Experiement 1

孟昶-2021214431

October 2, 2021

1　线性回归

1. 简单线性回归

1) 库

sklearn.datasets，sklearn.linear_model，numpy(numpy.random，numpy.linalg)lm()，matplotlib

2) 要求及步骤

a) 划分数据集，分训练集和测试集；用 sklearn.linear_model.LinearRegression() 完成一个简单线性回归，了解预测变量和响应变量之间的关系，关系强弱，正负相关性。

b) 绘制响应变量和预测变量关系图，绘制最小二乘回归线。

c) 使用 LinearRegression 模型自带的评估模块，并输出评估结果。from sklearn.metrics import r2_score,mean_squared_error,mean_absolute_error

d) 使用线性回归模型 LinearRegression 和 SGDRegressor 分别对波士顿房价数据进行训练及预测，给出评估结果

答:

a) 详见代码

b) 如图

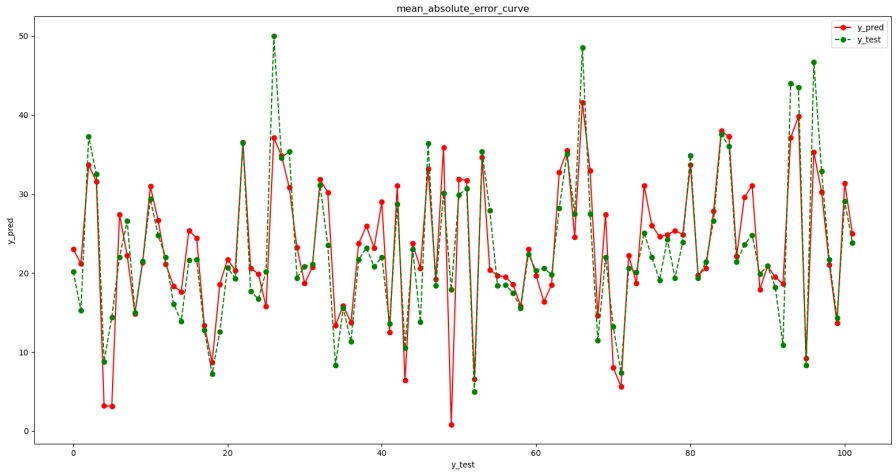

Figure 1: 最小二乘回归曲线

c) 如图

```
r2_score: 0.7789207451814428
mean_squared_error 18.495420122448312
mean_absolute_error 3.113043746893405
```

Figure 2: LinearRegression 评估指标

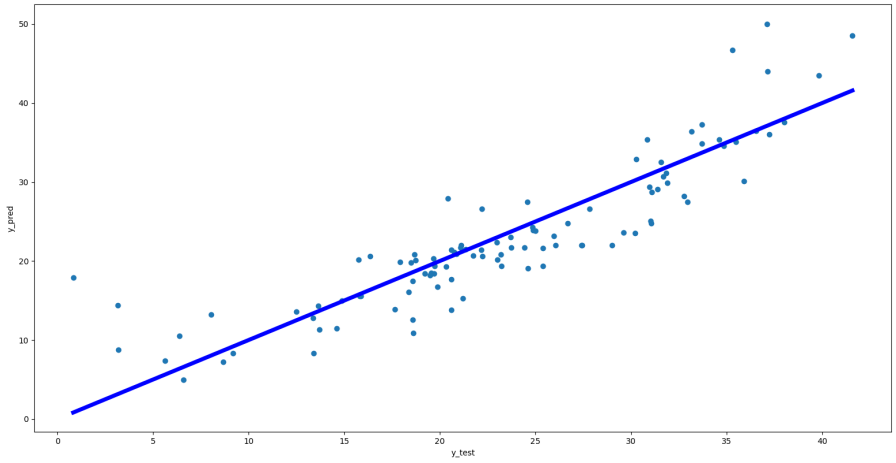

Figure 3: 响应变量和预测变量散点图

d) 如图

```
sg_r2_score: -4.704810996950406e+26
sg_mean_squared_error 3.9360299118399443e+28
sg_mean_absolute_error 194575761101054.8
```

Figure 4: SGDRegressor 评估指标

2. 多元线性回归

用 Boston 房价数据集进行多元线性回归。

1) 要求及步骤

a) 作出数据集中的所有变量的散点图矩阵；matplotlib.pyplot.scatter(a,b)

b) 计算变量之间的相关系数矩阵。

c) 用 polynominalData = sklearn.preprocessing.PolynomialFeatures(degree=2).fit _transform(boston.data) 进行多元线性回归，给出性能评估。

d) 进行交叉验证分析。可参考：http://scikit-learn.org/stable/auto_examples/model _selection/plot_cv_predict.html #sphx-glr-auto-examples-model-selection-plot-cv-predict-py

e) 随着通过划分比例观察训练数据量的增加，对训练的性能评分的变化和测试评分的变化？绘制一个曲线。

答：

a) 如图

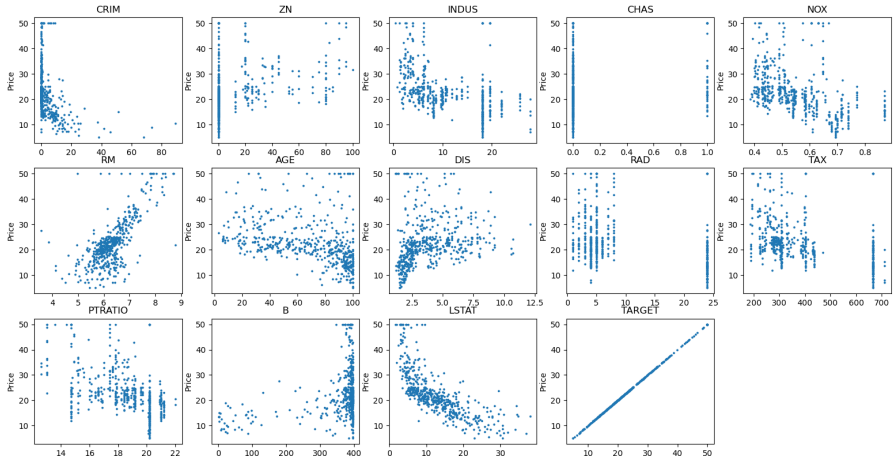

Figure 5: 所有变量的散点图矩阵

b) 如图

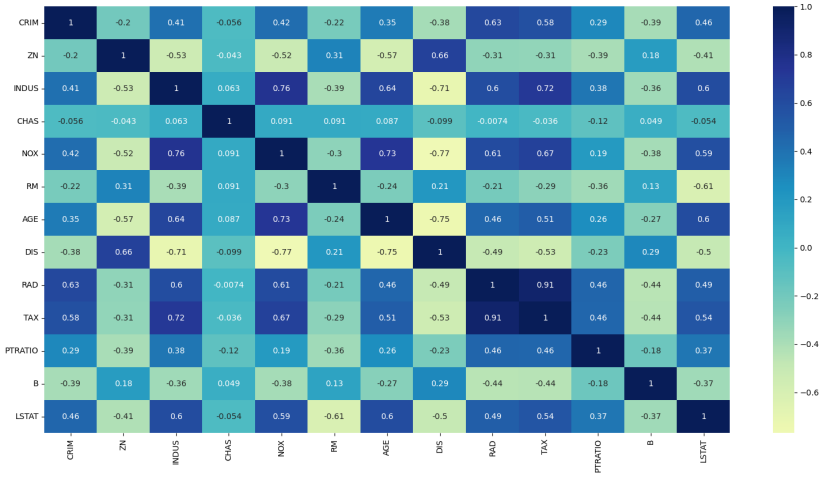

Figure 6: 相关系数矩阵

c) 如图

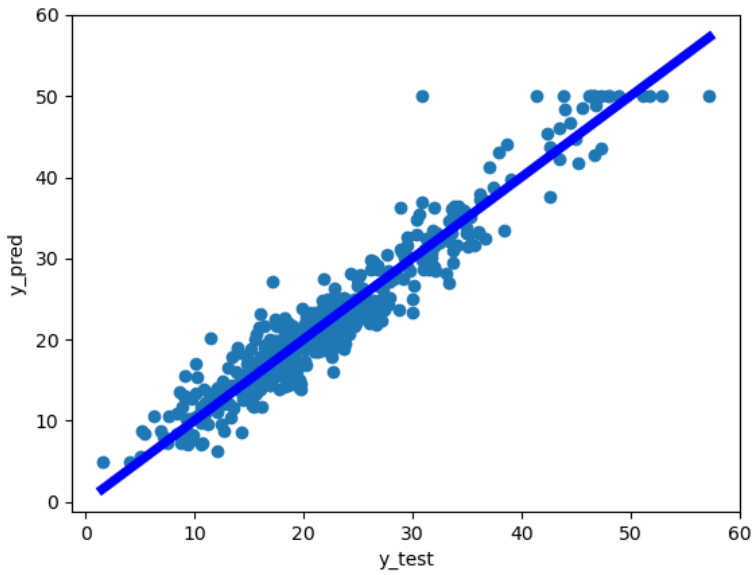

Figure 7: 多元线性回归之后的响应变量和预测变量散点图

```
r2_score: 0.9115088173072019
mean_squared_error 7.470386366660169
mean_absolute_error 2.066887805207445
```

Figure 8: 多元线性回归之后的 LinearRegression 评估指标

d) 详见代码

e)

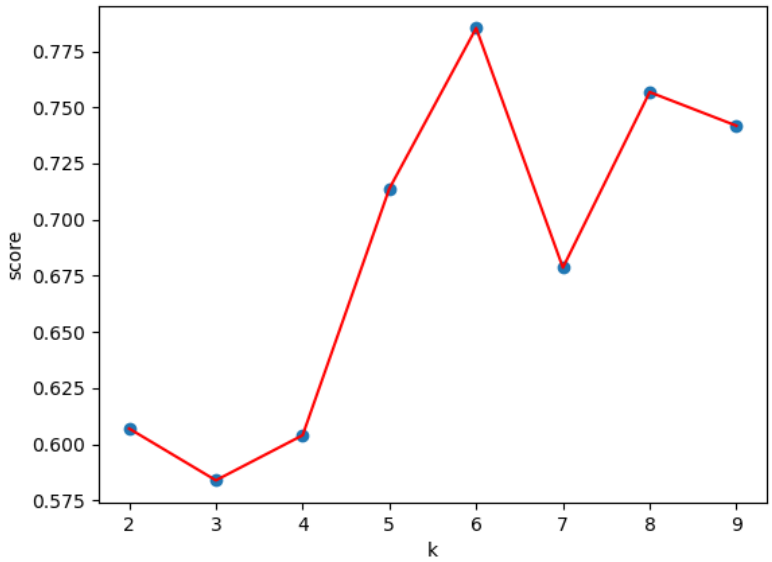

Figure 9: 测试评分随 k 的变化

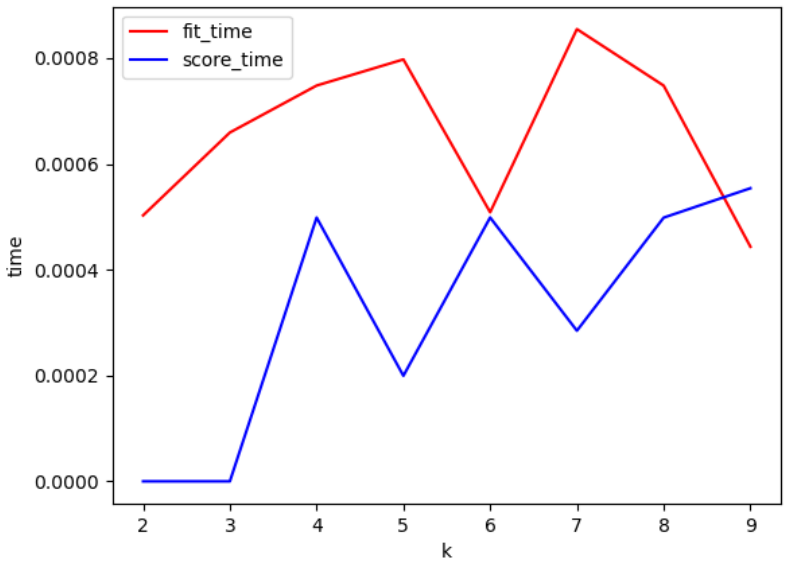

Figure 10: 训练的性能评分随 k 的变化

3. 岭回归与 Lasso 回归（参考下面链接掌握这部分内容）

用 Boston 房价数据集进行岭回归和 Lasso 回归。

1) 要求及步骤

a) 分别用 sklearn.linear_model. Ridge () 和 sklearn.linear_model. Lasso() 实现岭回归和 Lasso 回归进行模型训练，进一步理解数据，分析不同输入特征与输出变量之间的关系强弱和相关性。

b) 在测试集上完成预测，并输出评估结果，与一般的多元线性回归的结果进行对比。

c) 改变岭回归和 Lasso 回归中的参数 alpha 的值，绘制回归系数随 alpha 的变化图，观察预测效果的变化和不同输入变量对预测结果的影响力。

可参考：

https://www.cnblogs.com/magle/p/5878967.html https://zhuanlan.zhihu.com/p/165493873

答：

a) 如图

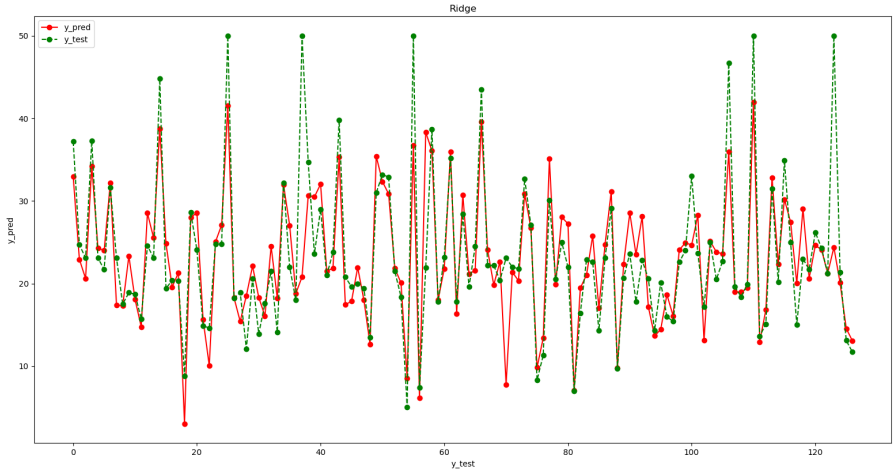

Figure 11: Ridge 中输出预测和输出实际值的关系曲线

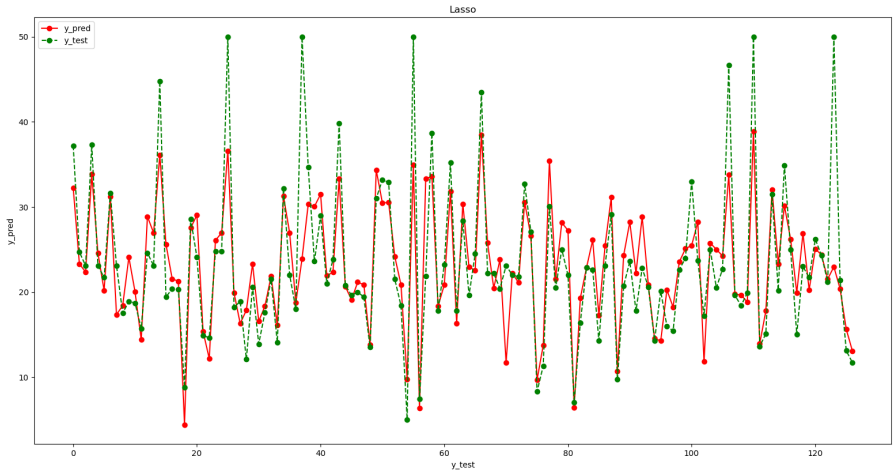

Figure 12: Lasso 中输出预测和输出实际值的关系曲线

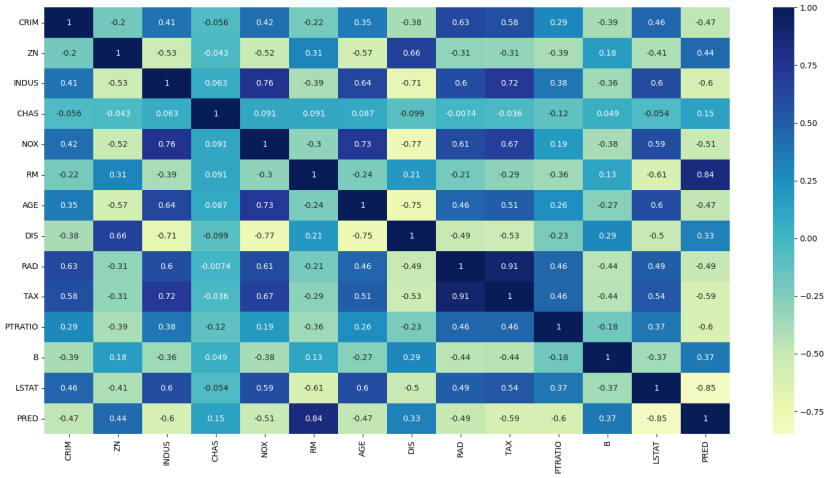

Figure 13: 不同输入特征与输出变量之间的关系强弱和相关性图

b) 如图

```
Ridge_r2_score: 0.6370890428531533
Ridge_mean_squared_error 36.63999874539582
Ridge_mean_absolute_error 3.772288934990351
Lasso_r2_score: 0.6190017441017177
Lasso_mean_squared_error 38.466117771314465
Lasso_mean_absolute_error 4.026564679997734
Poly_r2_score: 0.5039488640629357
Poly_mean_squared_error 50.08201775244782
Poly_mean_absolute_error 4.231661104968214
```

Figure 14: Ridge 和 Lasso 与一般的多元线性回归的结果对比

c) 如图

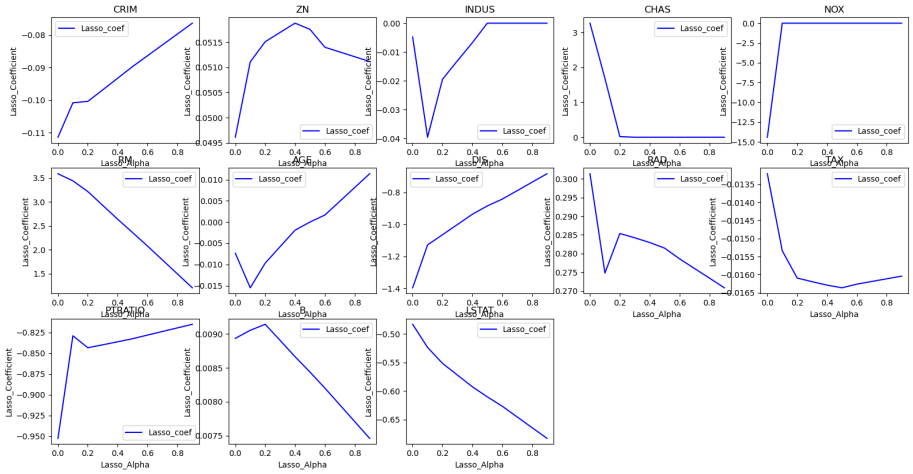

Figure 15: Lasso 回归中不同变量回归系数随 alpha 的变化图

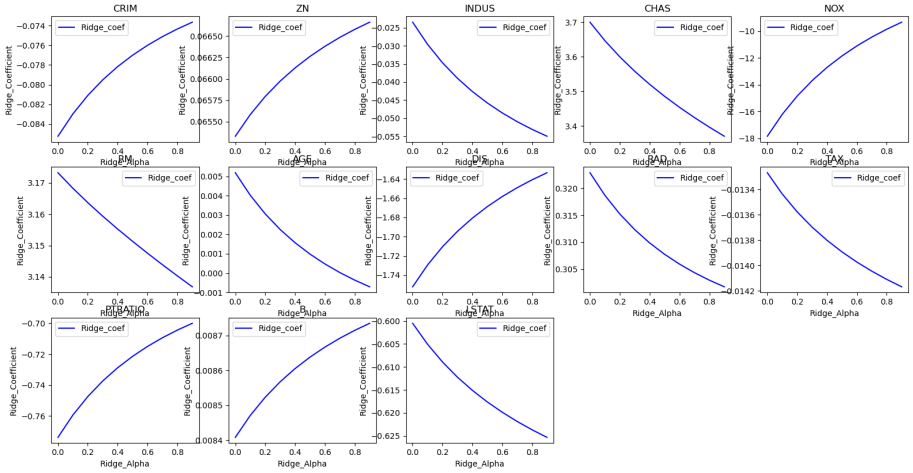

Figure 16: 岭回归中不同变量回归系数随 alpha 的变化图

2 分类

根据手写体数据集，熟悉如何对图像进行分类。

1. 要求及步骤

a) 认识数据集（可视化数据）

b) 参考

运行：http://scikit-learn.org/stable/_downloads/plot_digits_classification.py 给出注释。

c) 用 KNN 分类模型，对手写体数据集进行识别。讨论 k 变化时分类性能变化。

d) 用 SVM(课堂没讲，请直接调用 Scikit-learn 模块) 分类模型，对比与最好 KNN 模型的性能好坏。

答:

a) 如图

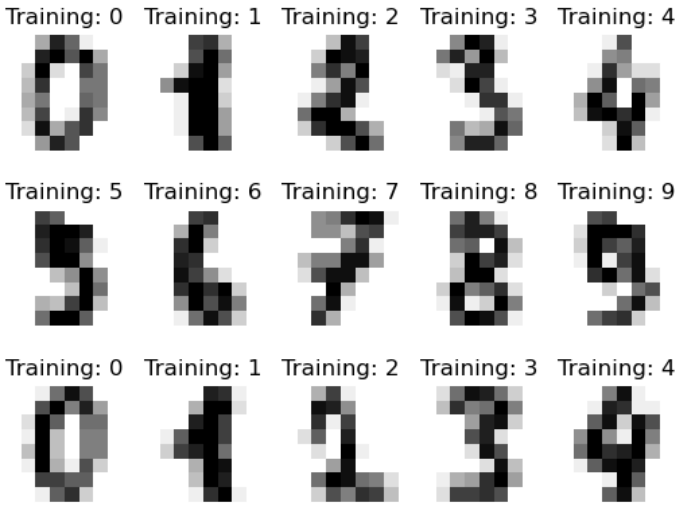

Figure 17: 数据可视化图

b) 详见附件代码，如图为生成图片

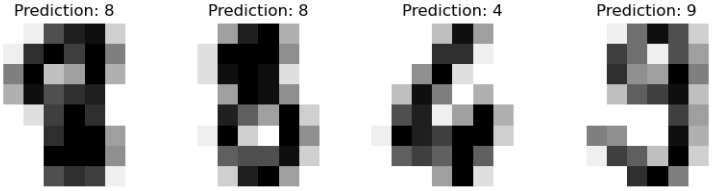

Figure 18: 测试的部分结果

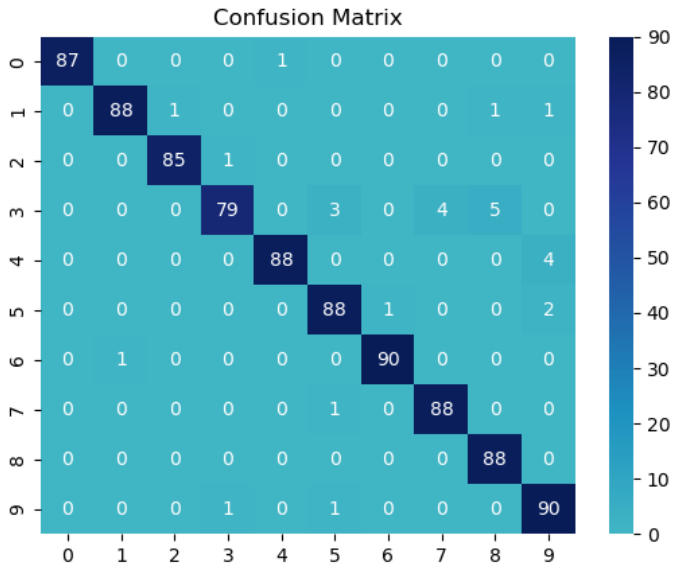

Figure 19: 混淆矩阵

c) 如图

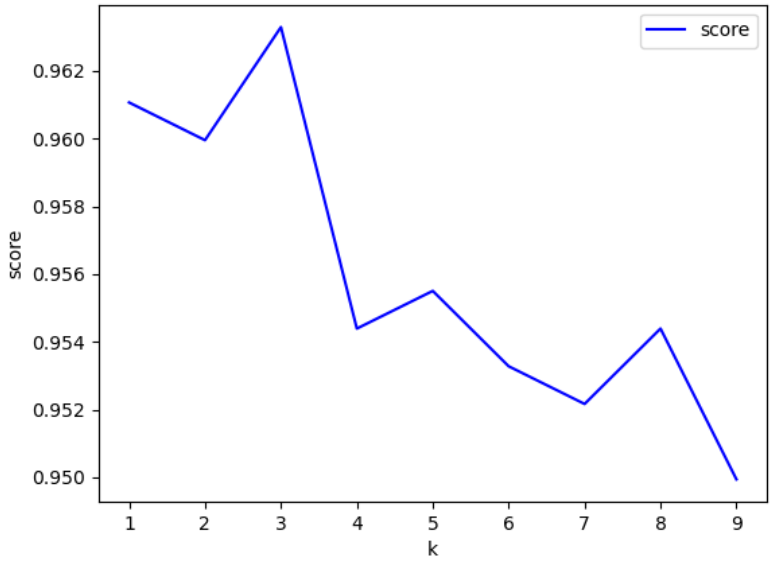

Figure 20: 分类性能随 k 变化曲线

d) 由上一题知，当 k = 3 时，KNN 结果最佳。如图

```
KNN_r2_score: 0.9048265802552166
KNN_mean_squared_error 0.7819799777530589
KNN_mean_absolute_error 0.1546162402669633
SVM_r2_score: 0.9260815260588169
SVM_mean_squared_error 0.60734149054505
SVM_mean_absolute_error 0.12680756395995552
```

Figure 21: SVM 分类模型与最优的 KNN 模型性能对比

3 Perceptron

Task:

Write your own perceptron program (Do not use the off-the-shelf package) and use the given data to check your model

Goal:

We hope that you can use the pandas or other package to read the dataset and make some preprocess. You need to know how the perceptron works and try to achieve it by yourself.

Data:

For CS background students, you should use the attached dataset file. For Non CS background students, you can use the sklearn package to load the MNIST dataset.

Divide the data into train set and test set. Choose two labels and use the corresponding data to finish the experiment. It is not necessary to use the whole data of two labels you choose; you can pad the data or downsample the data for data balance.

Data:

- Load the data and write the code to classify. Train the model with the train set and calculate the metric on the test set.
- Try to improve the performance and give your explanation.
- Try to make your code more readable and better to transfer.

答:

步骤 1：使用 pandas 导入数据 Data.csv，取 data.iloc[:,:5](前 5 列) 作为数据的特征，最后一列 data['fetal_health'] 作为数据的标签。

步骤 2：数据归一化处理，将特征数据的分布映射到 [-1,1] 之间。

步骤 3：按照 9：1 划分数据集，并设置随机种子为 666.

步骤 4：得到数据标签的种类，并同时得到对应的 one_hot 编码

步骤 5：实现感知机模型：初始化权重 w 和 b，设置反向传播参数 alpha 和 beta，配置 num_args 和 num_class 使得代码的应用性更广泛。感知机类实现了训练、评估和测试的方法。

步骤 6：训练 1500 个 epoch，得到权重 w 和 b

步骤 7：载入权重，分析指标

如图：

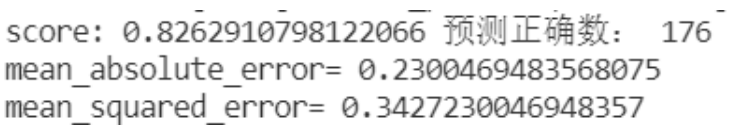

Figure 22: 感知器指标