NEMO
Manual

Table of contents

Part I: Preparing to Use NEMO

1. Environment Prerequisites

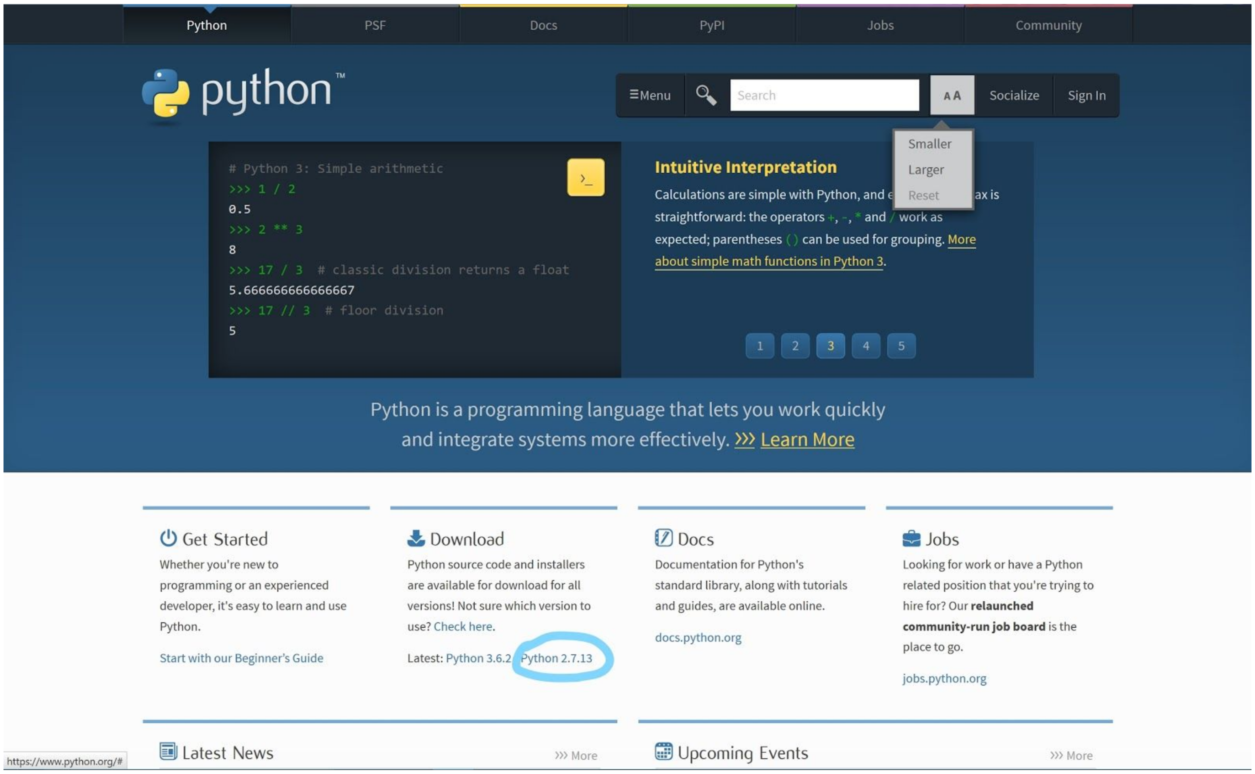
A. Introduction

NEMO is built using OS-independent languages and tools such as Python, MySQL Server, and

B. Python v 2.7.12 or above

Python version 2.7.12 or above is required to run NEMO.

To install, visit www.python.org which will take you to the webpage referenced in Figure 1.1. In the ‘Download’ section, click ‘Python 2.7.X’. This will download the version of Python with which NEMO is compatible.



```
# Python 3: Simple arithmetic
>>> 1 / 2
0.5
>>> 2 ** 3
8
>>> 17 / 3  # classic division returns a float
5.666666666666667
>>> 17 // 3  # floor division
5
```

Intuitive Interpretation

Calculations are simple with Python, and e ax is straightforward: the operators +, -, * and / work as expected; parentheses () can be used for grouping. More about simple math functions in Python 3.

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> Learn More

Get Started

Whether you're new to programming or an experienced developer, it's easy to learn and use Python.

Start with our Beginner's Guide

Download

Python source code and installers are available for download for all versions! Not sure which version to use? Check here.

Latest: Python 3.6.2 Python 2.7.13

Docs

Documentation for Python's standard library, along with tutorials and guides, are available online.

docs.python.org

Jobs

Looking for work or have a Python related position that you're trying to hire for? Our **relaunched community-run job board** is the place to go.

jobs.python.org

Latest News >>> More

Upcoming Events >>> More

https://www.python.org/#

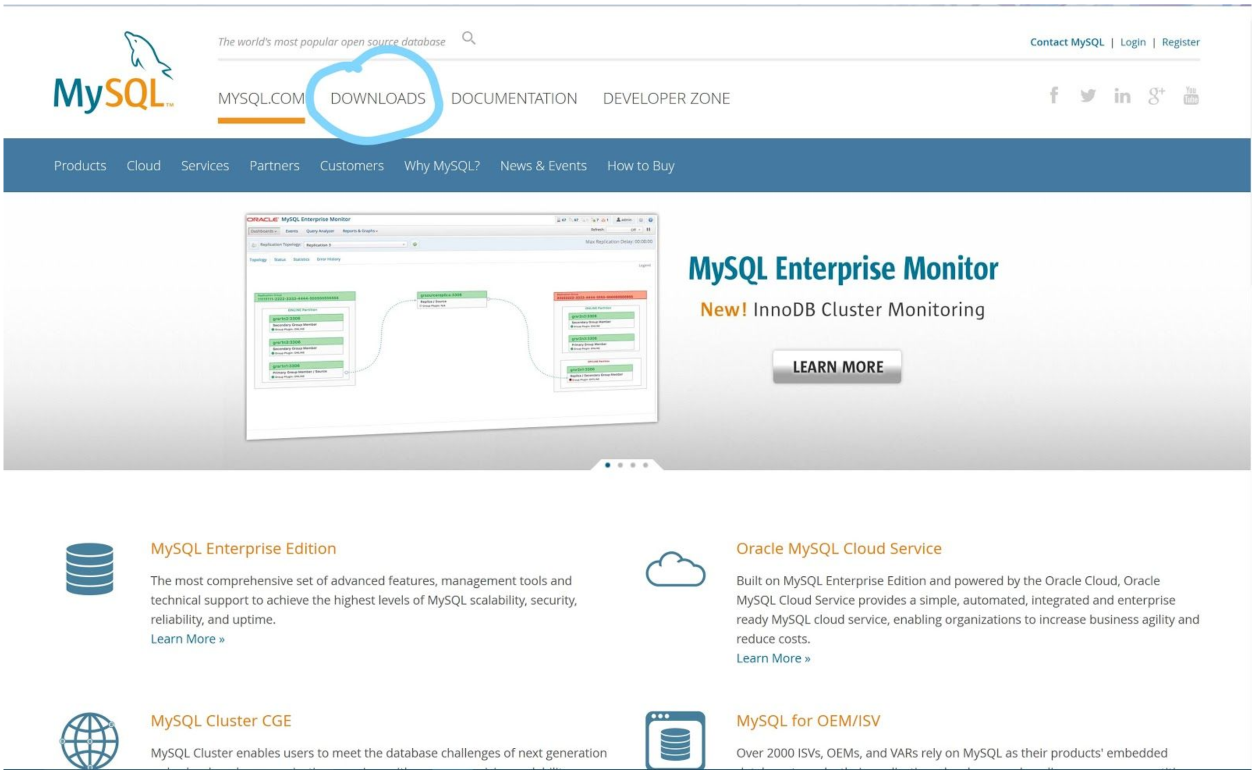
Figure 1.1

Complete the install instructions detailed by Python.

C. MySQL Server Version 5.7.18 or above

NEMO uses MySQL Server for persistent data storage.

For Unix systems, MySQL Server can be installed via the APT, YUM, or SUSE repositories. Navigate to the ‘Downloads’ page (Figure 1.2).



MySQL Enterprise Edition

The most comprehensive set of advanced features, management tools and technical support to achieve the highest levels of MySQL scalability, security, reliability, and uptime.
Learn More »

Oracle MySQL Cloud Service

Built on MySQL Enterprise Edition and powered by the Oracle Cloud, Oracle MySQL Cloud Service provides a simple, automated, integrated and enterprise ready MySQL cloud service, enabling organizations to increase business agility and reduce costs.
Learn More »

MySQL Cluster CGE

MySQL Cluster enables users to meet the database challenges of next generation

MySQL for OEM/ISV

Over 2000 ISVs, OEMs, and VARs rely on MySQL as their products' embedded

Figure 1.2

From there, navigate to the page corresponding to the your system's repository (Figure 1.3) and follow the appropriate instructions.

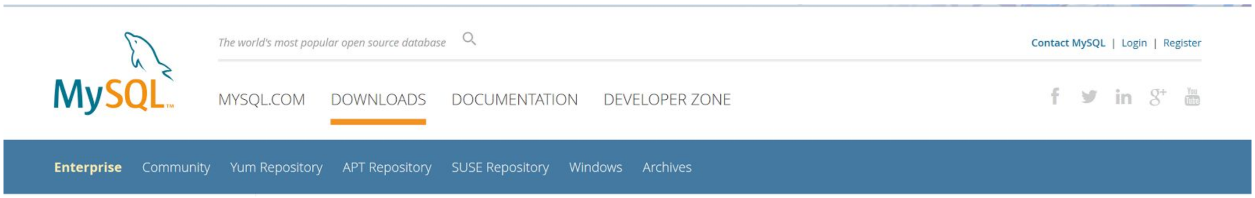
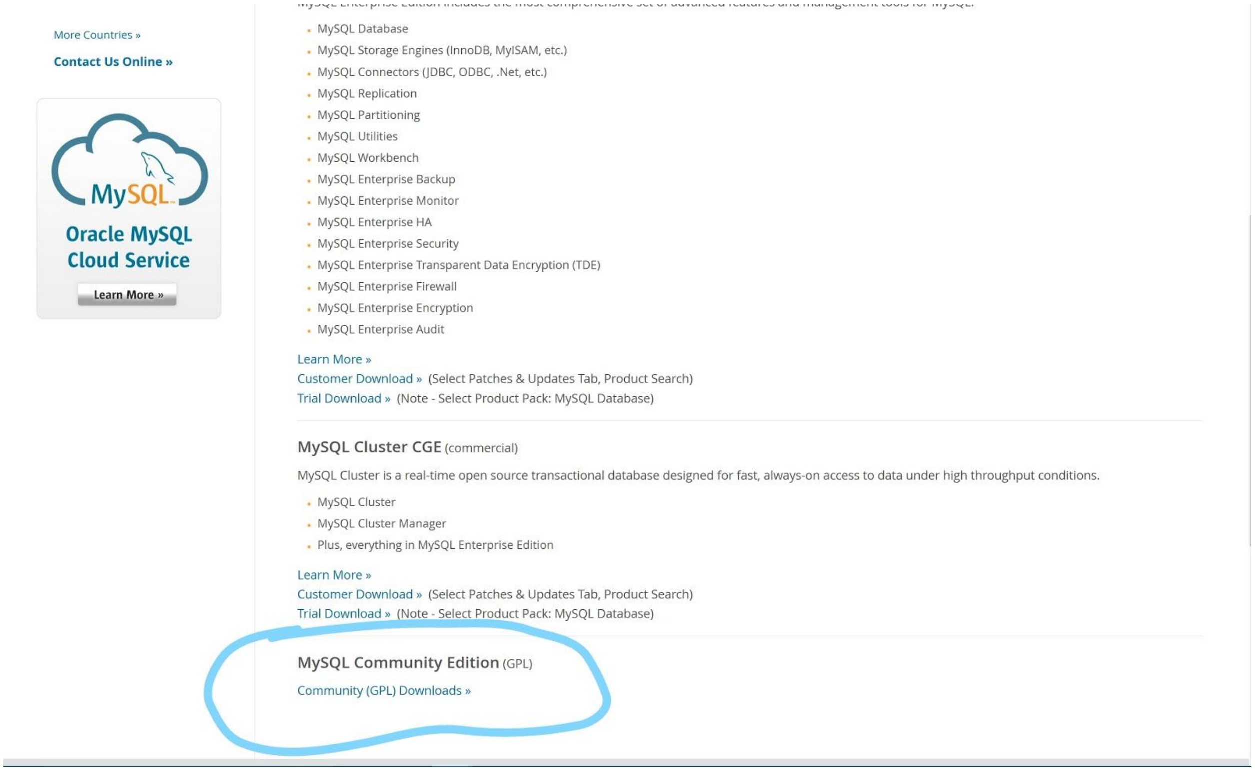


Figure 1.3

For Windows users, visit www.mysql.com. Navigate to the ‘Downloads’ page (Figure 1.2).

From there, navigate to the ‘MySQL Community Edition’ section and click on the ‘Community (GPL) Downloads’ link (Figure 1.4).



More Countries »

Contact Us Online »

MySQL Enterprise Edition includes the most comprehensive set of advanced features and management tools for MySQL.

- MySQL Database
- MySQL Storage Engines (InnoDB, MyISAM, etc.)
- MySQL Connectors (JDBC, ODBC, .Net, etc.)
- MySQL Replication
- MySQL Partitioning
- MySQL Utilities
- MySQL Workbench
- MySQL Enterprise Backup
- MySQL Enterprise Monitor
- MySQL Enterprise HA
- MySQL Enterprise Security
- MySQL Enterprise Transparent Data Encryption (TDE)
- MySQL Enterprise Firewall
- MySQL Enterprise Encryption
- MySQL Enterprise Audit

Learn More »
Customer Download » (Select Patches & Updates Tab, Product Search)
Trial Download » (Note - Select Product Pack: MySQL Database)

MySQL Cluster CGE (commercial)

MySQL Cluster is a real-time open source transactional database designed for fast, always-on access to data under high throughput conditions.

- MySQL Cluster
- MySQL Cluster Manager
- Plus, everything in MySQL Enterprise Edition

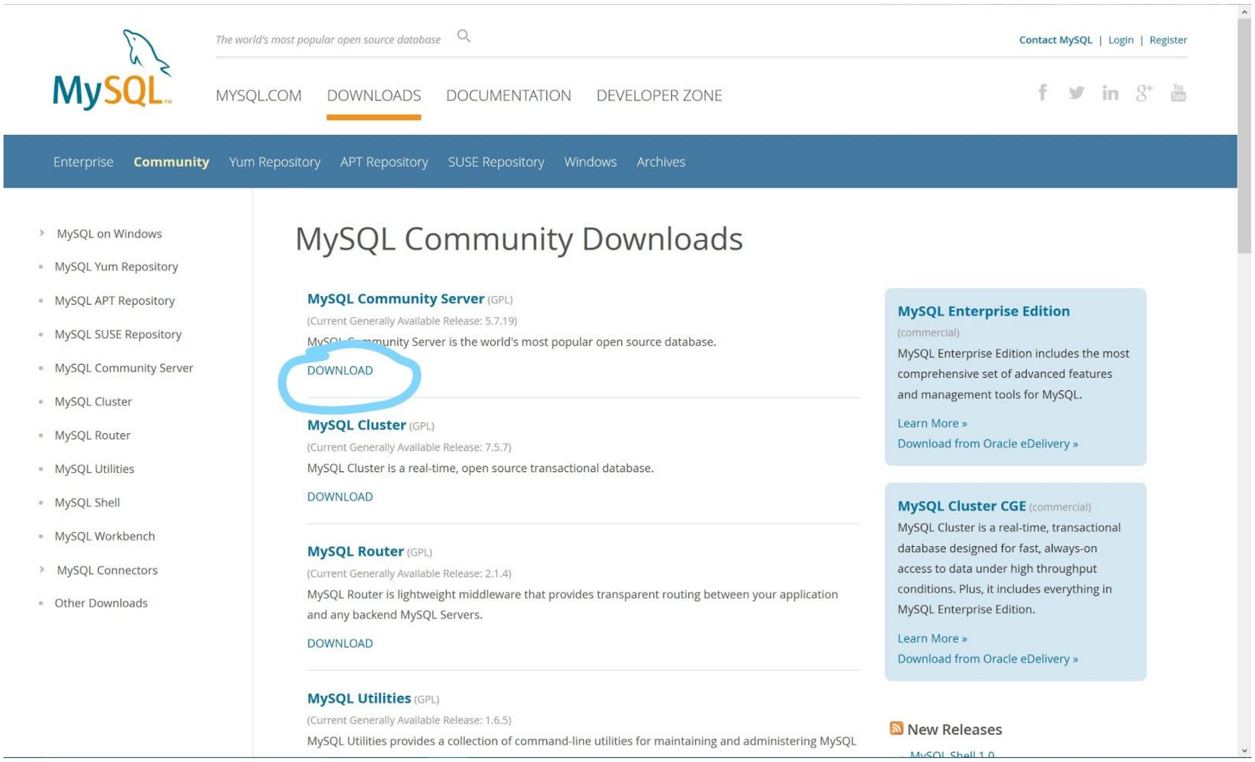
Learn More »
Customer Download » (Select Patches & Updates Tab, Product Search)
Trial Download » (Note - Select Product Pack: MySQL Database)

MySQL Community Edition (GPL)

Community (GPL) Downloads »

Figure 1.4

On this page, click ‘Download’ under ‘MySQL Community Server’ (Figure 1.5).



MySQL Community Downloads

MySQL Community Server (GPL)
(Current Generally Available Release: 5.7.19)
MySQL Community Server is the world's most popular open source database.
DOWNLOAD

MySQL Cluster (GPL)
(Current Generally Available Release: 7.5.7)
MySQL Cluster is a real-time, open source transactional database.
DOWNLOAD

MySQL Router (GPL)
(Current Generally Available Release: 2.1.4)
MySQL Router is lightweight middleware that provides transparent routing between your application and any backend MySQL Servers.
DOWNLOAD

MySQL Utilities (GPL)
(Current Generally Available Release: 1.6.5)
MySQL Utilities provides a collection of command-line utilities for maintaining and administering MySQL

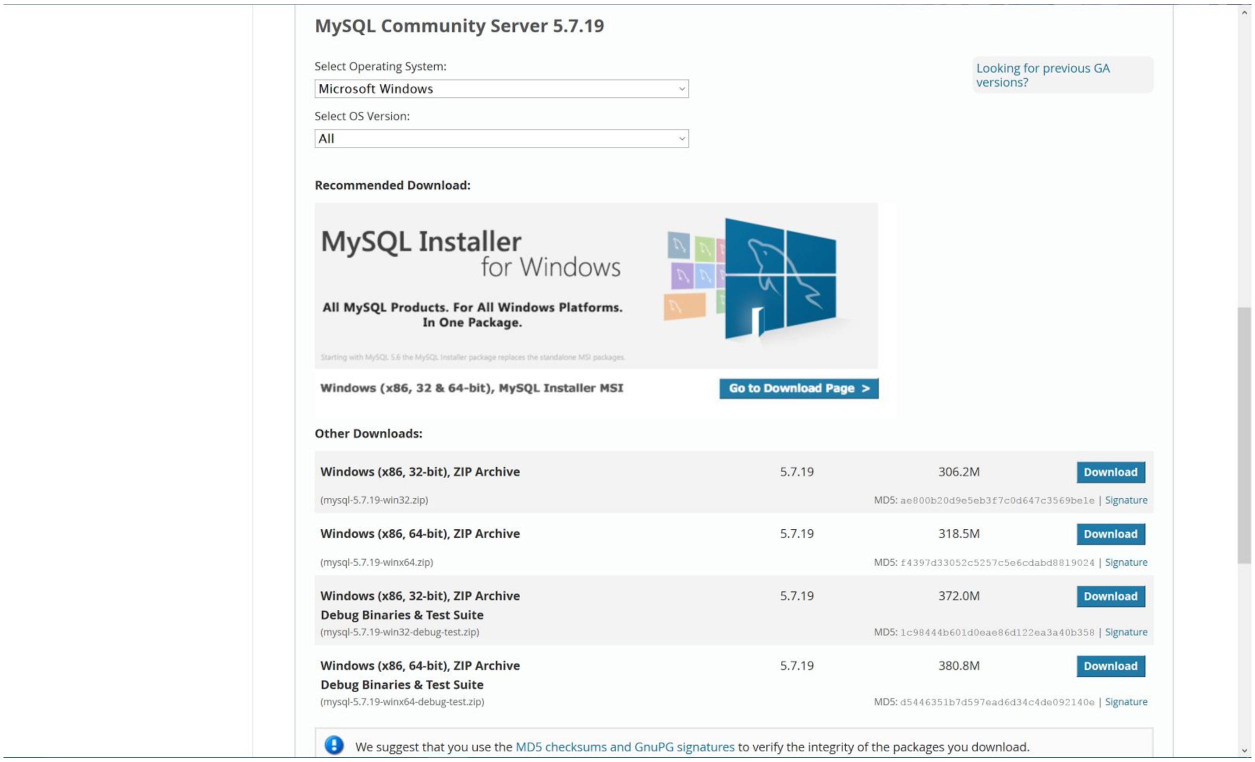
MySQL Enterprise Edition
(commercial)
MySQL Enterprise Edition includes the most comprehensive set of advanced features and management tools for MySQL.
Learn More »
Download from Oracle eDelivery »

MySQL Cluster CGE (commercial)
MySQL Cluster is a real-time, transactional database designed for fast, always-on access to data under high throughput conditions. Plus, it includes everything in MySQL Enterprise Edition.
Learn More »
Download from Oracle eDelivery »

New Releases
MySQL Shell 1.0

Figure 1.5

Download the appropriate file for your system (Figure 1.6). No account with MySQL is required for download.



MySQL Community Server 5.7.19

Select Operating System:
Microsoft Windows

Select OS Version:
All

Recommended Download:

MySQL Installer
for Windows
All MySQL Products. For All Windows Platforms. In One Package.
Starting with MySQL 5.6 the MySQL Installer package replaces the standalone MSI packages.

Windows (x86, 32 & 64-bit), MySQL Installer MSI

Other Downloads:

Windows (x86, 32-bit), ZIP Archive (mysql-5.7.19-win32.zip)	5.7.19	306.2M	Download MD5: ae800b20d9e5eb3f7c0d647c3569be1e | Signature
Windows (x86, 64-bit), ZIP Archive (mysql-5.7.19-winx64.zip)	5.7.19	318.5M	Download MD5: f4397d33052c5257c5e6cdabd8819024 | Signature
Windows (x86, 32-bit), ZIP Archive **Debug Binaries & Test Suite** (mysql-5.7.19-win32-debug-test.zip)	5.7.19	372.0M	Download MD5: 1c98444b601d0eae86d122ea3a40b358 | Signature
Windows (x86, 64-bit), ZIP Archive **Debug Binaries & Test Suite** (mysql-5.7.19-winx64-debug-test.zip)	5.7.19	380.8M	Download MD5: d5446351b7d597ead6d34c4de092140e | Signature

We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.

Figure 1.6

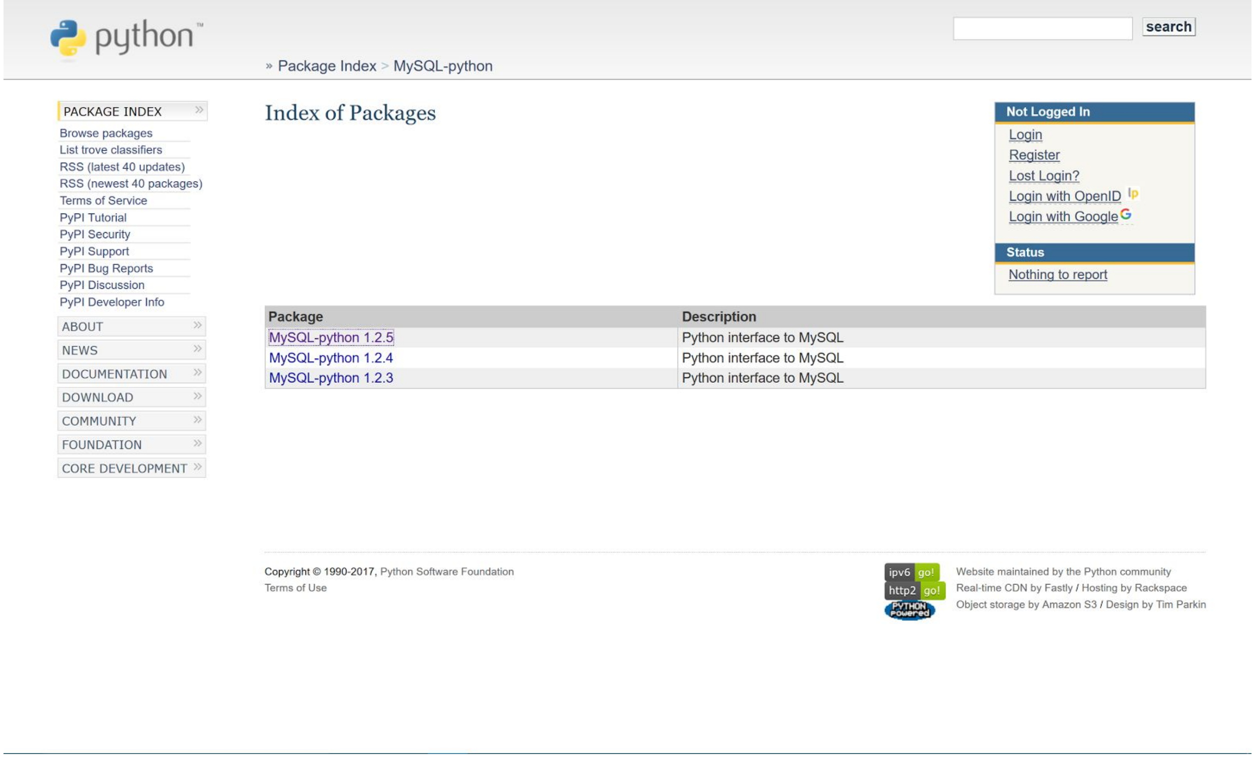
D. MySQL-Python Python Plugin

NEMO uses the MySQLdb Python interface to connect to the MySQL Server.

Linux Users: Install from Pip. Execute the following commands:

```
Pip install MySQL-Python
```

Windows Users: Visit the https://pypi.python.org/pypi/MySQL-python. Select version 1.2.5 (Figure 1.7).



Index of Packages

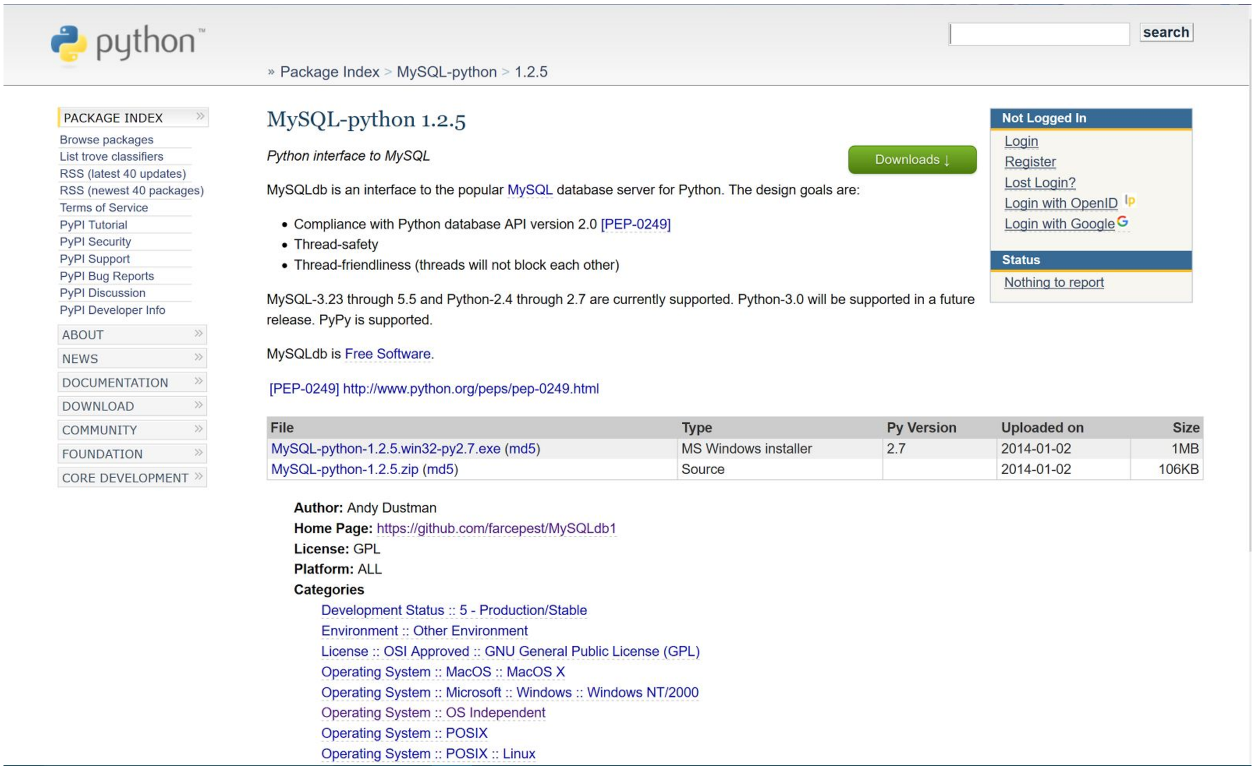
Package	Description
MySQL-python 1.2.5	Python interface to MySQL
MySQL-python 1.2.4	Python interface to MySQL
MySQL-python 1.2.3	Python interface to MySQL

Copyright © 1990-2017, Python Software Foundation
Terms of Use

Website maintained by the Python community
Real-time CDN by Fastly / Hosting by Rackspace
Object storage by Amazon S3 / Design by Tim Parkin

Figure 1.7

Install as instructed (Figure 1.8).



MySQL-python 1.2.5

Python interface to MySQL

MySQLdb is an interface to the popular MySQL database server for Python. The design goals are:

- Compliance with Python database API version 2.0 [PEP-0249]
- Thread-safety
- Thread-friendliness (threads will not block each other)

MySQL-3.23 through 5.5 and Python-2.4 through 2.7 are currently supported. Python-3.0 will be supported in a future release. PyPy is supported.

MySQLdb is Free Software.

[PEP-0249] http://www.python.org/peps/pep-0249.html

File	Type	Py Version	Uploaded on	Size
MySQL-python-1.2.5.win32-py2.7.exe (md5)	MS Windows installer	2.7	2014-01-02	1MB
MySQL-python-1.2.5.zip (md5)	Source		2014-01-02	106KB

Author: Andy Dustman
Home Page: https://github.com/farcepest/MySQLdb1
License: GPL
Platform: ALL
Categories

Development Status :: 5 - Production/Stable
Environment :: Other Environment
License :: OSI Approved :: GNU General Public License (GPL)
Operating System :: MacOS :: MacOS X
Operating System :: Microsoft :: Windows :: Windows NT/2000
Operating System :: OS Independent
Operating System :: POSIX
Operating System :: POSIX :: Linux

Figure 1.8

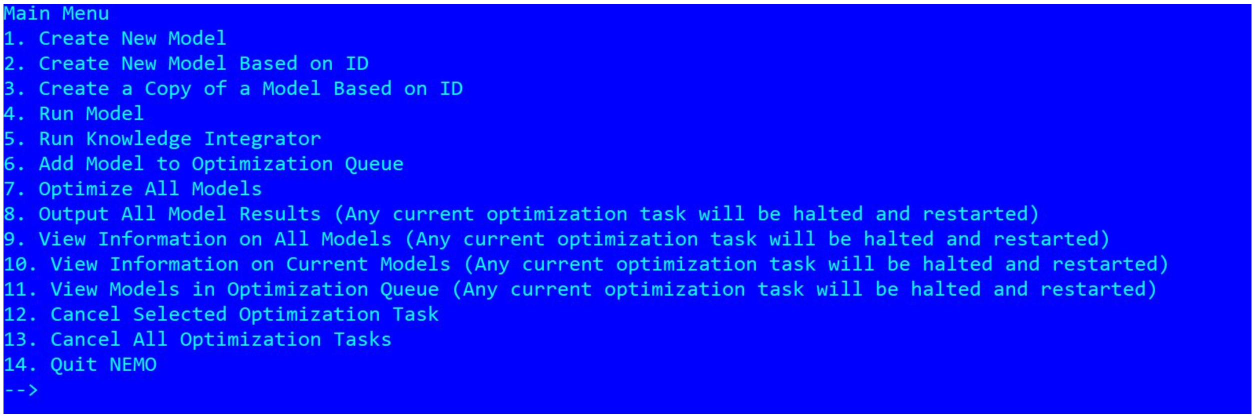
2. Compilation and Runtime Instructions

A. Starting NEMO

NEMO can be started by opening a command line utility in the root directory of your NEMO folder. Input the following command: `python NEMO.py`

B. Navigating NEMO

NEMO's primary mode of navigation consists of a 'Main Menu' (Figure 2.1) through which the various functions of NEMO can be accessed. The '`-->`' prompt indicates user input is required. Sections 4 - 8 detail these various modes.



```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
-->
```

Figure 2.1

C. Exiting NEMO

From the 'Main Menu' select option '14' to safely exit NEMO.

D. Restarting from Sudden Shutdown

Inevitably, a situation may arise in which NEMO must unexpectedly exit (power outage, etc.). NEMO is programmed to remember which model (if any) was running before shutdown and the current optimization queue.

When NEMO is restarted, models remaining to be evaluated are copied and evaluated. Models that were in the optimization queue are copied and optimization is restarted.

Only models in the optimization queue and those that were in the process of evaluation before the sudden shutdown will be recreated. However, NEMO keeps a store of all models created in NEMO's lifetime. Thus, the user can choose to copy models of their choosing (Section 4-D).

E. Troubleshooting NEMO

Should a user encounter issues with NEMO after restarting from a sudden shutdown, there are several restore options the user can employ.

The first method of troubleshooting is purging the current models and optimization queue. To do this, navigate to the KnowledgeBase directory, and run the following script `python KBRestore.py`. This will keep all the models.

If the issue is not resolved, first, contact the NEMO development team with screenshots and description of the error. If the team verifies this is a KnowledgeBase issue, run the following command: `mysql -u <USERNAME> -p < KBCreator.sql.` Insert your MySQL username for <USERNAME> and the corresponding password when prompted. This will remove all information stored in the database.

3. Non-Code Related Setup

A. Introduction

There is some non-code related setup that needs to be completed before NEMO is executed. First is the setup of the data to be read in by NEMO. Second is the configuration of the system itself through the `config.json` file.

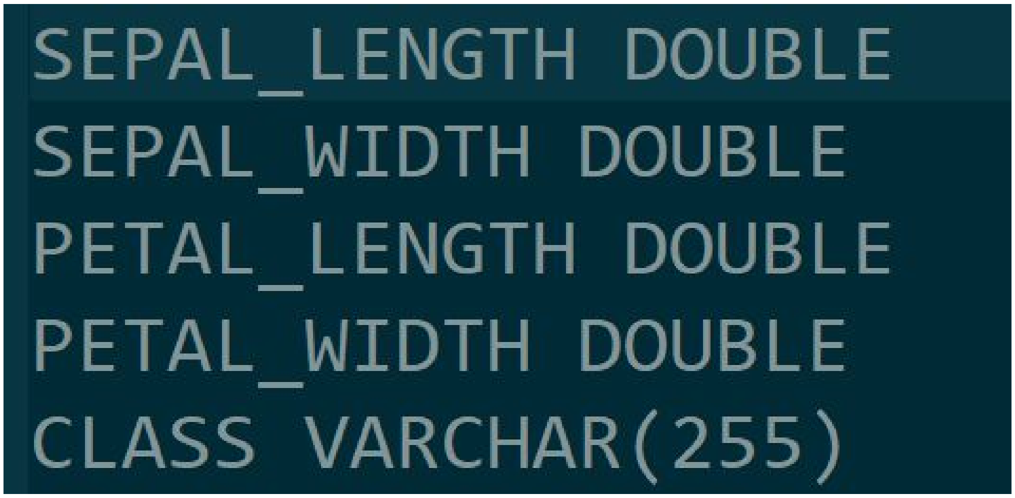
B. Data Setup

A. Schema Files

The schema file is a flat text file that describes the schema of the data provided to NEMO. Each row of a schema contains the name of the feature and its corresponding MySQL data type. Schema files are designated in the config.json file (Section 3-C).

The data types allowed are any legal MySQL type including, but not limited to, INT, DOUBLE, VARCHAR(X), and CHAR(X), where X is an integer denoting length of the string.

Figure 3.1 shows an example schema file for the Iris dataset (https://archive.ics.uci.edu/ml/datasets/Iris) .



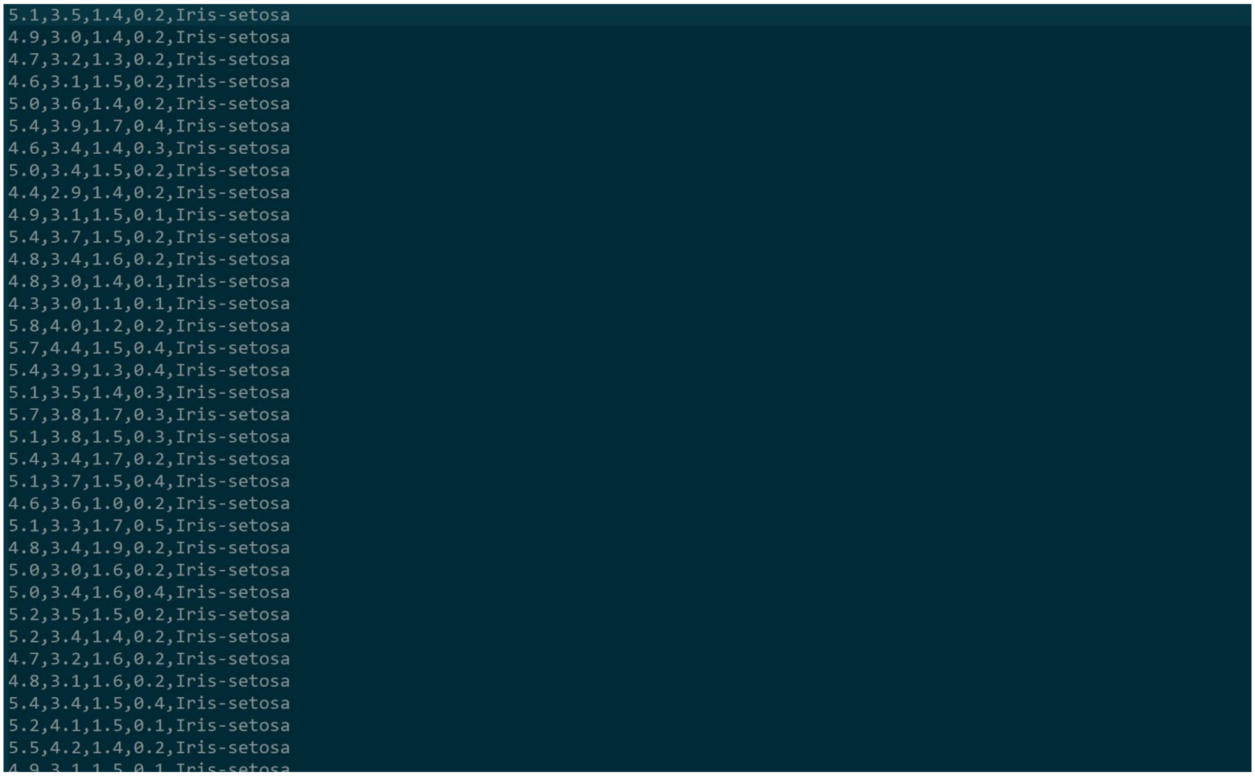
```
SEPAL_LENGTH DOUBLE
SEPAL_WIDTH DOUBLE
PETAL_LENGTH DOUBLE
PETAL_WIDTH DOUBLE
CLASS VARCHAR(255)
```

Figure 3.1

B. Data Files

The data file is a flat text file that describes the actual data to provide to NEMO. Each record in the data file is a comma-delimited row in the file. The order that features appear in the data file must match the order they are listed in the schema file (Section 3-B-A). Data files are designated in the config.json file (Section 3-C).

Figure 3.2 shows a partial data file for the Iris dataset (https://archive.ics.uci.edu/ml/datasets/Iris).



```
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
4.4,2.9,1.4,0.2,Iris-setosa
4.9,3.1,1.5,0.1,Iris-setosa
5.4,3.7,1.5,0.2,Iris-setosa
4.8,3.4,1.6,0.2,Iris-setosa
4.8,3.0,1.4,0.1,Iris-setosa
4.3,3.0,1.1,0.1,Iris-setosa
5.8,4.0,1.2,0.2,Iris-setosa
5.7,4.4,1.5,0.4,Iris-setosa
5.4,3.9,1.3,0.4,Iris-setosa
5.1,3.5,1.4,0.3,Iris-setosa
5.7,3.8,1.7,0.3,Iris-setosa
5.1,3.8,1.5,0.3,Iris-setosa
5.4,3.4,1.7,0.2,Iris-setosa
5.1,3.7,1.5,0.4,Iris-setosa
4.6,3.6,1.0,0.2,Iris-setosa
5.1,3.3,1.7,0.5,Iris-setosa
4.8,3.4,1.9,0.2,Iris-setosa
5.0,3.0,1.6,0.2,Iris-setosa
5.0,3.4,1.6,0.4,Iris-setosa
5.2,3.5,1.5,0.2,Iris-setosa
5.2,3.4,1.4,0.2,Iris-setosa
4.7,3.2,1.6,0.2,Iris-setosa
4.8,3.1,1.6,0.2,Iris-setosa
5.4,3.4,1.5,0.4,Iris-setosa
5.2,4.1,1.5,0.1,Iris-setosa
5.5,4.2,1.4,0.2,Iris-setosa
4.9,3.1,1.5,0.1,Iris-setosa
```

Figure 3.2

C. Configuration File

Configuration of NEMO occurs through the editing of `config.json` in the `config` directory. The following lists the sections and details the expected values.

`DATABASE` - Details information about the database connection. Table 3.4 details the layout for the DATABASE section.

Name	Type	Expected Value
`HOST`	`String`	Host name for the database server.
`PORT`	`Integer`	Port number for the database server
`USER`	`String`	Username to log on to the database server
`PASS`	`String`	Password to log on to the database server
`DB`	`String`	Name of database to use; default is NEMO_KB

Table 3.4

DATA - Details information about the data NEMO uses. Table 3.5 details the layout for the DATABASE section.

Name	Type	Expected Value
DATA	String	Filename for comma-delimited data (detailed in Section 3-B)
SCHEMA	String	Filename for schema file (detailed in Section 3-B)
CLASS	String	Name of class to predict
MULTI-CLASS	String	True if class to predict is not binary. False otherwise

Table 3.5

KNOWLEDGE_INTEGRATOR - Details settings for the Knowledge Integrator. Table 3.6 details the layout for the DATABASE section.

Name	Type	Expected Value
STACKER	String	Type of algorithm to use for stacking. Acceptable values: Logistic Regression Decision Tree SVM SVM is recommened
OTHER_PREDICTIONS	String	True if dataset includes other predictions to use in stacker. False otherwise. Keep this as False.

Table 3.6

Part II: Using NEMO

4. Creating Models with NEMO

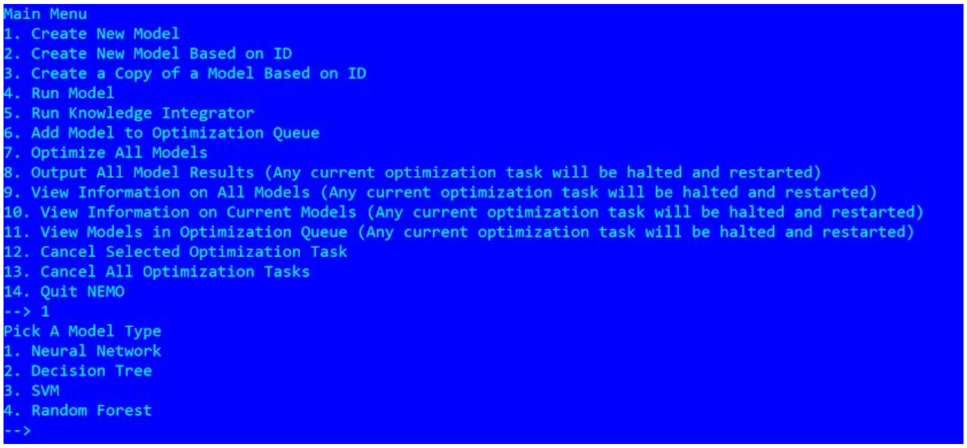
A. Introduction

There are multiple ways to create a predictive model within NEMO. The following sections list how and the optimal uses for each selection.

B. Creating a Brand New Model

This option creates a new, random predictive model in with one of the following algorithms: Artificial Neural Networks, Decision Tree (CART implementation), Support Vector Machines, and Random Forest.

To do so, select option '1' in the 'Main Menu' and input the integer beside the name of the algorithm to select it. This process is shown in Figure 4.1.



```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 1
Pick A Model Type
1. Neural Network
2. Decision Tree
3. SVM
4. Random Forest
-->
```

Figure 4.1

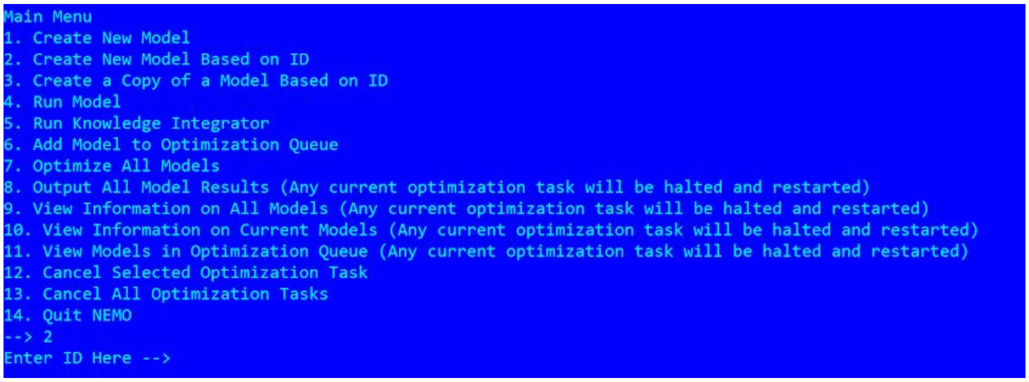
This creates a random model and is best used when there is no reason for using a predetermined model, or no model using the desired algorithm exists in NEMO.

C. Creating a Model Based on a Previously Created Model

This option creates a predictive model based on a previously created model; however, the new model will have a *different* 'ID', allowing for separate optimization. Please see 'A Note on IDs' below for more information on IDs of algorithms in NEMO.

A Note on IDs: IDs are a nine digit personal identification number for models in NEMO. They are randomly generated upon the creation of a model. The only ID unavailable for use by the system is '000000000'. This is allocated for the Knowledge Integrator (see section 7). Barring the physical constraints of the system, NEMO can process 9^9 unique models.

To utilize this option, select option ‘2’ from the ‘Main Menu’ and then input the ID of the model. If the identification number does not match that of a model in the database, NEMO will notify the user. This process is illustrated in Figure 4.2. More information on viewing models and their IDs is found in Section 8.



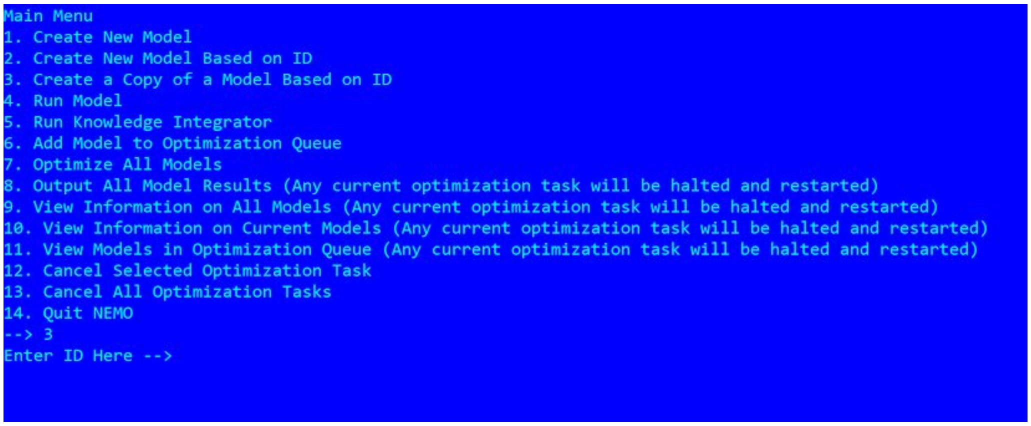
```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 2
Enter ID Here -->
```

Figure 4.2

D. Copying a previously created model for use

This option creates a predictive model based on a previously created model; however, the new model will have the *same* ‘ID’ as the model chosen, effectively loading the model from the database for use by NEMO. Please see ‘A Note on IDs’ in Section 4-C for more information on IDs.

To utilize this option, select option ‘3’ from the ‘Main Menu’ and then input the ID of the model. If the identification number does not match that of a model in the database or is already loaded into NEMO, then NEMO will notify the user of this. This process is illustrated in Figure 4.3. More information on viewing models and their IDs is found in Section 8.



```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 3
Enter ID Here -->
```

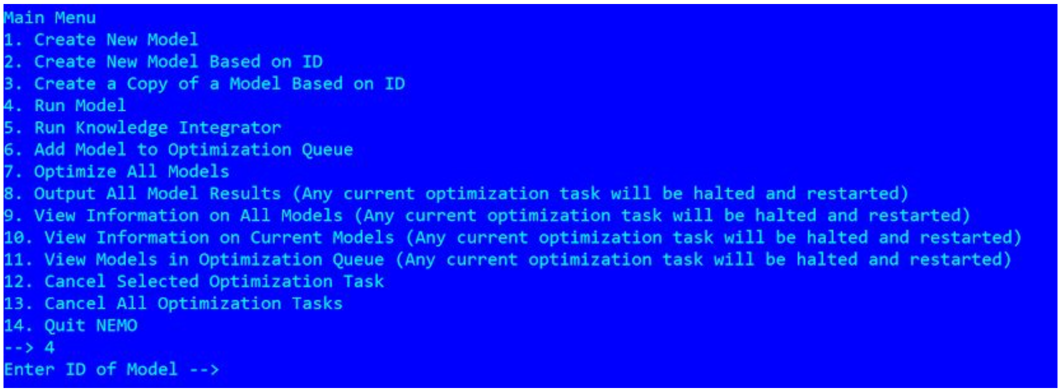
Figure 4.3

5. Running a Model with NEMO

After one or more models have been created, NEMO can evaluate the effectiveness of model on its predictive ability. This is done through holdout validation using Sci-Kit Learn's default setting of having 25% of the data used as the testing set and 75% of the data used as the training set.

To run a model, select option '4' from the 'Main Menu' and then input the model ID when prompted. If the provided ID does not match an ID of a model in the system, NEMO will exit the process. This process is illustrated in Figure 4.4. Please see 'A Note on IDs' in Section 4-C for more information on IDs.

Upon completion of the evaluation process, the measures of Accuracy, F1, Precision, and Recall will be saved in the database and can be viewed by choosing to view all model results.. More information on viewing model results is found in Section 8.



```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 4
Enter ID of Model -->
```

Figure 4.4

6. Optimizing a Model

A. Introduction

NEMO can optimize predictive models to increase a metric of choice. Currently, NEMO uses a *Coordinate Ascent* approach to improve the *accuracy* of the models.

In short, this implementation of Coordinate Ascent cycles through two to four parameters on each algorithm, modifying them in a single direction (e.g., increasing the max depth for a Decision Tree or Random Forest) until the metric has reached the most optimal value for the specific metric.

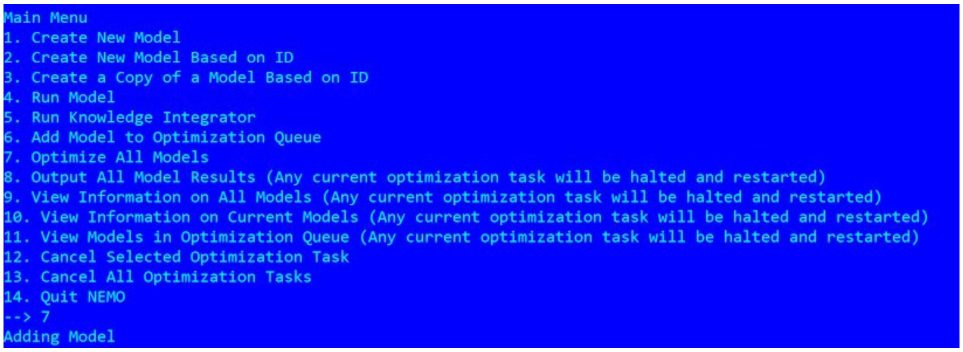
NEMO has the capability to run this optimization procedure on multiple algorithm of the user's choice. The system cycles through multiple models, optimizing one at a time until a shut-off signal is received. Should NEMO be placed as a background process, optimization will continue. When the user chooses to quit NEMO (see Section 2-C), the optimization will halt before shutdown.

Should NEMO be killed unexpectedly, the models being optimized will be restored upon restart of NEMO, and optimization will continue. Should NEMO begin re-optimizing from a sudden exit, accuracy may seem to decrease. This is to be expected due to the re-splitting of the data as a result of re-creating the models. Please see Section 2-D for more information about restarting from sudden exits.

B. Optimizing All Models

This option places all models that have been created in this session on the optimization queue.

To utilize this option, select option '7' from the 'Main Menu'. For each model successfully added to the queue, 'Adding Model' will be printed. Please see Figure 5.1.

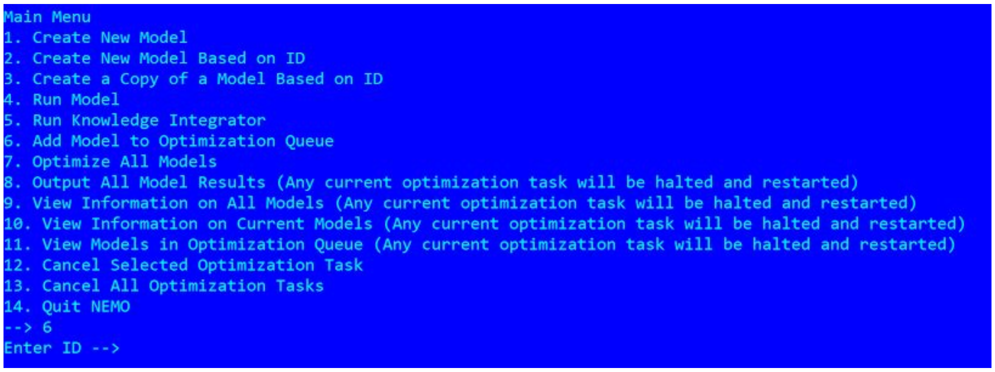


```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 7
Adding Model
```

Figure 5.1

C. Optimizing Specific Models

The option is available to optimize a specific model. This places a single model on the optimization queue as selected by ID. Upon success, 'Adding model' will be printed. Should there be an error with the ID, an error message will be printed. Please see Figure 5.2.



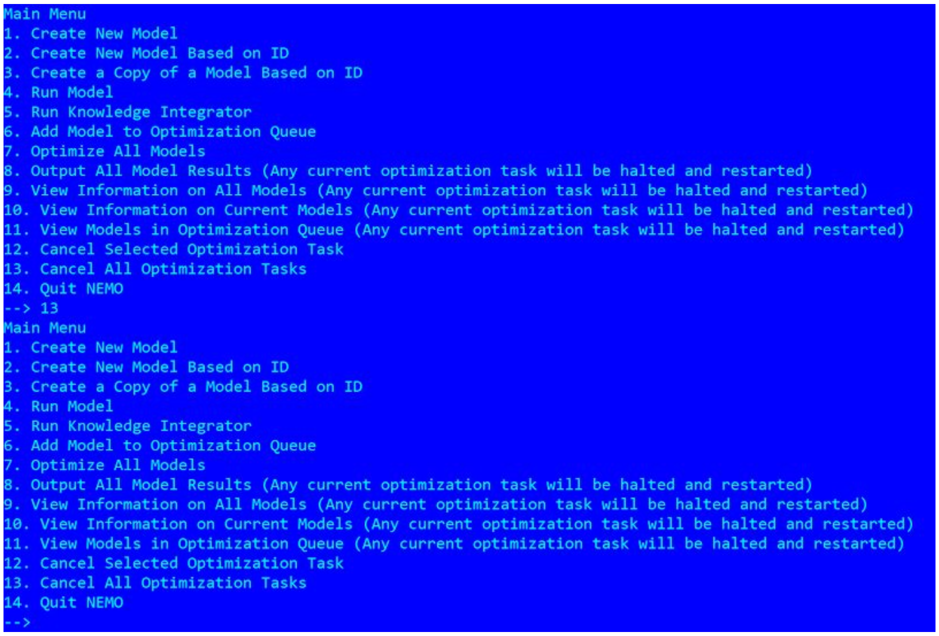
```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 6
Enter ID -->
```

Figure 5.2

D. Ending Optimization of All Models

The option is also available to end all optimization tasks. This will stop the optimization execution and remove all models from the optimization queue.

To utilize this, select option '13' from the 'Main Menu'. Upon successful completion, the menu will be re-printed with a new input prompt. See figure 5.3



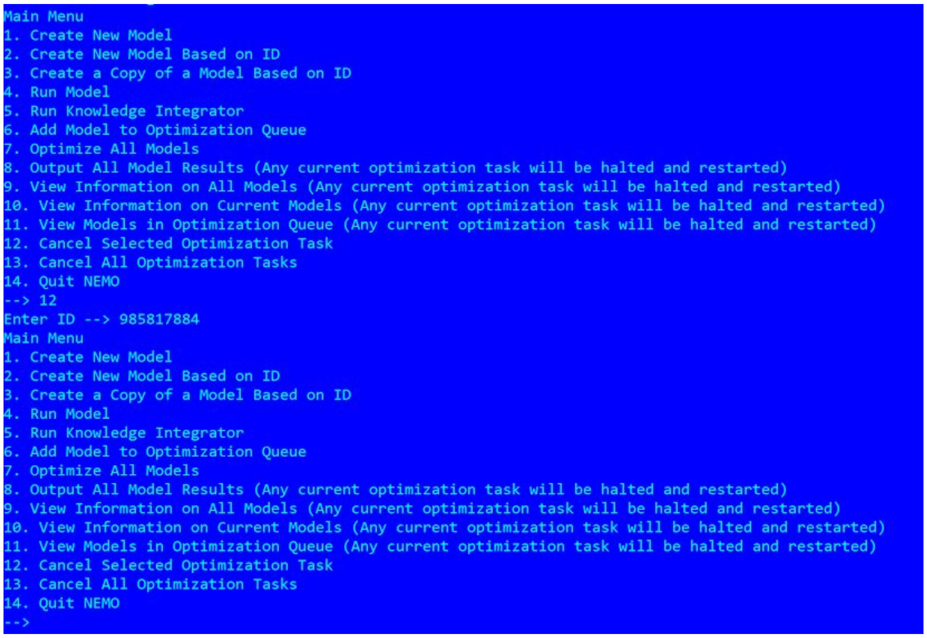
```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 13
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
-->
```

Figure 5.3

E. Ending Optimization of a Specific Model

The option is available to end optimization of a specific model, selected by ID.

To utilize this, select option '12' in the 'Main Menu' and enter the ID of the model to cancel optimization. Upon successful completion, the menu will be re-printed with a new input prompt. See figure 5.4.



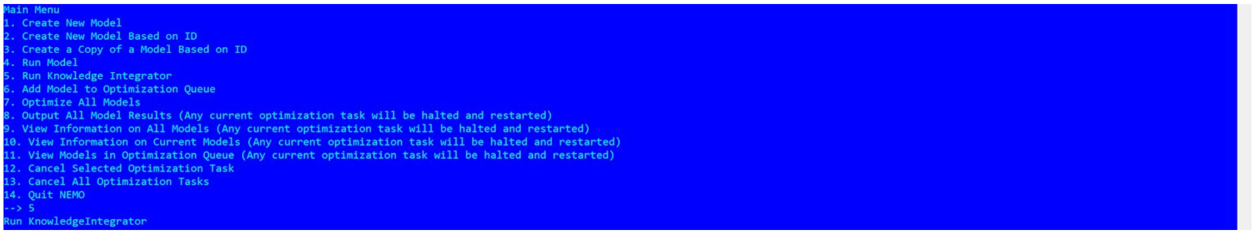
```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 12
Enter ID --> 985817884
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
-->
```

Figure 5.4. Note the ID varies by model.

7. Utilizing the Knowledge Integrator

After multiple models have been created, NEMO has the ability to use the predictions from the individual models to make more sound predictions. The machine learning algorithm used in this feature can be configured in the configuration file. See Section 3.X for more information.

To utilize this feature, select option ‘5’ from the ‘Main Menu’ (Figure 7.1). Upon completion of the evaluation process, the measures of Accuracy, F1, Precision, and Recall will be saved in the database and can be viewed by choosing to view all model results. More information on viewing model results is found in Section 8. If NEMO encounters an error with the Knowledge Integrator (i.e., no models created), an error message will be printed.



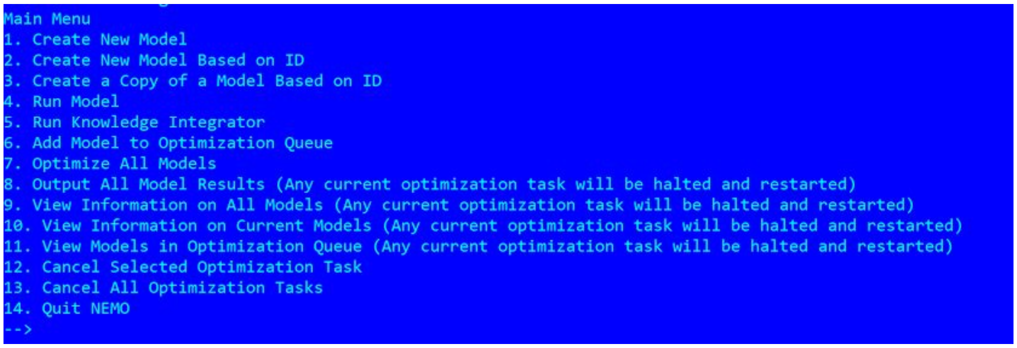
```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 5
Run KnowledgeIntegrator
```

Figure 7.1

8. Viewing Information on Models and Results

A. Introduction

NEMO can provide information on model results to information on individual models. Options 8-11 in Figure 8.1 are the possible viewing modes.

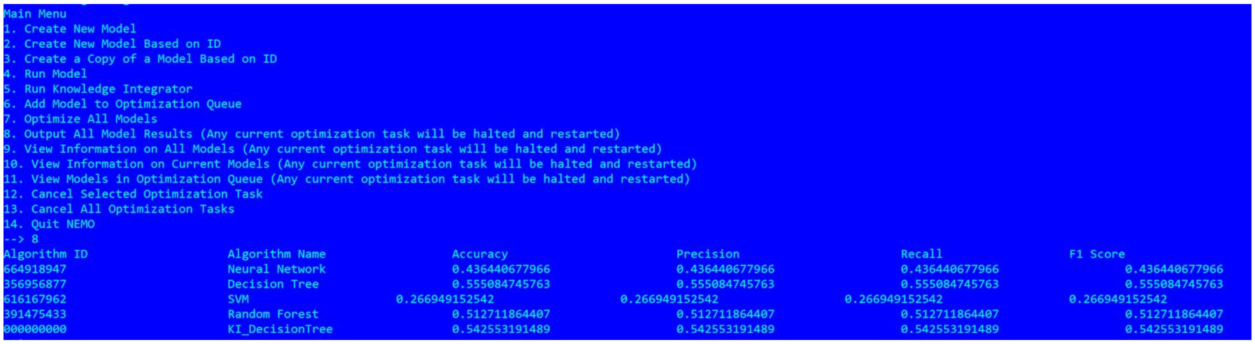


```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
-->
```

B. Viewing Model Results

This mode allows users to view information on model execution. The output of this mode contains the model's ID, the type of model (Neural Network, SVM, Decision Tree, Random Forest), and the metrics measured from the execution of the model (Accuracy, F1, Precision, and Recall).

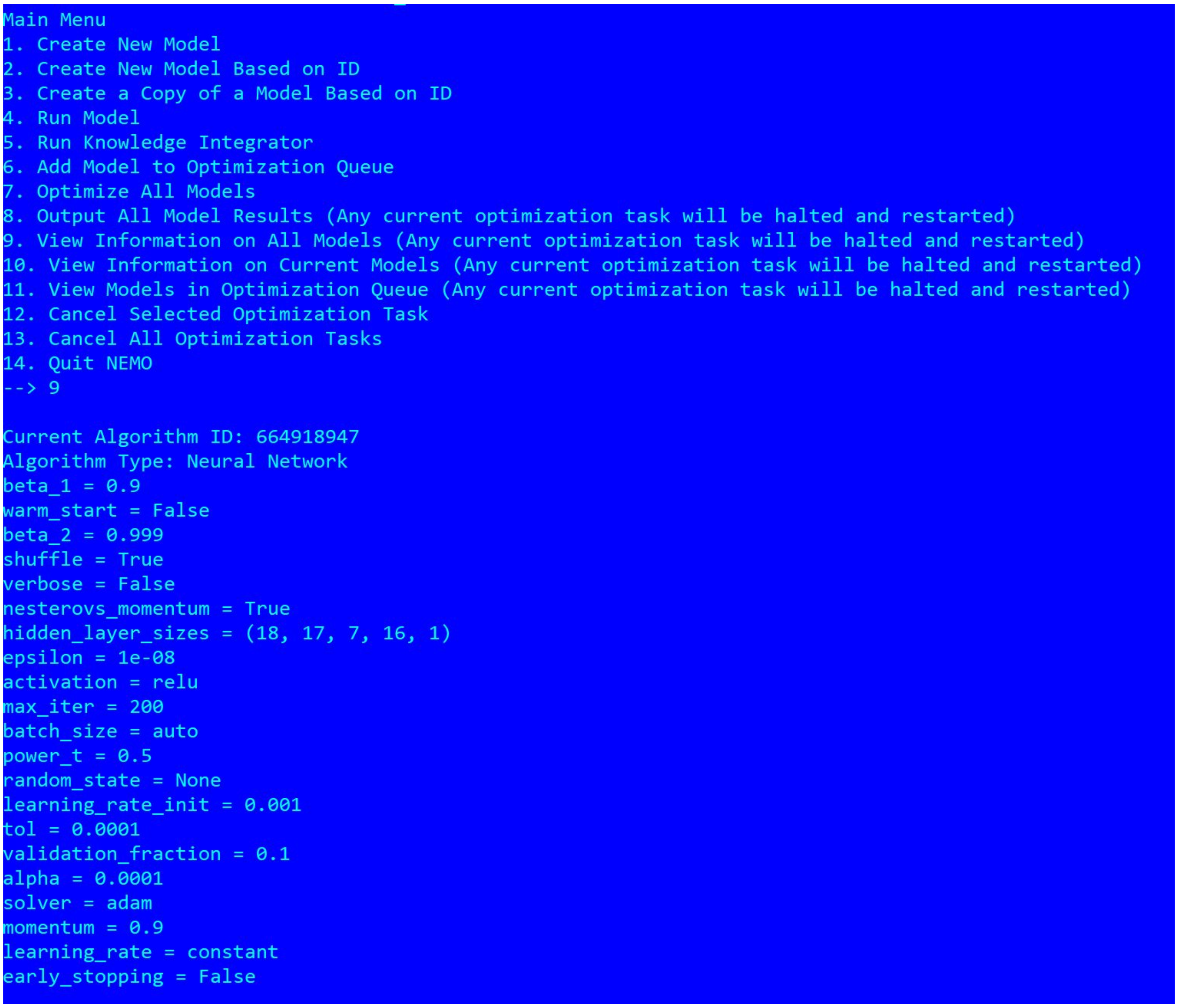
To use this, select option '8' from the 'Main Menu'. Figure 8.2 shows an example of the result output.



```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 8
Algorithm ID            Algorithm Name          Accuracy                Precision               Recall                  F1 Score
664918947               Neural Network          0.436440677966          0.436440677966          0.436440677966                  0.436440677966
356956877               Decision Tree           0.555084745763          0.555084745763          0.555084745763                  0.555084745763
616167962               SVM             0.266949152542          0.266949152542          0.266949152542          0.266949152542
391475433               Random Forest           0.512711864407          0.512711864407          0.512711864407                  0.512711864407
000000000               KI_DecisionTree         0.542553191489          0.542553191489          0.542553191489                  0.542553191489
```

C. Viewing Information on All Models

This mode allows users to view information on all models that have been created in NEMO's lifetime. To use this, select option '9' from the 'Main Menu'. The results include the model's ID, model type, and a list of all adjustable parameters and their current values. Figure 8.3 shows an example of the result output.



```
Main Menu
1. Create New Model
2. Create New Model Based on ID
3. Create a Copy of a Model Based on ID
4. Run Model
5. Run Knowledge Integrator
6. Add Model to Optimization Queue
7. Optimize All Models
8. Output All Model Results (Any current optimization task will be halted and restarted)
9. View Information on All Models (Any current optimization task will be halted and restarted)
10. View Information on Current Models (Any current optimization task will be halted and restarted)
11. View Models in Optimization Queue (Any current optimization task will be halted and restarted)
12. Cancel Selected Optimization Task
13. Cancel All Optimization Tasks
14. Quit NEMO
--> 9

Current Algorithm ID: 664918947
Algorithm Type: Neural Network
beta_1 = 0.9
warm_start = False
beta_2 = 0.999
shuffle = True
verbose = False
nesterovs_momentum = True
hidden_layer_sizes = (18, 17, 7, 16, 1)
epsilon = 1e-08
activation = relu
max_iter = 200
batch_size = auto
power_t = 0.5
random_state = None
learning_rate_init = 0.001
tol = 0.0001
validation_fraction = 0.1
alpha = 0.0001
solver = adam
momentum = 0.9
learning_rate = constant
early_stopping = False
```

D. Viewing Information on Current Models

This mode allows users to view information on all models that have been created in the current session of NEMO. To use this, select option ‘10’ from the ‘Main Menu’. The results include the model’s ID, model type, and a list of all adjustable parameters and their current values. Figure 8.3 shows an example of the result output.

E. Viewing information on Models in the Optimization Queue

This mode allows users to view information on all models that have been created in the current session of NEMO. To use this, select option ‘11’ from the ‘Main Menu’. The results include the model’s ID, model type, and a list of all adjustable parameters and their current values. Figure 8.3 shows an example of the result output.