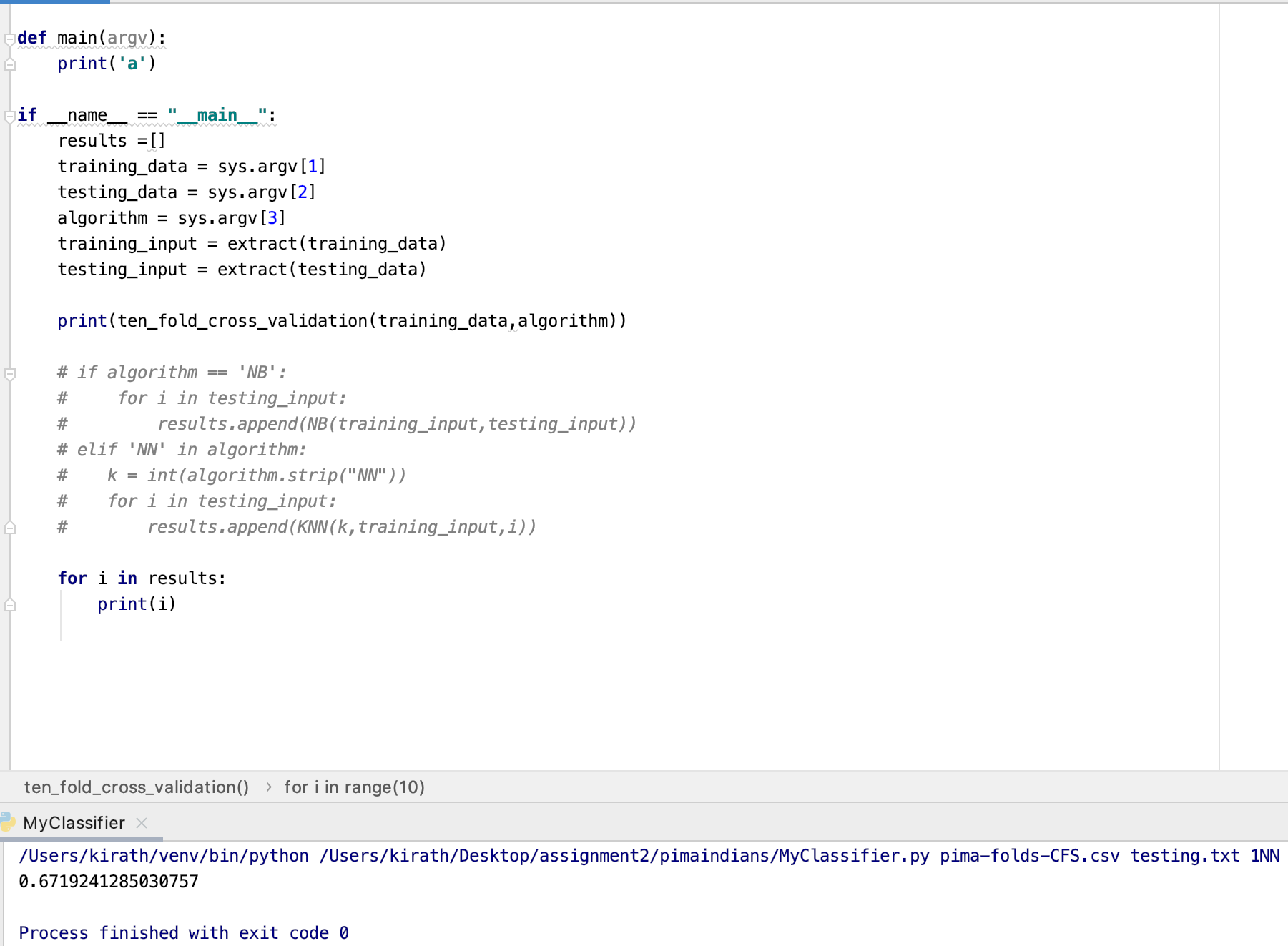
## RF (CFS)



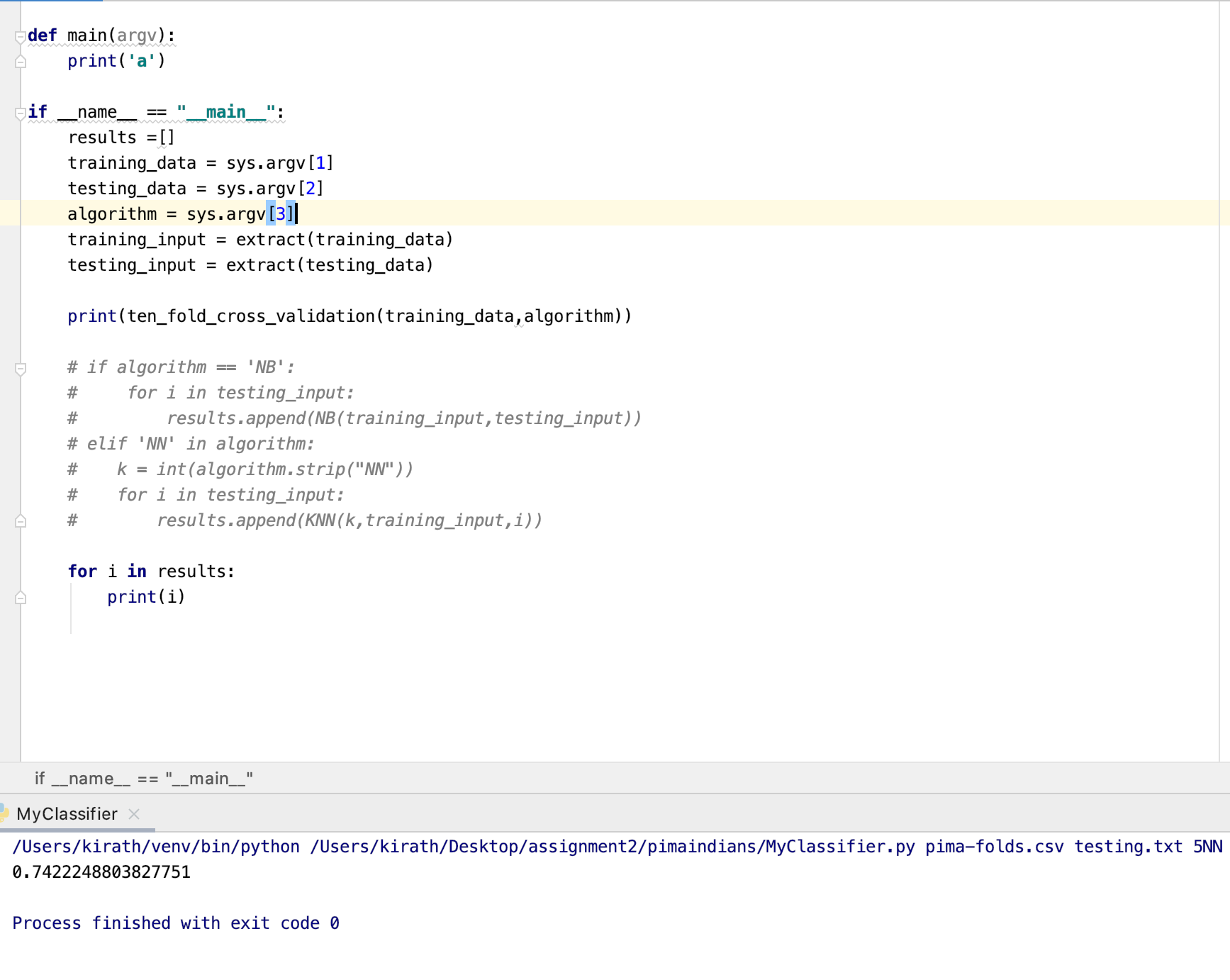
## 1NN (Implemented)



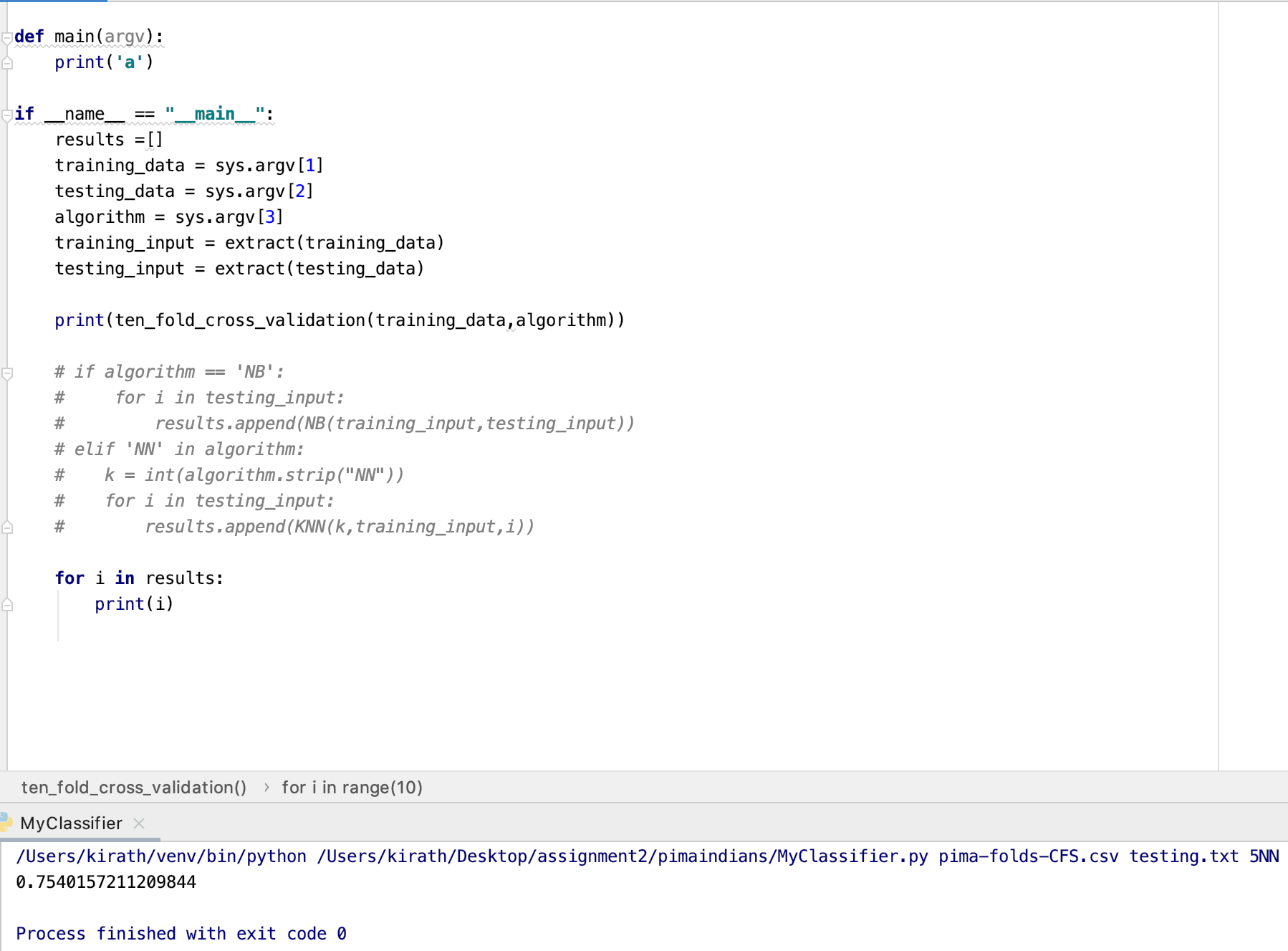
## 1NN (CFS) (Implemented)



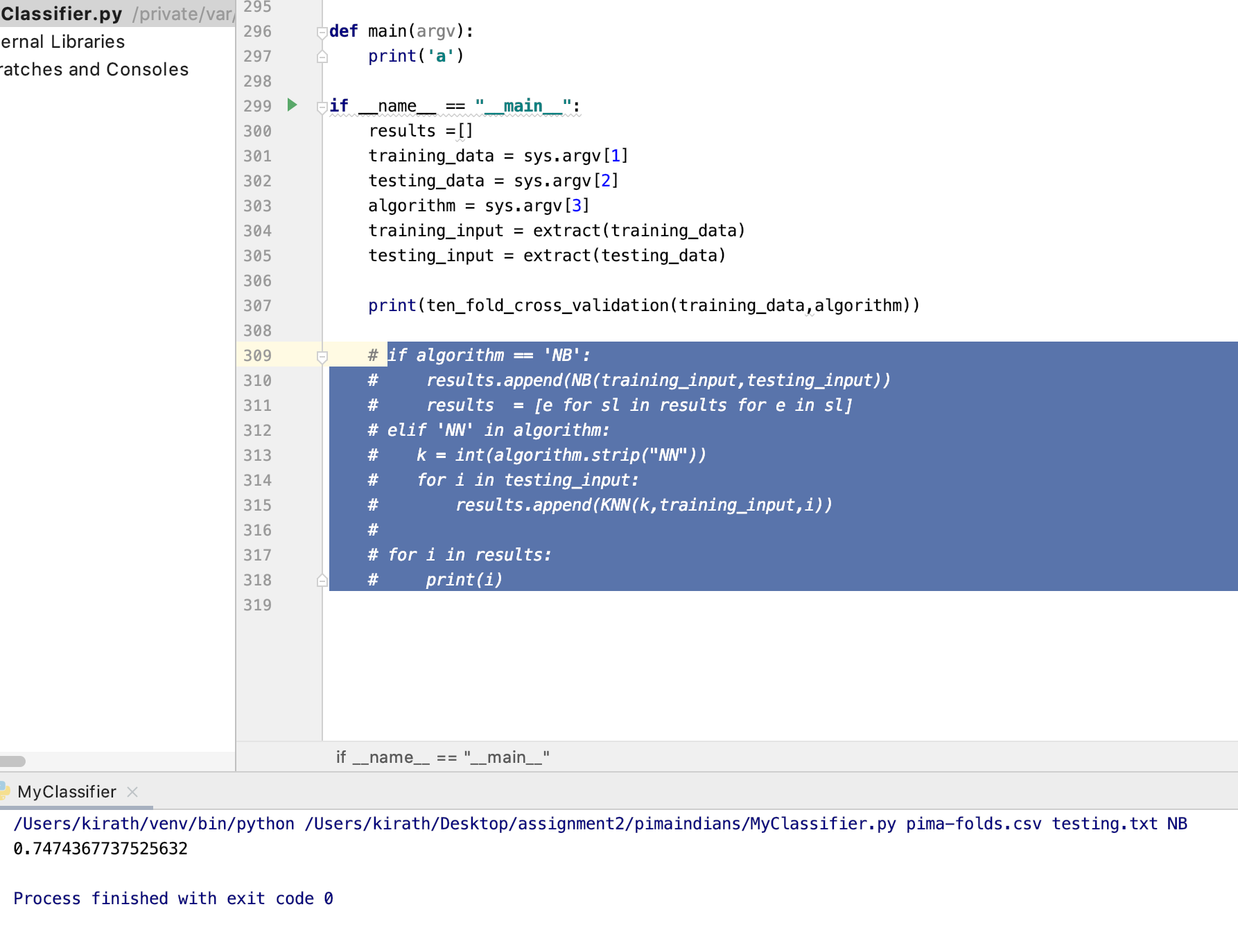
## 5NN(Implemented)



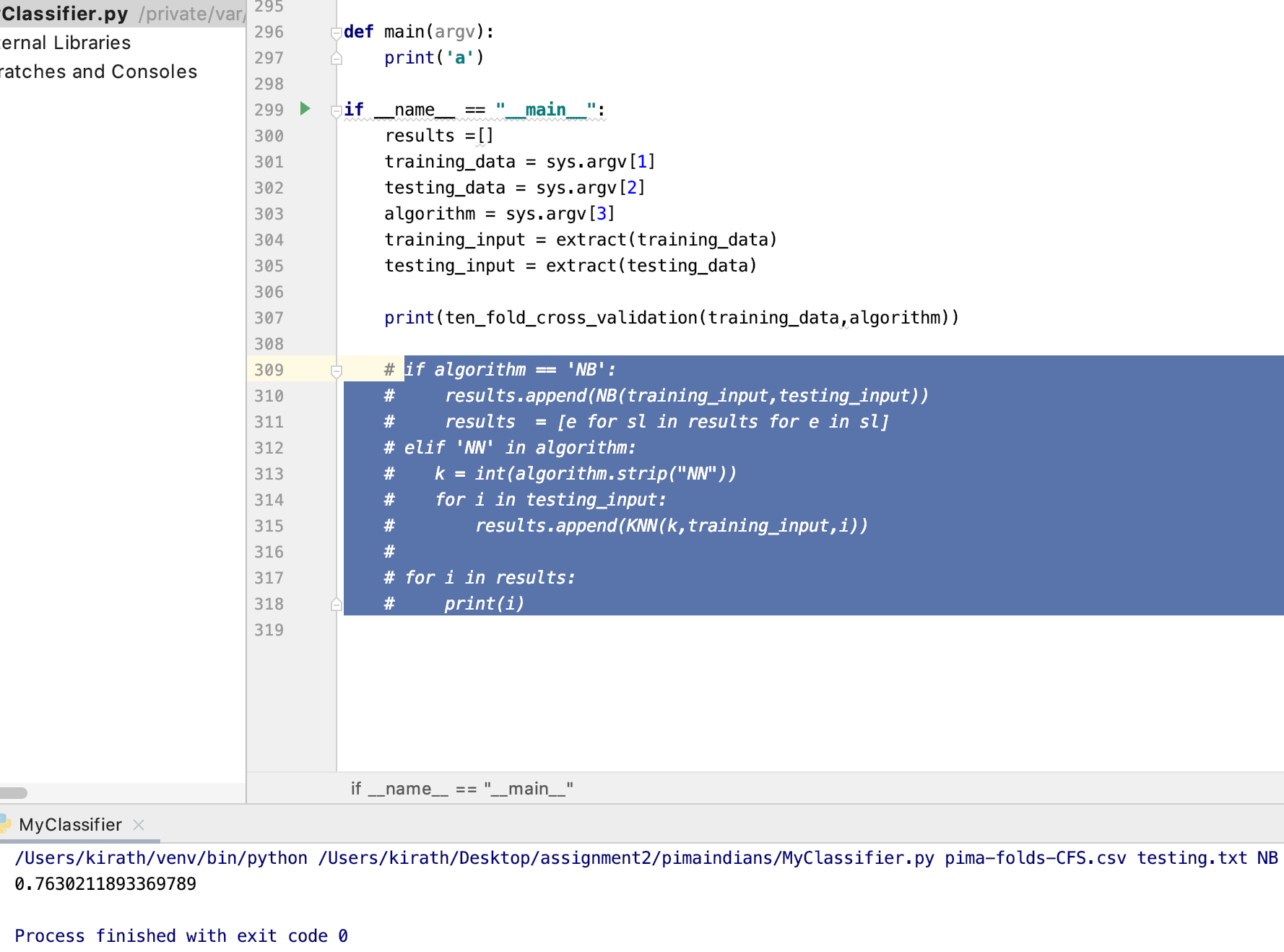
## 5NN (CFS)(Implemented)



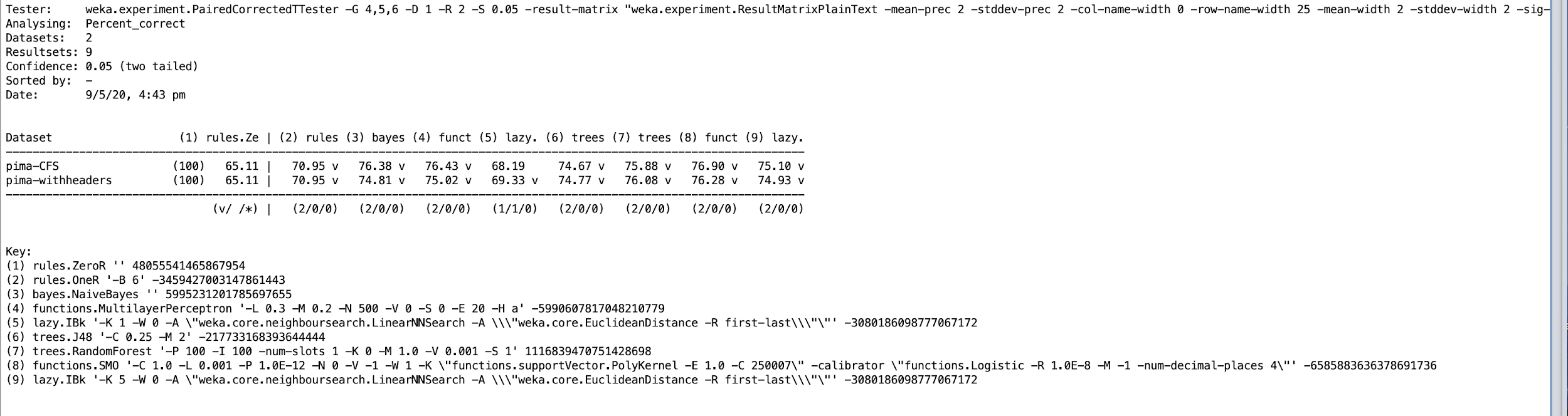
## NB (Implemented)



## NB (CFS)(Implemented)



## Analysis of ZeroR



## Analysis of 1R

## Analysis of NB

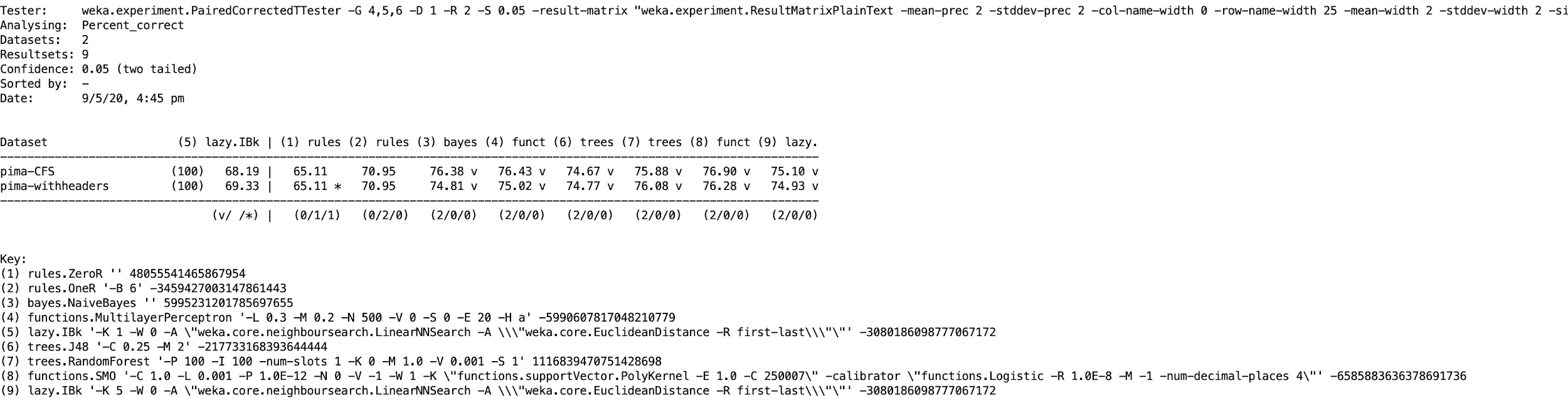
## Analysis of MLP

## 

## 

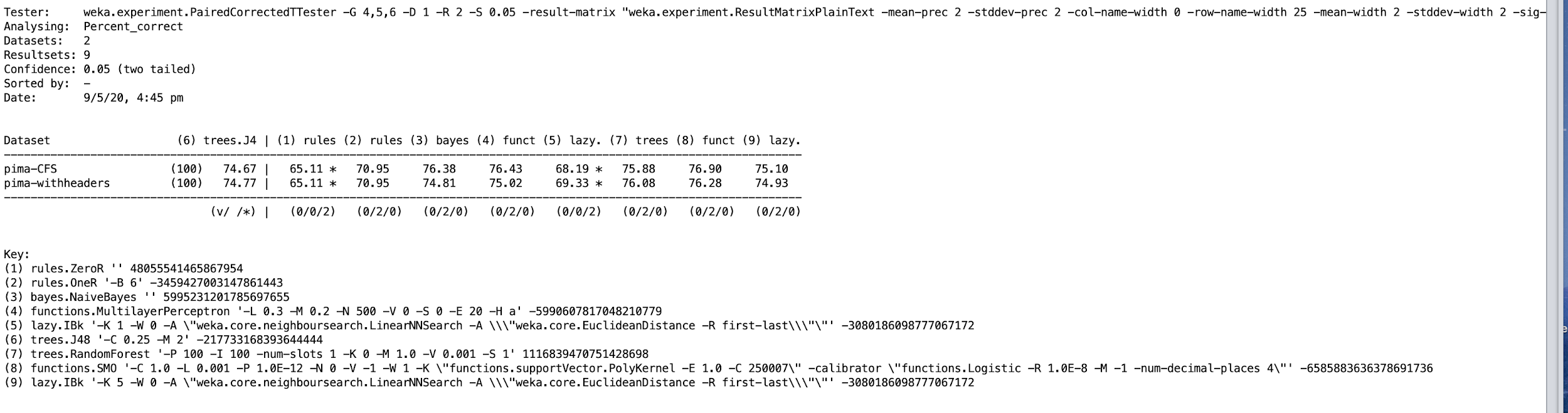
## 

## Analysis of 1NN

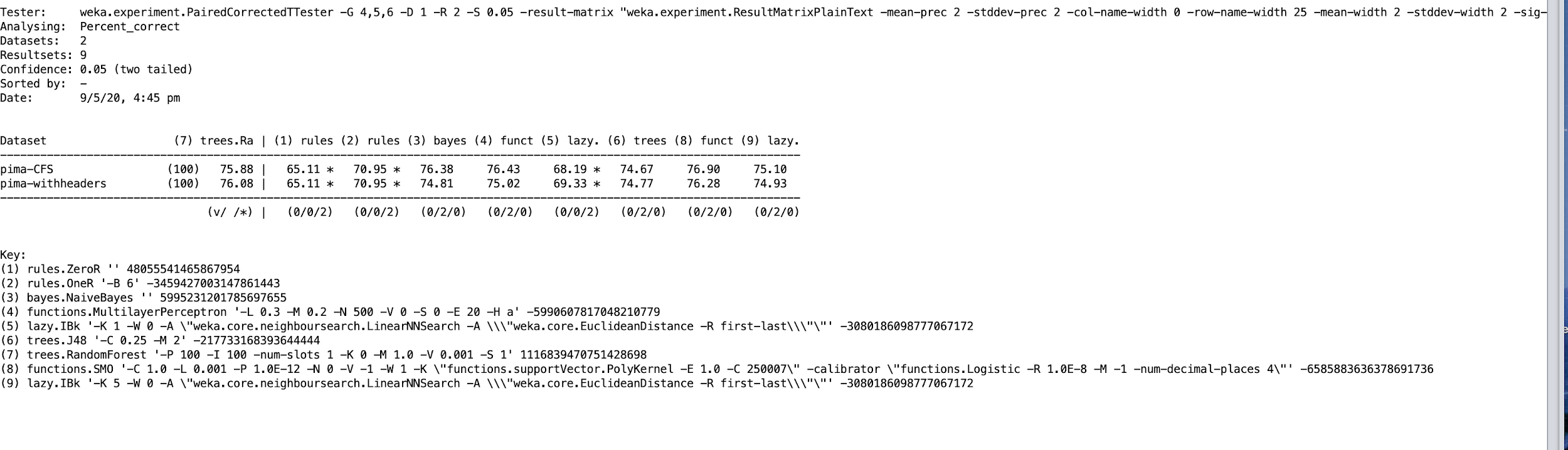


## Analysis of 5NN

## Analysis of DT(J48)



## Analysis of RF



## Analysis of SVM (SMO)

## 