機械学習概論前半レポート

05242628 三田村彰大

概要　レポート問題の問 1 の線形回帰の問題、問 2 の決定木の問題、問 3 の KL ダイバージェンスの問題に取り組んだ。また、問 1 及び問 2 においては Python を用いてプログラミングを行なった。その際コーディングに生成 AI を利用し、またライブラリとして sklearn を利用した。コードは kadai1.ipynb が問 1 に、kadai2.ipynb が問 2 に対応している。

※AI との会話ログ： https://chatgpt.com/share/6a3e634e-1f38-83ee-bb3a-0d9ecdb60c9c

※ 生成 AI の利用について

使用した AI

本課題においては chatGPT5.5 を利用した。（ちょうど実験的に Pro を契約していたため。）

工夫

AI との会話ログのリンクは冒頭に示した通りである。また、本課題においては各課題におけるコードの実装を AI を用いたバイブコーディング的な手法で行なった。特に、どのような API を利用すれば良いかといった点や、書いたコードのデバッグなどにおいて AI を利用した。ただし工夫として、AI が示したコードをそのまま書き写すのではなく、提示されたコードを読んでその仕組みを理解してから一行ずつ自分でコードを書いていった。[*1]

また、レポートの誤字脱字のチェックや内容の確認に AI を利用している。ただしレポートの本文は全て一度自分の言葉で書き、一度レポートが完成してから AI の査読を通すようにしている。[*2]

AI を利用した感想

機械学習のようなライブラリをふんだんに用いるプログラムは AI を利用せずに書くのはなかなか難しいのではと感じた。そもそもどんなライブラリが利用できるのか、関数の引数はどう定めるべきなのかといったことをいちいち調べながらコードを書くのは相当な労力が必要なように思われ、生成 AI が登場する前にこのレポートに取り組んでいたらと思うとゾッとしている。

問 1

1.1

まずデータの前処理を行う。今回データは、年齢や学習時間などの数値データと、性別や親の職業などの categorical なデータに分かれているため、まず categorical なデータを OneHotVector にする必要がある。今回はこれを sklearn の OneHotEncoder で行った。また、データを学習用：テスト用に 8 : 2 の比率で分割し、学習用データを用いてデータを標準化した。

その上で、今回は sklearn の LinearRegression,Lasso,Ridge を使って学習を行う。今目的変数である G3 という期末テストの結果を y_i、特徴量をそれぞれ x_i とするとき、

$$y = w_0 + w_1 x_1 + \cdots + w_D x_D \tag{1}$$

というモデルを考え、損失関数を

$$\mathcal{L} = \frac{1}{2} \sum_{n \in \text{data}} \left(y_n - \mathbf{w}^T \mathbf{x} \right)^2 + \alpha \mathcal{R}(\mathbf{w}) \tag{2}$$

と定める。この時、LinearRegression,Lasso,Ridge それぞれに対し正則化項 $\mathcal{R}(\mathbf{w})$ は次のようになる。

$$\mathcal{R}(\mathbf{w}) = 0 \qquad \text{LinearRegression} \tag{3}$$

$$\mathcal{R}(\mathbf{w}) = \sum_i |w_i| \qquad \text{Lasso} \tag{4}$$

$$\mathcal{R}(\mathbf{w}) = \sum_i w_i^2 \qquad \text{Ridge} \tag{5}$$

すなわち LinearRegression,Lasso,Ridge がそれぞれ正則化なし、L1 正則化、L2 正則化に対応していることがわかる。

その上でまず、（題意とは逸れるが）学習により得られた重みが特に大きくなった特徴量を Table1に示す。

特徴量 i	重み $\lvert w_i \rvert$
failures	1.361698
studytime	0.632581
absences	0.557590
Medu	0.503753
⋮	⋮

Table1: G3 期末テストの結果に対する寄与が特に大きい特徴量

Table1を見ると、生徒の過去の学業的な失敗の回数の多さが他の特徴量に比べても成績への影響が大きいことがわかる。また、母親の学歴なども上位に食い込む結果となっている。[*3]

[*1] そうしないとプログラミング力が鈍ってしまう気がしたので…
[*2] やはり、文章力が衰えるのが怖いので…
[*3] これは意外だった。

さて、今回正則化のモデル及びその強さ α をさまざまに変えた際の MSE の大きさを Figure1に示す。

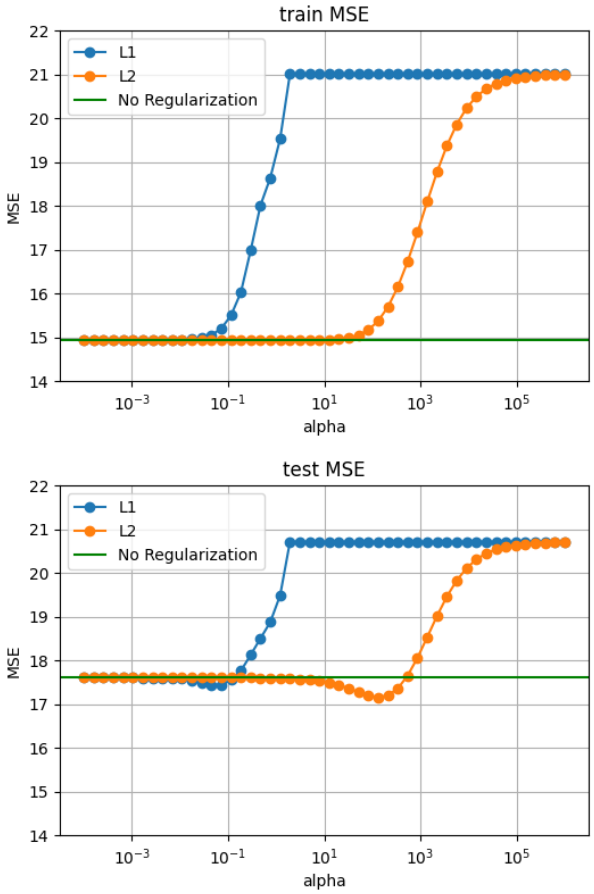

Figure1: 上図：学習用データに対する MSE（TrainMSE）のグラフ。横軸は正則化強度 α を、色の違いは正則化モデルの違いを表している。青線（L1）が Lasso 正則化、オレンジ（L2）が Ridge 正則化、緑線（No Regularization）が正則化なしに対応している。下図：テスト用データに対する MSE（TestMSE）のグラフ。

1.2

次に、それぞれの正則化強度・モデルに対し、学習後の重みが非ゼロ（数値的には $\geq 10^{-6}$）となる特徴量の数を Figure2に示す。

この時、OneHotVector 化後の特徴量の総数は 56 であり、Figure2の緑線及びオレンジ線の値も 56 となっている。すなわち L2 正則化及び正則化なしの場合においては全ての特徴量が非ゼロとなっている。

1.3

Figure1及び Figure2の観察から次のようなことがわかる。

- 全般的に、テスト用データを用いた場合の MSE は訓練用データを用いた場合のそれに比べて悪化する。
- L1,L2 いずれの正則化においても、正則化強度が弱い（α が小さい）と MSE が小さく、正則化強度が強いと MSE が大きくなる。
- L1,L2 いずれの正則化においても、適切な正則化強度のもとではテスト用データに対する MSE は正則化がない場合に比べて小さくなっている。
- L2 正則化及び正則化なしの場合においては全ての重みが非

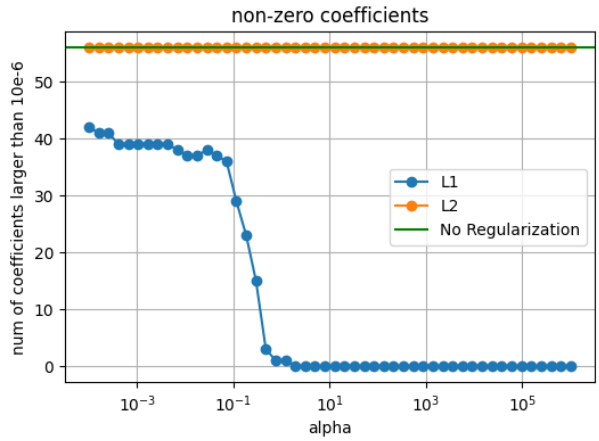

Figure2: 学習後の重みの大きさが非ゼロ（$\geq 10^{-6}$）となる特徴量の数。横軸およびグラフの色分けは正則化モデル・正則化強度の違いを表しており、Figure1と同様。縦軸が各モデルにおいて非ゼロとなる特徴量の数を表している。

ゼロとなっている。

- L1 正則化においては、正則化強度を強くするほど非ゼロの特徴量が減っていく。

1.4

先に述べた通り、L1 正則化においては正則化強度を強くするほど非ゼロの特徴量が減っていく（重みがゼロになる特徴量が多く現れる）ことが見て取れる。これは Lasso がスパースな解を誘導するからに他ならない。すなわち Lasso においては最適値が軸上に位置する可能性が高くなり、重みがゼロとなる特徴量が現れることになる。(Figure3参照。)

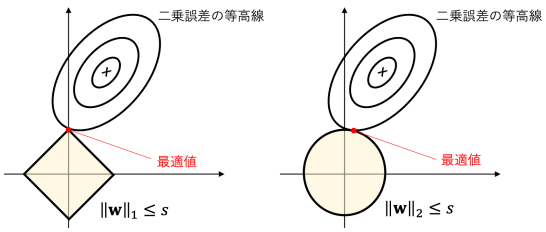

Figure3: lasso におけるスパース解の誘導。授業スライドより。

また、α 大で非ゼロの特徴量が減るのは正則化を強くすると Lasso の上記の性質が強く現れることを示している。

他にも、テスト用データを用いた場合の MSE が訓練用データを用いた場合のそれに比べて悪化するという結果が見られたが、これは過学習傾向が生じている可能性を示している。その上で、適切な正則化強度 α のもとでテスト用データに対する MSE が正則化がない場合に比べて小さくなっていることは、正則化により過学習を抑えられていることを示唆している。ただし正則化強度が強すぎると訓練テスト問わず MSE が大きくなってしまっており、これは最適化がほとんど正則化項に対して行われておりデータに対して重みを最適化できなくなっているためであると思われる。

1.5

今回用いた L1,L2 以外の正則化手法としては elastic-net などを考えることができる。(参考：[1]) これは Lasso 回帰と Ridge 回帰を組み合わせたような手法であり、損失関数は次のように表される。

$$\mathcal{L} = \frac{1}{2}\sum_{n\in\text{data}}\left(y_n - \mathbf{w}^T\mathbf{x}\right)^2 + \lambda_1\mathcal{R}_{L1}(\mathbf{w}) + \lambda_2\mathcal{R}_{L2}(\mathbf{w}) \quad (6)$$

このとき、ハイパーパラメタ λ_1, λ_2 を調整することにより正則化の強さと、L1/L2 正則化のバランスを決定する。

Elastic-net を採用する理由としては、Lasso 回帰と Ridge 回帰それぞれが持つ長所を両方織り込めるという点が挙げられる。今、Lasso 回帰はスパースな解を誘導することができるが、相関する回帰変数があるとそれらを一つを除いて 0 にしてしまうという問題がある。このとき逆に Ridge 回帰においてはこのような回帰変数群を近い値に推定することができ、これはグループ効果と呼ばれる。Elastic-net はこのような Lasso 回帰のもつスパース性と Ridge 回帰の持つグループ効果の両方を織り込んだ回帰をすることができると期待される。

問 2

2.1

次に、決定木を用いた回帰を行う。今回も問 1 の場合と同様、categorical なデータを sklearn の OneHotEncoder を用いて One-HotVector 化する。またデータを訓練用：テスト用 = 8 : 2 に分割するが、今回は random seed を 100 通り変化させることで 100 通りに分割の仕方を試すことにする。

また、今回は sklearn の DecisionTreeRegressor, BaggingRegressor, RandomForestRegressor を用いて学習を行なった。(それぞれが決定木・Bagging・Random Forest に対応している。) 今、100 通りの分割に対しこれら三つのモデルで学習を行なって得られる TestMSE の分布を Figure4に示す。

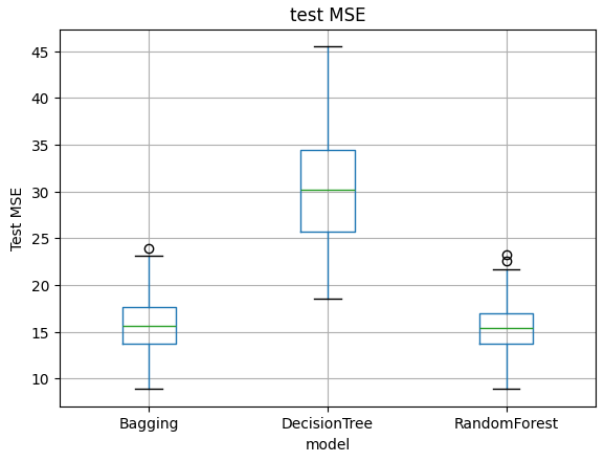

Figure4: 左から、Bagging、決定木、Random Forest モデルにおける TestMSE の分布。

Figure4を見ると、Bagging 及び RandomForest において MSE が単純な決定木回帰の場合よりも小さくなっていることがわかる。

2.2

次に、Bagging と Random Forest において決定木の本数 B を変えることで予測値の分散がどう変化するかを観察する。今、それぞれの B の値に対して学習を 50 回行い、予測値の分散をプロットした図が Figure5である。ただし Random Forest の max features= 0.6 としている。

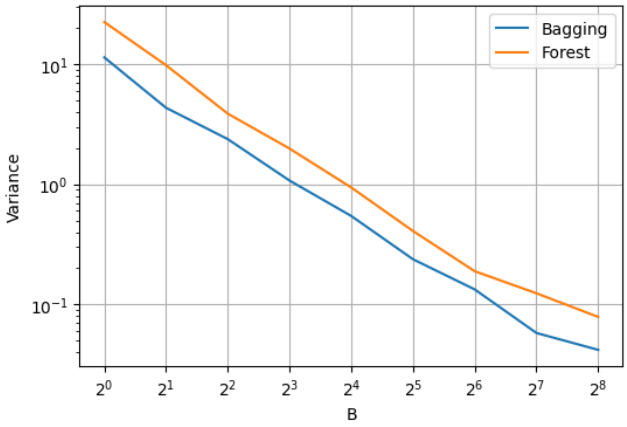

Figure5: 決定木の本数を変えた時の予測値の分散の様子。

2.3

2.1 及び 2.2 の結果から次のことがわかる。

- 決定木に比べて Bagging や Random Forest で TestMSE が小さくなっている。
- Bagging に比べて Random Forest の方が TestMSE のばらつきが小さく結果が安定している。
- Bagging・Random Forest ともに決定木の本数 B に対する予測値の分散の両対数プロットが直線に近くなっている。

2.4

今回、決定木に比べて Bagging や Random Forest で TestMSE が小さくなっていることはアンサンブル学習の特性から説明できる。一般に、決定木は柔軟なモデルである代わりに不安定性が大きく過学習を起こすなどの問題が生じやすい。この時決定木を組み合わせることでアンサンブル学習を行うとこの不安定性を改善することができ、今回の Bagging や Random Forest の TestMSE の結果につながるような高い汎化性能を実現できると考えることができる。

また今、Bagging 及び Random Forest における予測値の分散は理論的に次のように計算することができる。(参考：[2]) まず Bagging についてだが、今一本の決定木の予測値が分散 σ^2 を持ち、決定木同士の相関係数がおおよそ ρ 程度であると考えると、最終的に決定木間の平均として与えられる予測値 $\overline{X}$ の分散は次のようになる。

$$\begin{aligned}
\text{Var}\left(\overline{X}\right) &= \text{Var}\left(\frac{X_1 + X_2 + \cdots X_B}{B}\right) && (7)\\
&= \frac{1}{B^2}\sum_i \text{Var}\left(X_i\right) + \frac{1}{B^2}\sum_{i<j} 2\,\text{Cov}\left(X_i, X_j\right) && (8)\\
&= \frac{1}{B^2}\left(B\sigma^2\right) + \frac{1}{B^2}B\left(B-1\right)\rho\sigma^2 && (9)\\
&= \sigma^2\left(\rho + \frac{1-\rho}{B}\right) && (10)
\end{aligned}$$

次に Random Forest についてだが、この場合においては採用す

る特徴量をランダムに選択するため異なる決定木間の予測値の相関係数を小さく、理想的には $\rho = 0$ とすることができると考えることができ、よって予測値 $\overline{X}$ の分散は（10）式で $\rho = 0$ とすることを考えると次のようになると考えることができる。

$$\mathrm{Var}\left(\overline{X}\right) = \frac{\sigma^2}{B} \tag{11}$$

この時、$\rho \geq 0$ から

$$\rho \geq 0 \Leftrightarrow \underbrace{\frac{\sigma^2}{B}}_{\mathrm{Var\_Random\_Forest}} \leq \underbrace{\sigma^2\left(\rho + \frac{1-\rho}{B}\right)}_{\mathrm{Var\_Bagging}} \tag{12}$$

が従うので、予測値の分散は Bagging より Random Forest で小さくなるはずである。これが、Bagging に比べて Random Forest の方が TestMSE のばらつきが小さく結果が安定している理由であると思われる。ただし Figure5の両対数プロットを見ると予測値の分散は Bagging の方が小さくなっており、上記の理論と整合性が取れない。これは、Random Forest によるランダムな特徴量選択によって決定木どうしの相関 ρ が下がっていたとしても、それ以上に使う特徴量の数を制限したことによって各決定木における予測値の分散 σ^2 が大きくなってしまっている可能性がある。

また、両対数プロットは理論的には、

$$\begin{aligned}&\log \mathrm{Var}\left(\overline{X}_{\mathrm{Bagging}}\right)\\ &\quad = -\log B + \left\{\log \sigma^2 + \log\left(\rho B + 1 - \rho\right)\right\} \end{aligned} \tag{13}$$

$$\begin{aligned}&\log \mathrm{Var}\left(\overline{X}_{\mathrm{Random\_Forest}}\right)\\ &\quad = -\log B + \log \sigma^2 \end{aligned} \tag{14}$$

のようになるはずである。すなわち Random Forest においては両対数プロットは直線になることが予測されるが、Bagging においては $\log\left(\rho B + 1 - \rho\right)$ の分だけ直線からのずれが生じることが期待される。（しかし Figure5を見ると、Bagging も Random Forest と同じぐらい直線に近くなっていることがわかる。これは決定木同士の相関 ρ が小さいことなどが理由として考えられる。）

問 3

3.1

情報エントロピー及び KL ダイバージェンスの定義は次のよう。

$$S\left(p\right) = -\sum_x p\left(x\right)\log p\left(x\right) \tag{15}$$

$$\mathrm{KL}\left(p||q\right) = \sum_x p\left(x\right)\log\frac{p\left(x\right)}{q\left(x\right)} \tag{16}$$

3.2

KL ダイバージェンスの $p\left(x\right)$ にボルツマン分布 $p\left(x\right) = \frac{e^{-\beta E(x)}}{Z}$ を代入すると、次のように計算することができる。

$$\begin{aligned}\mathrm{KL}\left(q||p\right) &= \sum_x q\left(x\right)\left(\log q\left(x\right) - \log\frac{e^{-\beta E(x)}}{Z}\right) &(17)\\ &= \sum_x q\left(x\right)\log q\left(x\right) + \beta\sum_x E\left(x\right)q\left(x\right)\\ &\qquad + \log Z\sum_x q\left(x\right) &(18)\\ &= -S\left(q\right) + \beta\langle E\rangle_q + \log Z &(19)\end{aligned}$$

ただし $\langle E\rangle_q$ は状態が確率変数 q で表される時のエネルギーの期待値を表す。よって今自由エネルギーに関する公式 $F[p] = -\frac{1}{\beta}\log Z$ 及び $F[q] = \langle E\rangle_q - TS\left(q\right)$ を用いれば、（19）式は次のように変形できる。

$$\begin{aligned}\mathrm{KL}\left(q||p\right) &= \beta\left(\langle E\rangle_q - \frac{1}{\beta}S\left(q\right)\right) - \beta F[p] &(20)\\ &= \beta\left(F[q] - F[p]\right) &(21)\end{aligned}$$

よって確かにこの時 KL ダイバージェンスは自由エネルギーの差分の形になっている。

3.3

今、$x > 0$ に対し $\ln x \leq x - 1$ であるから、これを変形して以下を得る。

$$\ln x \geq 1 - \frac{1}{x} \tag{22}$$

これを用いれば KL 情報量は次のように変形することが可能である。

$$\begin{aligned}\mathrm{KL}\left(q||p\right) &\coloneqq \sum_x q\left(x\right)\ln\frac{q\left(x\right)}{p\left(x\right)} &(23)\\ &\geq \sum_x q\left(x\right)\left(1 - \frac{p\left(x\right)}{q\left(x\right)}\right) &(24)\\ &= \sum_x q\left(x\right) - \sum_x p\left(x\right) &(25)\\ &= 1 - 1 &(26)\\ &= 0 &(27)\end{aligned}$$

よって $\mathrm{KL}\left(q||p\right) \geq 0$ は示される。

3.4

3.2 の結果より

$$F[q] = F[p] + \frac{1}{\beta}\mathrm{KL}\left(q||p\right) \tag{28}$$

となっているから、KL ダイバージェンスはボルツマン分布からの自由エネルギーのずれを表していると解釈することができる。特に 3.3 より KL ダイバージェンスは $\mathrm{KL}\left(q||p\right) \geq 0$ を満たすので、q がボルツマン分布となるときこれは最小となる。

参考文献

[1] https://www.orist.jp/technicalsheet/23-06.pdf

[2] http://www.sakurai.comp.ae.keio.ac.jp/classes/infosem-class/2017/11BaggingAndRF.pdf