

Table of Contents

Title	Page No.
Bona-fide Certificate	ii
Declaration Certificate	iii
Acknowledgements	iv
List of Figures	v
List of Tables	v
Abstract	1
1. Introduction	
1.1 Machine Learning	1
• Logistic Regression Model	
• Decision Tree Model	
• Random Forest Model	
• K-Fold Cross-Validation	
1.2 Cloud Storage Services	
• Cloud Storage	
• Cloud SQL	
• Cloud Spanner	
• Cloud Firestore	
• Cloud Bigtable.	
2. Experimental work	5
• Architecture	
• Dataset and Attribute information	
• Techniques used	
3. Results	11
4. Conclusion	12
5. References	12
6. Source Code	13
7. Plagiarism Report	16

ABSTRACT

The objective of this project is to match and identify an accurate data processing model to predict the occurrence of carcinoma supported patients clinical records by using Machine Learning model for pre-processing and prediction. Data processing models used are Logistic Regression, Decision Tree and Random Forest. to evaluate the performance of those models, the test data set used is Wisconsin Diagnostic breast cancer (1995). Cross-validation method is implemented to estimate the test error of every model. The simplest method is obtained using classification accuracy, which was obtained by comparing actual to predicted values. To manage the vast dataset and improve the prediction, we use Google Cloud Services like Cloud Shell and Cloud Storage. It's expected that with the assistance of the model identified, physicians and patients can enjoy the feature recognition outcome to predict carcinoma in early stages.

1.INTRODUCTION

Breast cancer represents the second primary cause of cancer deaths in women today and has become one among the foremost common cancers among women both within the developed and therefore the developing world within the last years. Carcinoma may be a progressive disease that depends on when the patient is aware of it, that the treatment begins deciding the extent of which the patient is affected and therefore the likelihood of the patient overcoming it. Research focused on 187 countries about carcinoma mortality and incidences from 1980 to 2010 shows that global carcinoma incidence increased from 641,000 cases in 1980 to 1,643,000 cases in 2010, with an annual increase rate of three 3.1%. The newest, advanced screening methods are helping to diagnose the disease when it's still at a localized stage. To assist the prediction of carcinoma, many researchers and students are using various Machine Learning and AI techniques.

Breast cancer develops within the breast cells of an individual. Lobules are the milk-producing glands, and ducts are the channels that carry milk from the glands into the nipple. Cancer also can develop inside your breast within the adipose tissue or within the fibrous connective tissue. Sometimes the uncontrolled cancer cells invade other healthy breast tissue and may visit the lymph nodes under the arms. The lymph nodes are a key means of transferring cancer cells to other areas of the body. Therefore the cancer cells can form within the breast and spread to other parts of the body and grow into tumours.

1

In this project, we are predicting breast cancer supported three data processing models: Decision Tree model, Logistic Regression model and therefore the Random Forest model.

1.1 LOGISTIC REGRESSION

Logistic Regression is a Supervised Learning model which is used for predicting discrete-valued outputs.

1.2 DECISION TREE

Decision tree provides a strong technique for classification and prediction in Breast Cancer diagnosis problem. Various decision tree algorithms are available to classify the data, including ID3, C4.5, C5, J48, CART and CHAID. In this paper we have chosen the J48 decision tree algorithm to establish the model.

1.3 RANDOM FOREST

Random forest is one of many classification techniques, and it is an algorithm for big data classification. Random forest classification is applied to cancer microarray data to achieve a more accurate and reliable classification performance.

1.4 CROSS VALIDATION

Test error is the most important measure for evaluating classification output in any classification problem. A variety of statistical techniques are proposed to estimate test error, in the absence of a very large test range. K-fold cross-validation for calculating the efficiency of each model is applied here. This approach begins by randomly dividing the collection of data into equally sized groups. Each fold is viewed as a test set in the cycle and the remaining groups are known as the training set. The test error is determined for every iteration, and thus the model with the highest accuracy is found.

Compute Engine and VM Instance

VMs (Virtual Machines) are the compute engine services provided by GCP. A VM is similar to a hardware computer. VMs consist of a virtual CPU, some amount of memory, disk storage, and an IP address. Compute Engine is GCP's service to create VMs; it is very flexible. For example, a micro VM shares a CPU with other virtual machines, so you can get a VM with less capacity at a lower cost. Another example of a function that can't exist in hardware is that some VMs offer burst capability, meaning that the virtual CPU will run above its rated capacity for a brief period,

2

using the available shared physical CPU. The main VM options are CPUs, memory, disks, and networking.

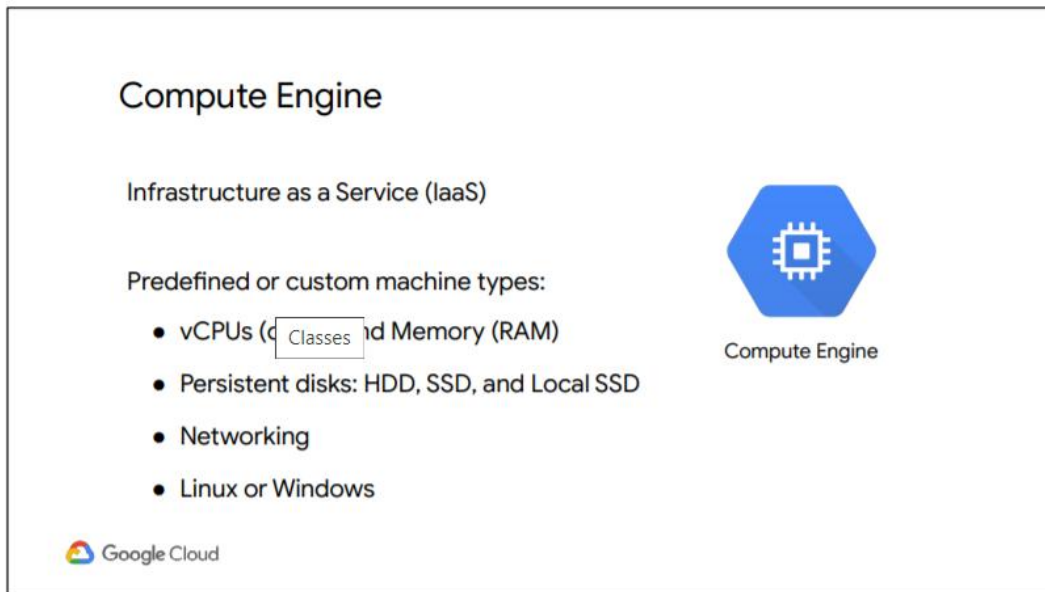


Fig.1:Compute Engine

Cloud Storage Services

OverView

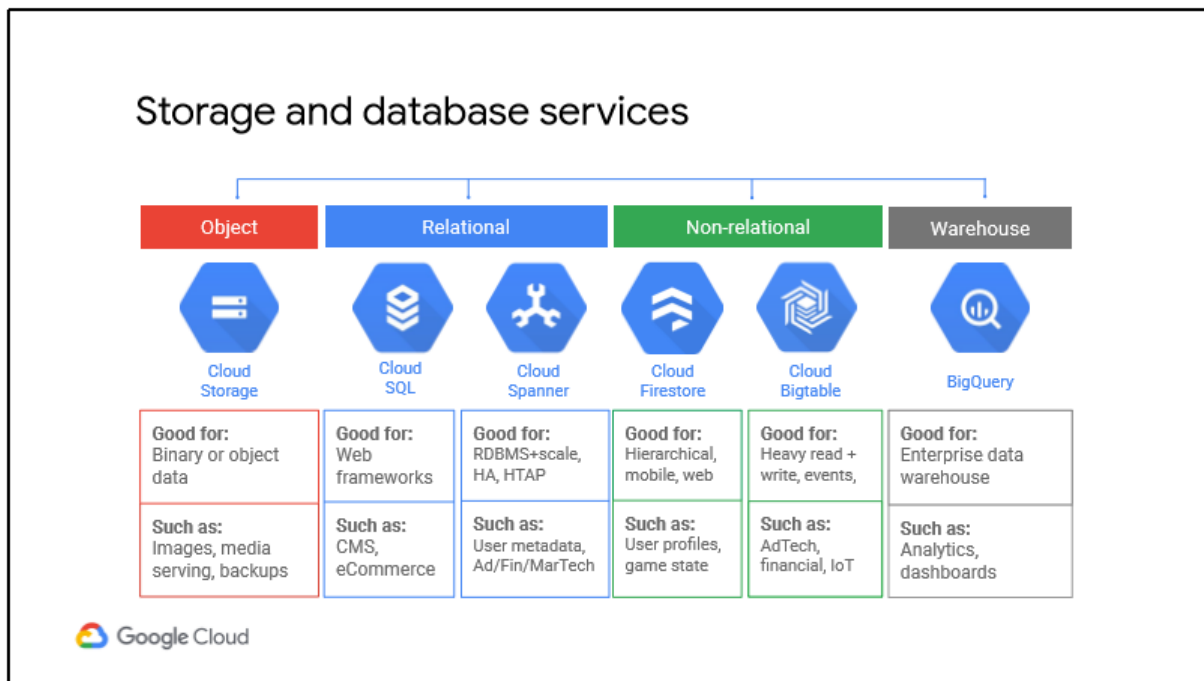


Fig.2 : Storage and Database Services

1. Cloud Storage

Cloud Storage is an object storage service of GCP, and it allows storage and retrieval of data. Key features of google cloud storage are:

- Scalable to exabytes of data
- The time to first byte is very low.
- Very high availability across all storage classes
- And It has a single API across those storage classes

Cloud Storage is not similar to a file system. Instead, Cloud Storage is a collection of buckets that you place objects into .

2. Cloud SQL

Cloud SQL is a fully managed service of MySQL or PostgreSQL databases. The updates are automatically applied but still you have to administer MySQL users with the native authentication tools that come with these databases. Cloud SQL supports many clients such as cloud shell, command line, Gsuite..etc

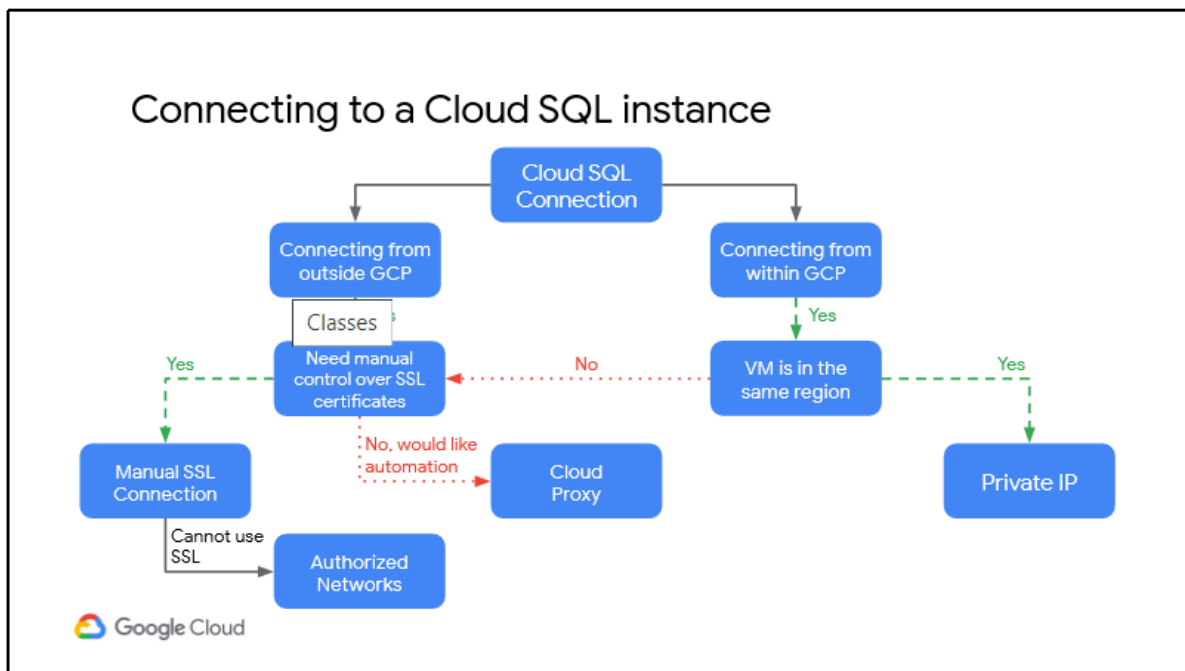


Fig. 3: Connecting to a Cloud SQL Instance

3. Cloud BigTable

Cloud Bigtable is a fully managed NoSQL database with petabyte-scale and very low latency. It scales for throughput and it adjusts itself to the access patterns. Cloud Bigtable is a great choice for both operational and analytical applications, including IoT, user analytics, and financial data

analysis, because it supports high read and write throughput at low latency. It's also a great storage engine for machine learning applications.

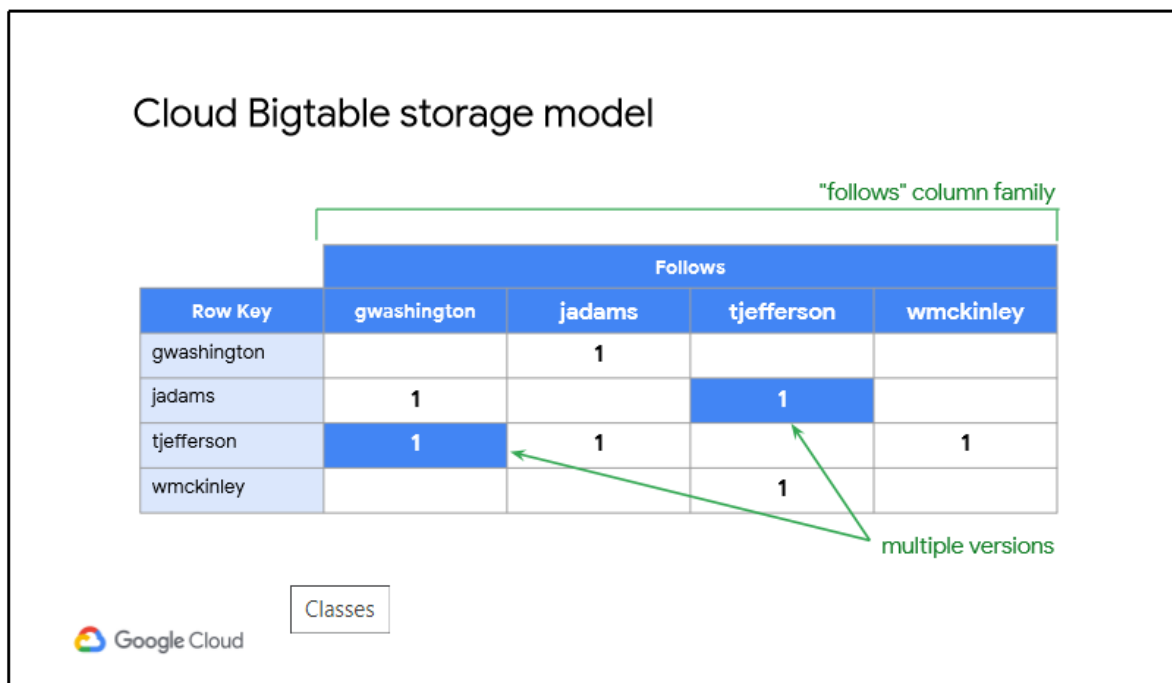


Fig. 4: Cloud Bigtable Storage Model

2. EXPERIMENTAL WORK

2.1 ARCHITECTURE

The training dataset is given to all the three data mining models and the data is preprocessed to remove unnecessary data and the relevant columns are taken. After training the data models, the test data is given and from that the accuracy is determined and the cross validation of all the data models are computed to find the model which can predict breast cancer better.

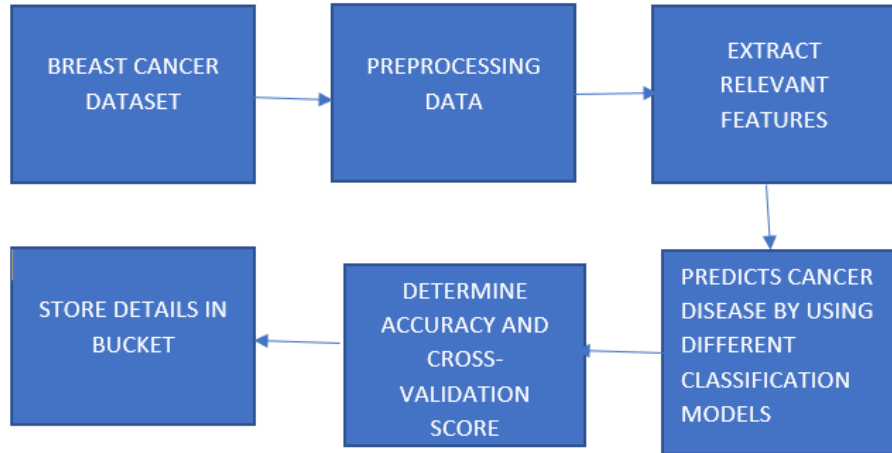


Fig. 5a: Process Flow Chart

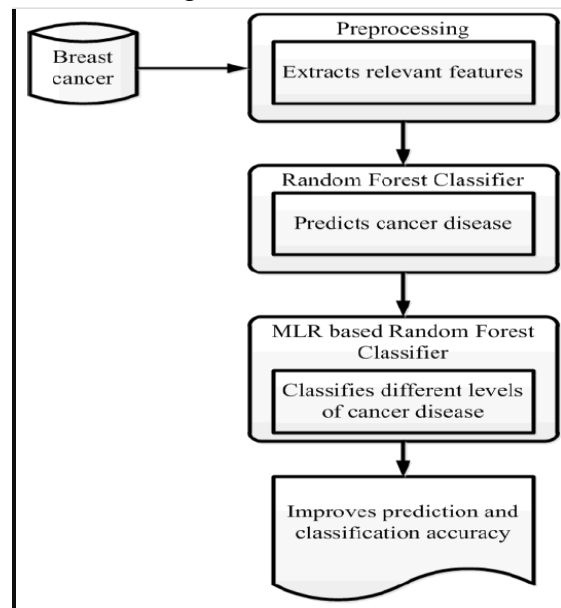


Fig. 5b: Process Flow Chart

2.2 DATA SET AND ATTRIBUTE INFORMATION

Here in this experiment we use the data set which has the following schema :

- 1) ID number
- 2) Diagnosis (M = malignant, B = benign)

Ten real-valued features are computed for each cell nucleus:

- a) radius (mean of distances from center to points on the perimeter)

- b) texture (standard deviation of gray-scale values)
- c) perimeter
- d) area
- e) smoothness (local variation in radius lengths)
- f) compactness ($\text{perimeter}^2 / \text{area} - 1.0$)
- g). concavity (severity of concave portions of the contour)
- h). concave points (number of concave portions of the contour)
- i). Symmetry
- j). fractal dimension ("coastline approximation" - 1)

2.3 Techniques used

In this experiment we used the machine learning algorithms and also used google cloud services in the prediction of breast cancer.

A. Machine learning

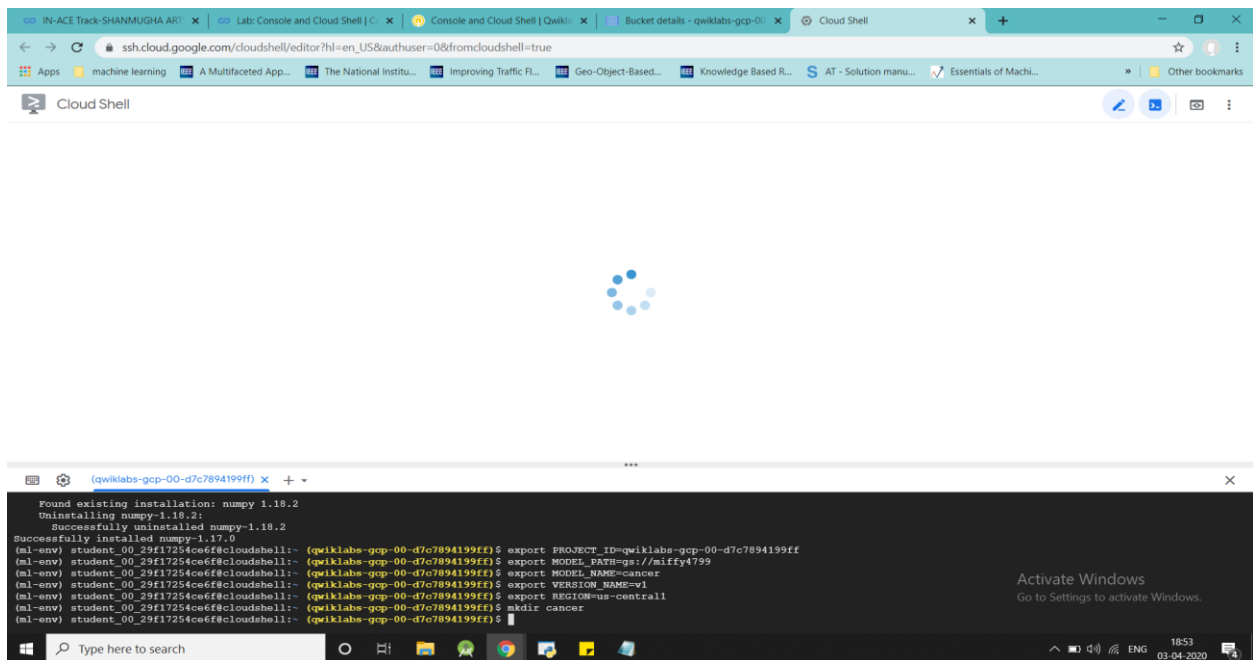
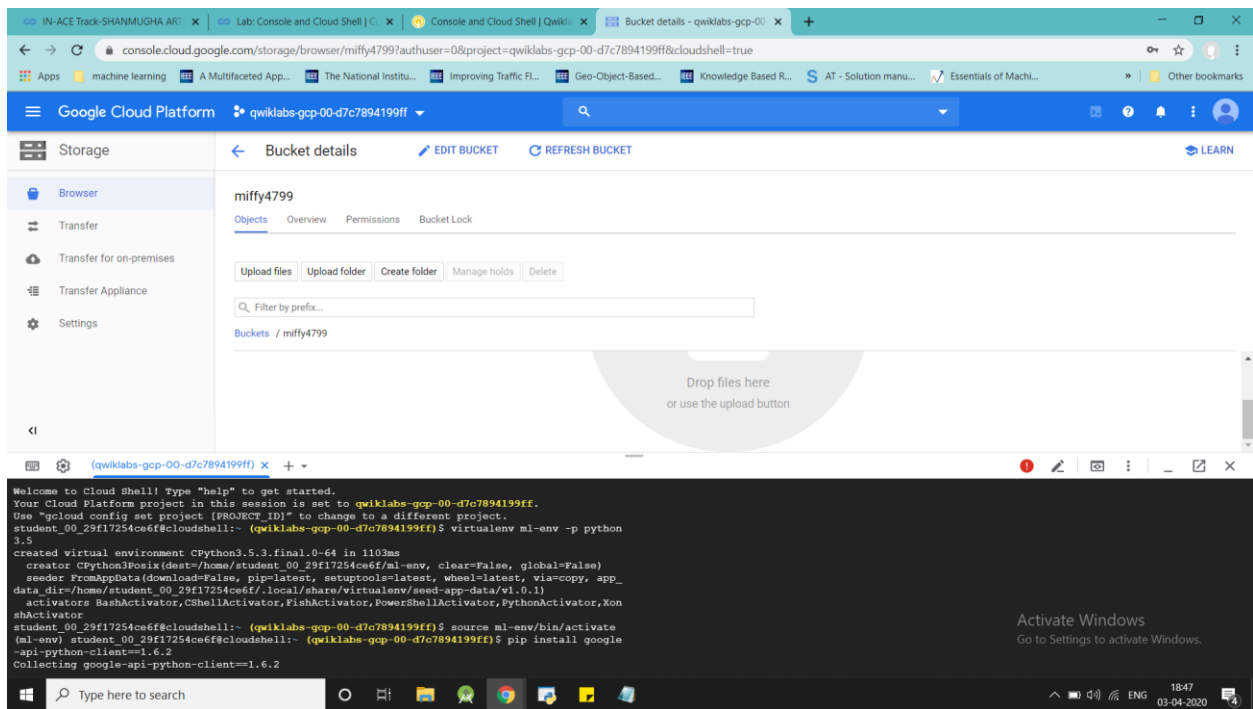
- 1.Clean and Prepare dataset
- 2.Create a test set and train set
- 3.Using Model Classification Algorithms(Logistic Regression,Decision Tree,Random Forest)
- 4.Predicting the best Model using Evaluation Metrics like(Accuracy and Cross-Validation Values)

B. Cloud Services

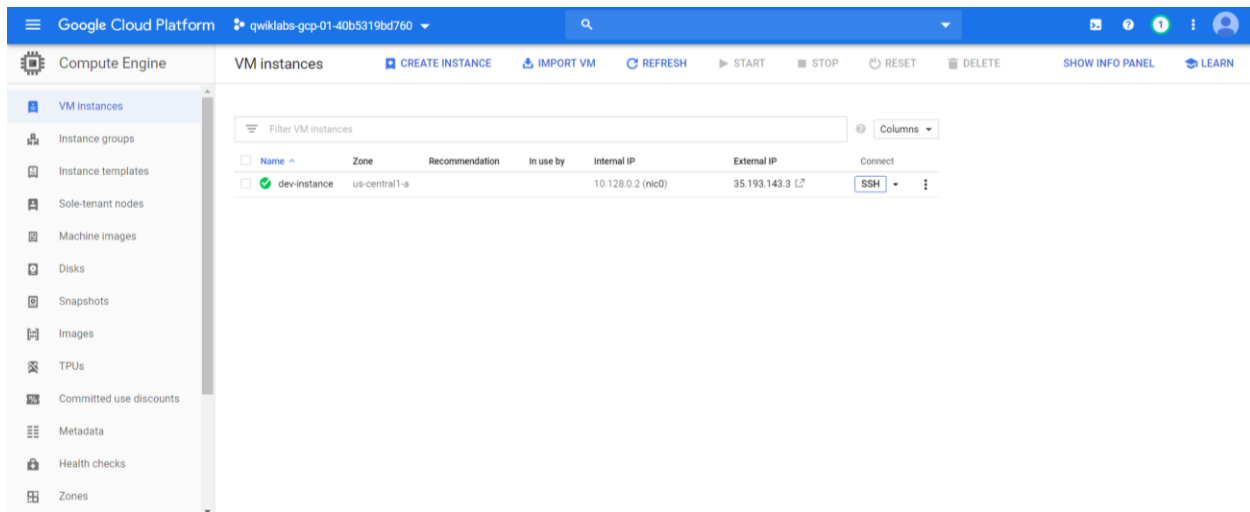
Steps involved:

- 1.Cloud Storage Buckets to store the trained model and test model
- 2.Create isolated Python environment and Activate virtualenv in Cloud shell
- 3.Upload the DataSet to the Cloud Shell
- 4.Setup the Environment Variables
- 5.Open the Cloud Shell Editor and paste the classification model into the editor and save as .py file
- 6.Run the file and verify the results
- 7.Store the Classification Model into cloud storage bucket

1.Creating the bucket



2.Creating a VM instance



3.Setting Python Environment using VM instance

```
student-01-8e1dc2fc339@dev-instance: ~ - Google Chrome
ssh.cloud.google.com/projects/qwiklabs-gcp-01-40b5319bd760/zones/us-central1-a/instances/dev-instance?authuser=1&hl=en_US&projectNumber=824834228820

Setting up rename (0.20-4) ...
update-alternatives: using /usr/bin/file-rename to provide /usr/bin/rename (rename) in auto mode
Setting up git (1:2.11.0-3+deb9u5) ...
student-01-8e1dc2fc339@dev-instance:~$ sudo apt-get install python3-setuptools python3-dev build-essential
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  binutils bzip2 cpp cpp-6 dpkg-dev fakeroot g++ g++-6 gcc gcc-6 libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan3 libatomic1 libc-dev-bin libc6-dev libc6i-0
  libcilkrts5 libdpkg-perl libexpat1-dev libfakeroot libfile-fcntllock-perl libgcc-6-dev libgomp1 libisl15 libitm1 liblsan0 libmpc3 libmpfr4 libmpx2 libpython3-dev libpython3.5 libpython3.5-dev
  libquadmath0 libstdc++-6-dev libtsan0 libubsan0 linux-libc-dev make manpages manpages-dev python3.5-dev
Suggested packages:
  binutils-doc bzip2-doc cpp-doc gcc-6-locales debian-keyring g++-multilib g++-6-multilib gcc-6-doc libstdc++6-6-dbg gcc-multilib autoconf automake libtool flex bison gdb gcc-doc gcc-6-multilib
  libgcc1-dbg libgomp1-dbg libitm1-dbg libatomic1-dbg libasan3-dbg liblsan0-dbg libtsan0-dbg libubsan0-dbg libcilkrts5-dbg libmpx2-dbg libquadmath0-dbg glibc-doc libstdc++-6-doc make-doc
  python3-setuptools-doc
The following NEW packages will be installed:
  binutils build-essential bzip2 cpp cpp-6 dpkg-dev fakeroot g++ g++-6 gcc gcc-6 libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan3 libatomic1 libc-dev-bin
  libc6-dev libc6i-0 libcilkrts5 libdpkg-perl libexpat1-dev libfakeroot libfile-fcntllock-perl libgcc-6-dev libgomp1 libisl15 libitm1 liblsan0 libmpc3 libmpfr4 libmpx2 libpython3-dev
  libpython3.5 libpython3.5-dev libquadmath0 libstdc++-6-dev libtsan0 libubsan0 linux-libc-dev make manpages manpages-dev python3-dev python3-setuptools python3.5-dev
0 upgraded, 46 newly installed, 0 to remove and 1 not upgraded.
Need to get 81.2 MB of archives.
After this operation, 227 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://deb.debian.org/debian stretch/main amd64 bzip2 amd64 1.0.6-8-1 [47.5 kB]
Get:2 http://deb.debian.org/debian stretch/main amd64 manpages all 4.10-2 [1,222 kB]
Get:3 http://deb.debian.org/debian stretch/main amd64 binutils amd64 2.28-5 [3,770 kB]
Get:4 http://deb.debian.org/debian stretch/main amd64 libc-dev-bin amd64 2.24-11+deb9u4 [259 kB]
Get:5 http://deb.debian.org/debian stretch/main amd64 linux-libc-dev amd64 4.9.210-1 [1,481 kB]
Get:6 http://deb.debian.org/debian stretch/main amd64 libc6-dev amd64 2.24-11+deb9u4 [2,364 kB]
Get:7 http://deb.debian.org/debian stretch/main amd64 libisl15 amd64 0.18-1 [564 kB]
Get:8 http://deb.debian.org/debian stretch/main amd64 libmpfr4 amd64 3.1.5-1 [556 kB]
Get:9 http://deb.debian.org/debian stretch/main amd64 libmpc3 amd64 1.0.3-1+b2 [39.9 kB]
Get:10 http://deb.debian.org/debian stretch/main amd64 cpp-6 amd64 6.3.0-18+deb9u1 [6,584 kB]
Get:11 http://deb.debian.org/debian stretch/main amd64 cpp amd64 4:6.3.0-4 [18.7 kB]
Get:12 http://deb.debian.org/debian stretch/main amd64 libc6i-0 amd64 6.3.0-18+deb9u1 [30.6 kB]
Get:13 http://deb.debian.org/debian stretch/main amd64 libgomp1 amd64 6.3.0-18+deb9u1 [73.3 kB]
Get:14 http://deb.debian.org/debian stretch/main amd64 libitm1 amd64 6.3.0-18+deb9u1 [27.3 kB]
Get:15 http://deb.debian.org/debian stretch/main amd64 libatomic1 amd64 6.3.0-18+deb9u1 [6,966 B]
Get:16 http://deb.debian.org/debian stretch/main amd64 libasan3 amd64 6.3.0-18+deb9u1 [311 kB]
Get:17 http://deb.debian.org/debian stretch/main amd64 liblsan0 amd64 6.3.0-18+deb9u1 [115 kB]
Get:18 http://deb.debian.org/debian stretch/main amd64 libtsan0 amd64 6.3.0-18+deb9u1 [257 kB]
Get:19 http://deb.debian.org/debian stretch/main amd64 libubsan0 amd64 6.3.0-18+deb9u1 [107 kB]
Get:20 http://deb.debian.org/debian stretch/main amd64 libcilkrts5 amd64 6.3.0-18+deb9u1 [40.5 kB]
Get:21 http://deb.debian.org/debian stretch/main amd64 libmpx2 amd64 6.3.0-18+deb9u1 [11.2 kB]
Get:22 http://deb.debian.org/debian stretch/main amd64 libquadmath0 amd64 6.3.0-18+deb9u1 [131 kB]
Get:23 http://deb.debian.org/debian stretch/main amd64 libgcc-6-dev amd64 6.3.0-18+deb9u1 [2,296 kB]
Get:24 http://deb.debian.org/debian stretch/main amd64 gcc-6 amd64 6.3.0-18+deb9u1 [6,900 kB]
Get:25 http://deb.debian.org/debian stretch/main amd64 gcc amd64 4:6.3.0-4 [5,196 B]
Get:26 http://deb.debian.org/debian stretch/main amd64 libstdc++-6-dev amd64 6.3.0-18+deb9u1 [1,420 kB]
Get:27 http://deb.debian.org/debian stretch/main amd64 g++-6 amd64 6.3.0-18+deb9u1 [7,094 kB]
```

```
student-01-8e1dc2fc339@dev-instance: ~/training-data-analyst/courses/developingapps/python/devenv - Google Chrome
ssh.cloud.google.com/projects/qwiklabs-gcp-01-40b5319bd760/zones/us-central1-a/instances/dev-instance?authuser=1&hl=en_US&projectNumber=824834228820

% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 1764k 100 1764k 0 0 9098k 0 --:--:-- --:--:-- --:--:-- 9144k
student-01-8e1dc2fc339@dev-instance:~$ sudo python3 get-pip.py
Collecting pip
  Downloading pip-20.0.2-py2.py3-none-any.whl (1.4 MB)
    | 1.4 MB 3.4 MB/s
Collecting wheel
  Downloading wheel-0.34.2-py2.py3-none-any.whl (26 kB)
Installing collected packages: pip, wheel
Successfully installed pip-20.0.2 wheel-0.34.2
student-01-8e1dc2fc339@dev-instance:~$ python3 --version
Python 3.5.3
student-01-8e1dc2fc339@dev-instance:~$ pip3 --version
pip 20.0.2 from /usr/local/lib/python3.5/dist-packages/pip (python 3.5)
student-01-8e1dc2fc339@dev-instance:~$ git clone https://github.com/GoogleCloudPlatform/training-data-analyst
Cloning into 'training-data-analyst'...
remote: Enumerating objects: 150, done.
remote: Counting objects: 100% (150/150), done.
remote: Compressing objects: 100% (133/133), done.
remote: Total 32898 (delta 98), reused 45 (delta 24), pack-reused 32740
Receiving objects: 100% (32898/32898), 374.77 MiB | 39.07 MiB/s, done.
Resolving deltas: 100% (19979/19979), done.
cd ~/training-data-analyst/courses/developingapps/python/devenv/
student-01-8e1dc2fc339@dev-instance:~/training-data-analyst/courses/developingapps/python/devenv$ sudo python3 server.py
Server is starting...
Started: Press Ctrl + C to stop
103.49.121.170 - - [04/Apr/2020 05:13:46] "GET / HTTP/1.1" 200 -
103.49.121.170 - - [04/Apr/2020 05:13:47] "GET /favicon.ico HTTP/1.1" 200 -
C-----
Exception happened during processing of request from ('103.49.121.170', 64767)
Traceback (most recent call last):
  File "/usr/lib/python3.5/socketserver.py", line 313, in _handle_request_noblock
    self.process_request(request, client_address)
  File "/usr/lib/python3.5/socketserver.py", line 341, in process_request
    self.finish_request(request, client_address)
  File "/usr/lib/python3.5/socketserver.py", line 354, in finish_request
    self.RequestHandlerClass(request, client_address, self)
  File "/usr/lib/python3.5/socketserver.py", line 681, in __init__
    self.handle()
  File "/usr/lib/python3.5/http/server.py", line 422, in handle
    self.handle_one_request()
  File "/usr/lib/python3.5/http/server.py", line 390, in handle_one_request
    self.raw_requestline = self.rfile.readline(65537)
  File "/usr/lib/python3.5/socket.py", line 576, in readinto
    return self._sock.recv_into(b)
KeyboardInterrupt
-----
35.193.143.3 - - [04/Apr/2020 05:14:08] "GET / HTTP/1.1" 200 -
```

4. Running a Python File in the cloud shell editor

```
IN-ACE x Lab Co x Console x Bucket x Cloud S x https:// x IN-ACE x Lab Co x Cloud S x Bucket x Cloud S x Machin x +
ssh.cloud.google.com/cloudshell/editor?hl=en_US&fromcloudshell=true
Apps machine learning A Multifaceted App... The National Institu... Improving Traffic RL... Geo-Object-Based... Knowledge Based R... AT - Solution manu... Essentials of Machi... Other bookmarks
Cloud Shell
File Edit Selection View Go Help
EXPLORER STUDENT_0... myfile.py x
data
ml-env
data.csv
myfile.py
README-cloudshell.txt
error = []
for train, test in kf:
    # Filter training data
    train_predictors = (data[predictors].iloc[train,:])

    # The target we're using to train the algorithm.
    train_target = data[outcome].iloc[train]

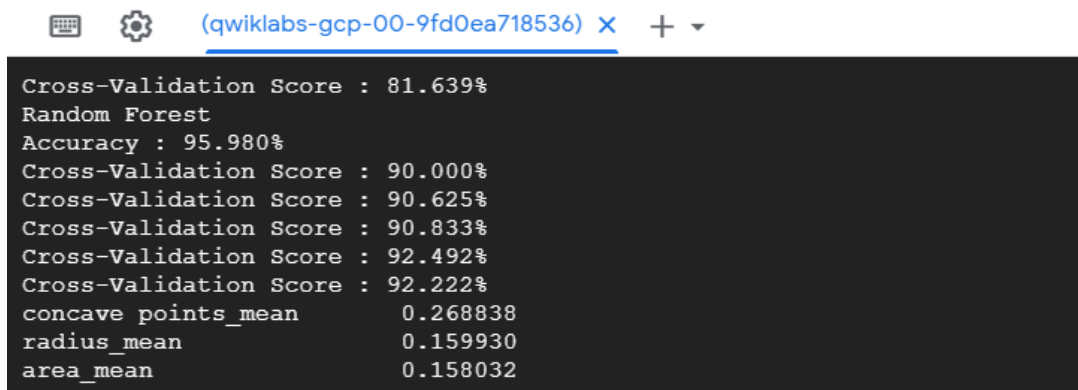
    # Training the algorithm using the predictors and target.
    model.fit(train_predictors, train_target)

    #Record error from each cross-validation run
    error.append(model.score(data[predictors].iloc[test,:], data[outcome].iloc[test]))

E305 expected 2 blank lines after class or function definition, found 0 pycodestyle(E305)
E231 missing whitespace after ',' pycodestyle(E231)
Peek Problem No quick fixes available
predictor_var = ['radius_mean', 'perimeter_mean', 'area_mean', 'compactness_mean', 'concave points_mean']
102
(qwiklabs-gcp-00-9fd0ea718536) x +
Cross-Validation Score : 82.8574
Cross-Validation Score : 88.4874
Cross-Validation Score : 89.3844
Cross-Validation Score : 90.5674
(ml-env) student_00_9a4aef2703768@cloudshell:~ (qwiklabs-gcp-00-9fd0ea718536) $ gsuutil cp ./model.joblib gs://xox4799/model.joblib
CommandException: No URLs matched: ./model.joblib
(ml-env) student_00_9a4aef2703768@cloudshell:~ (qwiklabs-gcp-00-9fd0ea718536) $ gsuutil cp ./myfile.py gs://xox4799/myfile.py
Copying file://./myfile.py (Content-Type=text/x-python)
/ [1 files] 3.5 KiB / 3.5 KiB
Operation completed over 1 objects/3.5 KiB.
(ml-env) student_00_9a4aef2703768@cloudshell:~ (qwiklabs-gcp-00-9fd0ea718536) $
Activate Windows
Go to Settings to activate Windows.
```

3. RESULTS

After doing an accuracy check and cross-validation score, we can choose the best model to use for the breast cancer prediction by its performance. In the Logistic Regression model, we use it for binary classification. The accuracy of the predictions are good but not great. The cross-validation scores are reasonable. In the Decision Tree Model, the accuracy of the prediction is much better here but the cross-validation score is not that great. Using all the features improves the prediction accuracy and the cross-validation score is great. But the advantage with Random Forest is that it returns a feature importance matrix which can be used to select features. Using all the features improves the prediction accuracy and the cross-validation score is great. An advantage with Random Forest is that it returns a feature importance matrix which can be used to select features.



```

Cross-Validation Score : 81.639%
Random Forest
Accuracy : 95.980%
Cross-Validation Score : 90.000%
Cross-Validation Score : 90.625%
Cross-Validation Score : 90.833%
Cross-Validation Score : 92.492%
Cross-Validation Score : 92.222%
concave points_mean      0.268838
radius_mean              0.159930
area_mean                0.158032

```

Fig. 10: Cross-Validation Score of Random Forest

Model	Accuracy (in percentage)
Logistic Regression	88.442
Decision Tree Model	100.00
Random Forest	94.724

Table.1 :Accuracy Results

4.CONCLUSION

There was a striking improvement in the accuracy of classification of women with and without breast cancer achieved with Machine learning by using different models such as Logistic Regression model, Decision Tree model and Random Forest .

Improvements in computational capacity and data management in healthcare systems can be followed by opportunities to exploit ML to enhance risk prediction of disease and survival prognosis in clinical practice.

5.REFERENCES

1. Breast Cancer Research, ML techniques for personalised breast cancer risk prediction
<https://breast-cancer-research.biomedcentral.com/articles>
2. Breast Cancer Research Results and Study updates, National Cancer Institute,
<https://www.cancer.gov/types/breast/research/articles>
3. Breast Cancer Prediction, <https://www.kaggle.com/buddhiniw/breast-cancer-prediction>
4. Predicting breast cancer risk using personal health data and ML models,
<https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0226765>
5. Forouzanfar, M. H., Foreman, K. J., Delossantos, A. M., Lozano, R., Lopez, A. D., Murray, C. J., and Naghavi, M., 2011, "Breast and Cervical Cancer in 187 Countries between 1980 and 2010: A Systematic Analysis," *The Lancet*, 378(9801), 1461-1484.
6. A Survey on Breast Cancer Prediction Using Data Mining Techniques
<https://ieeexplore.ieee.org/document/8544268>
7. Mousavi SM, Montazeri A, Mohagheghi MA, Jarrahi AM, Harirchi I, Najafi M, Ebrahimi M. Breast cancer in Iran: an epidemiological review. *The breast journal*. 2007 Jul;13(4):383-91.