**Lab 01 วิชา การเรียนรู้ของเครื่องกล (Machine Learning)**

หัวข้อ

1. Ridge regression
2. LASSO regression
3. ElasticNet regression

คำสั่งส่วนที่ 1

1. เปรียบเทียบการทำงานของ Linear Regression, Ridge Regression, Lasso Regression และ ElasticNet Regression
   1. ใช้ชุดข้อมูล Diabetes (คำสั่ง data = load\_diabetes() )
   2. แยกข้อมูล Train และ Test เป็นสัดส่วนตามที่โจทย์กำหนด
2. ใช้ Linear Regression, Ridge Regression, Lasso Regression และ ElasticNet Regression ในการทำนาย โดย
   1. เก็บผล
      1. Mean Squared Error ของทุกโมเดล (MSE ที่วัดได้น้อยกว่าคือดีกว่า)
      2. เก็บค่า alpha (lambda) ของโมเดล Ridge, Lasso และ ElasticNet Regression
   2. บันทึกผลลงในใบงาน (หน้า 2) ไม่ต้องกังวลเกี่ยวกับแบบฟอร์มหรือจำนวนหน้า แม้ว่า ตารางจะตกหน้า แยกออกจากกัน หรือทำให้มีพื้นที่ว่างในบางหน้ามากเกินไป ก็ไม่มีปัญหา
   3. เซฟไฟล์นี้ ตั้งชื่อว่า Lab01\_result\_รหัสนักศึกษา.docx

คำสั่งส่วนที่ 2

1. หาชุดข้อมูลที่สนใจที่จะใช้ในการทำ Assignment
2. เตรียมจัดกลุ่มสำหรับทำ Assignment

**ใบงานส่วนที่ 1** ทำความรู้จักกับชุดข้อมูล

|  |  |
| --- | --- |
| ชุดข้อมูล Diabete มีข้อมูลบรรจุอยู่จำนวน | 442 Samples |
| มิติของข้อมูล (จำนวน Feature) | 10 Features |
| ระบุรายชื่อของ Feature ทั้งหมด | ['age', 'sex', 'bmi', 'bp', 's1', 's2', 's3', 's4', 's5', 's6'] |
| ข้อมูล Target (ที่ต้องการพยากรณ์) เป็นข้อมูลแบบ… (เลือก 1 ข้อ) | ไม่ต่อเนื่อง (Discrete) |
| ใช้คำสั่ง Print และคัดลอกชุดข้อมูล diabete ที่เป็น data ทั้งหมดแล้ววางลงในช่องนี้ >>>>> | array([[ 0.03807591, 0.05068012, 0.06169621, ..., -0.00259226,  0.01990749, -0.01764613],  [-0.00188202, -0.04464164, -0.05147406, ..., -0.03949338,  -0.06833155, -0.09220405],  [ 0.08529891, 0.05068012, 0.04445121, ..., -0.00259226,  0.00286131, -0.02593034],  ...,  [ 0.04170844, 0.05068012, -0.01590626, ..., -0.01107952,  -0.04688253, 0.01549073],  [-0.04547248, -0.04464164, 0.03906215, ..., 0.02655962,  0.04452873, -0.02593034],  [-0.04547248, -0.04464164, -0.0730303 , ..., -0.03949338,  -0.00422151, 0.00306441]]) |
| ใช้คำสั่ง Print และคัดลอกชุดข้อมูล diabete ที่เป็น Target ทั้งหมดแล้ววางลงในช่องนี้ >>>>> | array([151., 75., 141., 206., 135., 97., 138., 63., 110., 310., 101.,  69., 179., 185., 118., 171., 166., 144., 97., 168., 68., 49.,  68., 245., 184., 202., 137., 85., 131., 283., 129., 59., 341.,  87., 65., 102., 265., 276., 252., 90., 100., 55., 61., 92.,  259., 53., 190., 142., 75., 142., 155., 225., 59., 104., 182.,  128., 52., 37., 170., 170., 61., 144., 52., 128., 71., 163.,  150., 97., 160., 178., 48., 270., 202., 111., 85., 42., 170.,  200., 252., 113., 143., 51., 52., 210., 65., 141., 55., 134.,  42., 111., 98., 164., 48., 96., 90., 162., 150., 279., 92.,  83., 128., 102., 302., 198., 95., 53., 134., 144., 232., 81.,  104., 59., 246., 297., 258., 229., 275., 281., 179., 200., 200.,  173., 180., 84., 121., 161., 99., 109., 115., 268., 274., 158.,  107., 83., 103., 272., 85., 280., 336., 281., 118., 317., 235.,  60., 174., 259., 178., 128., 96., 126., 288., 88., 292., 71.,  197., 186., 25., 84., 96., 195., 53., 217., 172., 131., 214.,  59., 70., 220., 268., 152., 47., 74., 295., 101., 151., 127.,  237., 225., 81., 151., 107., 64., 138., 185., 265., 101., 137.,  143., 141., 79., 292., 178., 91., 116., 86., 122., 72., 129.,  142., 90., 158., 39., 196., 222., 277., 99., 196., 202., 155.,  77., 191., 70., 73., 49., 65., 263., 248., 296., 214., 185.,  78., 93., 252., 150., 77., 208., 77., 108., 160., 53., 220.,  154., 259., 90., 246., 124., 67., 72., 257., 262., 275., 177.,  71., 47., 187., 125., 78., 51., 258., 215., 303., 243., 91.,  150., 310., 153., 346., 63., 89., 50., 39., 103., 308., 116.,  145., 74., 45., 115., 264., 87., 202., 127., 182., 241., 66.,  94., 283., 64., 102., 200., 265., 94., 230., 181., 156., 233.,  60., 219., 80., 68., 332., 248., 84., 200., 55., 85., 89.,  31., 129., 83., 275., 65., 198., 236., 253., 124., 44., 172.,  114., 142., 109., 180., 144., 163., 147., 97., 220., 190., 109.,  191., 122., 230., 242., 248., 249., 192., 131., 237., 78., 135.,  244., 199., 270., 164., 72., 96., 306., 91., 214., 95., 216.,  263., 178., 113., 200., 139., 139., 88., 148., 88., 243., 71.,  77., 109., 272., 60., 54., 221., 90., 311., 281., 182., 321.,  58., 262., 206., 233., 242., 123., 167., 63., 197., 71., 168.,  140., 217., 121., 235., 245., 40., 52., 104., 132., 88., 69.,  219., 72., 201., 110., 51., 277., 63., 118., 69., 273., 258.,  43., 198., 242., 232., 175., 93., 168., 275., 293., 281., 72.,  140., 189., 181., 209., 136., 261., 113., 131., 174., 257., 55.,  84., 42., 146., 212., 233., 91., 111., 152., 120., 67., 310.,  94., 183., 66., 173., 72., 49., 64., 48., 178., 104., 132.,  220., 57.]) |

**ใบงานส่วนที่ 2** ทำความรู้จักกับชุดข้อมูล

|  |  |  |
| --- | --- | --- |
| Model: ตัวอย่าง | | |
| Train:test | MSE | Alpha (Lambda) |
| XXX:YYY | 43.012176064834804 | 0.4951412800000001 |

|  |  |  |
| --- | --- | --- |
| Model: Linear | | |
| Train:test | MSE | Alpha (Lambda) |
| 99:1 | 5750.96282763106 | - |
| 95:5 | 9533.126570168464 | - |
| 80:20 | 2653.2259346358373 | - |
| 20:80 | 3172.1271466846943 | - |
| 5:95 | 5151.638002309709 | - |
| 1:99 | 10865.039736849252 | - |

|  |  |  |
| --- | --- | --- |
| Model: Ridge | | |
| Train:test | MSE | Alpha (Lambda) |
| 99:1 | 1049.9139048944414 | 1.0 |
| 95:5 | 4555.831089111439 | 1.21 |
| 80:20 | 2792.157569397645 | 0.49 |
| 20:80 | 3270.5632858009044 | 0.01 |
| 5:95 | 4356.393744402327 | 0.1 |
| 1:99 | 4684.991746426427 | 0.01 |
| 80:20 | 2612.1937760534843 | 0.1 |

|  |  |  |
| --- | --- | --- |
| Model: Lasso | | |
| Train:test | MSE | Alpha (Lambda) |
| 99:1 |  |  |
| 95:5 |  |  |
| 80:20 |  |  |
| 20:80 |  |  |
| 5:95 |  |  |
| 1:99 |  |  |

|  |  |  |
| --- | --- | --- |
| Model: ElasticNet | | |
| Train:test | MSE | Alpha (Lambda) |
| 99:1 |  |  |
| 95:5 |  |  |
| 80:20 |  |  |
| 20:80 |  |  |
| 5:95 |  |  |
| 1:99 |  |  |

**สรุปผลการทดลอง**

|  |
| --- |
| **จากการทดลอง ทดสอบประสิทธิภาพ ระหว่าง model** Linear กับ model Ridge สรุป ได้ว่า model Ridge มีค่า mean squared error ต่ำกว่า model Linear เพราะ มีค่า Lambda ในการปรับเส้น Regression ใน การทดสอบผมว่าการปรับ ค่า Lambda มีผลต่อค่า mean squared error อย่างมาก |

**แนบโค้ด**

|  |
| --- |
| **from sklearn.datasets import load\_diabetes**  **from sklearn.model\_selection import train\_test\_split**  **from sklearn.linear\_model import LinearRegression**  **from sklearn.metrics import mean\_squared\_error**  **from sklearn.linear\_model import Ridge**  **# LinearRegression**  **x = diabetes.data**  **y = diabetes.target**  **x\_train,x\_test,y\_train,y\_test = train\_test\_split(x,y,test\_size = 0.05)**  **model = LinearRegression()**  **# การเรียนรู้**  **model.fit(x\_train,y\_train)**  **# ทดสอบ**  **y\_pred = model.predict(x\_test)**  **mse = mean\_squared\_error(y\_test,y\_pred)**  **print(mse)**  **# Ridge**  **x = diabetes.data**  **y = diabetes.target**  **x\_train,x\_test,y\_train,y\_test = train\_test\_split(x,y,test\_size = 0.8)**  **# การเรียนรู้**  **model = Ridge(alpha=0.01)**  **model.fit(x\_train,y\_train)**  **# ทดสอบ**  **y\_pred = model.predict(x\_test)**  **mse = mean\_squared\_error(y\_test,y\_pred)**  **print(mse)** |