openGauss DB4AI 特性实验

姓名： 汤清云 学号： 2013536

实验步骤：

- ECS 弹性云服务器购买
- openGauss 数据库安装
- openGauss DB4AI 特性实验
- 云服务器资源释放/云服务器关机

实验报告

实验步骤截图：

截图 1： 数据库编译安装版本截图

```
ERROR:  The old password is invalid.
postgres=# select version();
                                                            version

-----------------------------------------------------------------------------------------------------------------------------
---------------------------------------------------------
 PostgreSQL 9.2.4 (GaussDB Kernel V500R001C20 build f40ff5a0) compiled at 2022-03-04 10:36:23 commit 0 last mr  debug on aarch6
4-unknown-linux-gnu, compiled by g++ (GCC) 7.3.0, 64-bit
(1 row)

postgres=# 
```

截图 2： 数据库状态截图

```
postgres=# \q
[omm@opengauss01 openGauss-server]$ gs_ctl status
[2022-03-04 10:48:15.041][279178][][gs_ctl]: gs_ctl status,datadir is /opt/software/openGauss/data
gs_ctl: server is running (PID: 279089)
/opt/software/openGauss/bin/gaussdb "-D" "/opt/software/openGauss/data"
[omm@opengauss01 openGauss-server]$ 
```

截图 3：数据库进程截图（包含数据库服务器主机名）

```
[omm@opengauss01 openGauss-server]$ ps -ef|grep omm
root        6796    6559  0 10:02 pts/0    00:00:00 su - omm
omm         6797    6796  0 10:02 pts/0    00:00:00 -bash
omm       279089       1  0 10:45 pts/0    00:00:02 /opt/software/openGauss/bin/gaussdb -D /opt/software/openGauss/data
omm       279201    6797  0 10:49 pts/0    00:00:00 ps -ef
omm       279202    6797  0 10:49 pts/0    00:00:00 grep --color=auto omm
[omm@opengauss01 openGauss-server]$
```

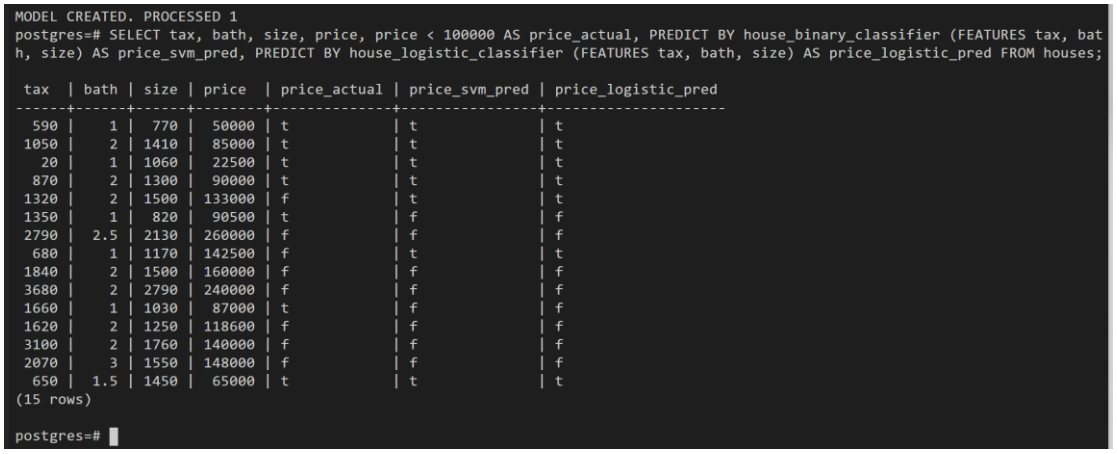
截图 4：SVM 二分类模型执行结果截图

```
postgres=# SELECT * FROM gs_model_warehouse WHERE modelname =
postgres-# 'house_binary_classifier';
        modelname         | modelowner |         createtime          | processedtuples | discardedtuples | preprocesstime | exect
ime | iterations | outputtype |      modeltype       |                                                          query
                                                                              | modeldata |                  weight                 |
               hyperparametersnames                                                     |             hyperparametersvalues
    |           hyperparametersoids             |  coefnames   |  coefvalues   | coefoids |          trainingscoresname          |
     trainingscoresvalue             | modeldescribe
-------------------------+------------+-----------------------------+-----------------+-----------------+----------------+------
----+------------+------------+---------------------+------------------------------------------------------------------------------
--------------------------------------------------------------------------------+-----------+-----------------------------------------+
--------------------------------------------------------------------------------------------------------------------------------------
--+------------------------------------------+--------------+---------------+----------+------------------------------------------+-----
-----------------------------------+---------------
 house_binary_classifier |         10 | 2022-03-04 10:54:31.162536 |              15 |               0 |              0 |  .003
694 |        100 |         16 | svm_classification | CREATE MODEL house_binary_classifier USING svm_classification FEATURES tax
, bath, size TARGET price < 100000 FROM houses; |           | {-.0623524,.000946067,.037792,.0225448} | {batch_size,decay,lambd
a,learning_rate,max_iterations,max_seconds,optimizer,tolerance,seed,verbose} | {1000,.95,.01,.8,100,0,gd,.0005,1646362471,false
} | {23,701,701,701,23,23,1043,701,23,16} | {categories} | {false,true} |          | {accuracy,f1,precision,recall,loss} | {.66
6667,.666667,.625,.714286,7.10655} |
(1 row)

postgres=#
```

截图 5：超参设置新模型结果截图

```
NOTICE:  Hyperparameter max_iterations takes value DEFAULT (100)
NOTICE:  Hyperparameter max_seconds takes value DEFAULT (0)
NOTICE:  Hyperparameter optimizer takes value DEFAULT (gd)
NOTICE:  Hyperparameter tolerance takes value DEFAULT (0.000500)
NOTICE:  Hyperparameter seed takes value DEFAULT (0)
NOTICE:  Hyperparameter verbose takes value DEFAULT (FALSE)
NOTICE:  GD shuffle cache size 342391
MODEL CREATED. PROCESSED 1
postgres=# SELECT * FROM gs_model_warehouse WHERE modelname = 'house_binary_classifier';
        modelname         | modelowner |         createtime          | processedtuples | discardedtuples | preprocesstime | exect
ime | iterations | outputtype |      modeltype       |
   query                                                                                                  | modeldata |                weight
                    |                          hyperparameternames                                                     |
         hyperparametersvalues                   |             hyperparametersoids            |  coefnames   |  coefvalues   | coefoids
|        trainingscoresname          |           trainingscoresvalue             | modeldescribe
-------------------------+------------+-----------------------------+-----------------+-----------------+----------------+------
----+------------+------------+---------------------+--------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------+-----------+-----------------
------------------+--------------------------------------------------------------------------------------------------------+----
------------------------------------------+--------------------------------------+--------------+--------------+----------
+-----------------------------------+---------------
 house_binary_classifier |         10 | 2022-03-04 10:58:54.367218 |              15 |               0 |              0 |  .011
772 |        100 |         16 | svm_classification | CREATE MODEL house_binary_classifier USING svm_classification FEATURES tax
, bath, size TARGET price < 100000 FROM houses WITH batch_size=1, learning_rate=0.001; |           | {-.00213339,2.51169e-05,.0
0191756,.000454951} | {batch_size,decay,lambda,learning_rate,max_iterations,max_seconds,optimizer,tolerance,seed,verbose} | {1,
.95,.01,.001,100,0,gd,.0005,1646362734,false} | {23,701,701,701,23,23,1043,701,23,16} | {categories} | {false,true} |
| {accuracy,f1,precision,recall,loss} | {.733333,.714286,.714286,.714286,9.40403} |
(1 row)

postgres=#
postgres=#
```

截图 6：SVM 二分类模型与逻辑回归模型执行结果比较截图



```
MODEL CREATED. PROCESSED 1
postgres=# SELECT tax, bath, size, price, price < 100000 AS price_actual, PREDICT BY house_binary_classifier (FEATURES tax, bat
h, size) AS price_svm_pred, PREDICT BY house_logistic_classifier (FEATURES tax, bath, size) AS price_logistic_pred FROM houses;

 tax  | bath | size | price  | price_actual | price_svm_pred | price_logistic_pred
------+------+------+--------+--------------+----------------+---------------------
  590 |    1 |  770 |  50000 | t            | t              | t
 1050 |    2 | 1410 |  85000 | t            | t              | t
   20 |    1 | 1060 |  22500 | t            | t              | t
  870 |    2 | 1300 |  90000 | t            | t              | t
 1320 |    2 | 1500 | 133000 | f            | t              | t
 1350 |    1 |  820 |  90500 | t            | f              | f
 2790 |  2.5 | 2130 | 260000 | f            | f              | f
  680 |    1 | 1170 | 142500 | f            | t              | t
 1840 |    2 | 1500 | 160000 | f            | f              | f
 3680 |    2 | 2790 | 240000 | f            | f              | f
 1660 |    1 | 1030 |  87000 | t            | f              | f
 1620 |    2 | 1250 | 118600 | f            | f              | f
 3100 |    2 | 1760 | 140000 | f            | f              | f
 2070 |    3 | 1550 | 148000 | f            | f              | f
  650 |  1.5 | 1450 |  65000 | t            | t              | t
(15 rows)

postgres=#
```

实验思考题：

1. 为什么需要通过源码编译，完成数据库的安装？简述安装数据库所需步骤

   答：1.通过源码编译可以自己设定变量，按照需求进行安装，且版本可以自选，使其更贴近于实验所需环境，灵活性大。

   1.1. 创建 openGauss 数据库的安装用户 omm 及其属组 dbgrp

   1.2. 创建 openGauss 源码存放及 openGauss 安装路径

   1.3. 下载第三方编译库并解压。

   1.4. 下载 openGauss 源码。

   1.5. 上传 cmake 包并解压。

   1.6. 使用 yum 安装依赖包。

   1.7. 替换 python 版本为最新版本。

   1.8. 修改/opt/software 路径的用户所属组及权限。

1.9. 切换 omm 用户，配置环境变量。

1.10. 生成配置文件并编译安装。

1.11. 初始化数据库、启动数据库。

1.12. 查询数据库版本，验证数据库状态。

2. 分类模型与回归模型有何不同？什么是 SVM 算法?

答：分类和回归的区别在于输出变量的类型不同。定量输出称为回归，或者说是连续变量预测；定性输出称为分类，或者说是离散变量预测。分类模型和回归模型本质一样，分类模型可将回归模型的输出离散化，回归模型也可将分类模型的输出连续化。

SVM 是一种二分类模型，将实例的特征向量映射为空间中的一些点，SVM 就是想画出一条线来最好地区分这两类点，适用于中小型数据样本、非线性、高维的分类问题。所希望找到的这条效果最好的线就是具有“最大间隔的划分超平面”。

3. 分类问题有哪些指标，其含义是什么?

答：

1. 精确率：正确预测为正的占全部预测为正的比例。

2. 准确率：所有预测正确的占总的比重。

3. 召回率/查准率：正确预测为正的占所有实际为正的比例。

4. F1 值：算术平均数除以几何平均数。

5. ROC 曲线：反映敏感性和特异性连续变量的综合指标。越靠拢(0,1)点，越偏离 45°对角线越好。

6. AUC 值：ROC 曲线下的面积，不会超过 1 。=1：完美分类器，存在至少一个阈值能得出完美预测。0.5<AUC<1：优于随机猜测，妥善设定阈值可以有预测价值。=0.5：和随即猜测一样，没有预测价值。<0.5：比随即猜测还差，但如果反预测而行则优于随机猜测。

4. 回归问题有哪些评价指标，其含义是什么?

答：

1.皮尔逊相关系数，值为【-1，1】，用来描述两个变量之间的线性相关性，越接近 1 说明模型性能越好。

2.解释方差分数，值为[0,1]，用来描述自变量对因变量方差变化的解释性成都，即特征对目标值在模型中的拟合程度，越接近 1 说明模型性能越好。

3.平均绝对误差，又称 L1 范数，值的范围为[0,+∞]，用来描述预测结果对目标值的接近程度，值越小说明模型性能越好。不可导不可求梯度。

4.均方差，又称 L2 范数，值的范围为[0,+∞]，用来描述预测结果对目标值的接近程度，值越小说明模型性能越好。一阶可导可求梯度。

5.r2 分数，值的范围为[-∞,1]，用来描述因变量的变化被自变量通过回归模型解释的比例，是定性评价而非定量评价。值越接近 1 说明模型性能越好。且 r2 分数会随着样本量的增加而增加。不能直接应用于不同回归

模型的性能比较。

6.调整 r2 分数，值的范围为[-∞,1]，用来描述因变量的变化被自变量通过回归模型解释的比例，是定性评价而非定量评价。值越接近 1 说明模型性能越好。能直接应用于不同回归模型的性能比较。