APRENDIZAGEM DE MÁQUINA

PROF. JOSENALDE OLIVEIRA

josenalde.oliveira@ufrn.br
https://github.com/josenalde/machinelearning

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - UFRN

QUESTION & ANSWER (Q&A) - REGRESSÃO

1. Qual algoritmo de treinamento de regressão linear podemos utilizar se tivermos um conjunto de treinamento com milhões de features?

R: Se tiver um conjunto de treinamento com milhões de features, você poderá usar um gradiente descendente estocástico ou um gradiente descendente em mini-batch e talvez um gradiente descendente batch se o conjunto de treinamento couber na memória. Mas você não pode usar a equação normal ou a abordagem SVD, porque a complexidade computacional cresce rapidamente (mais do quadraticamente) com o número de features

QUESTION & ANSWER (Q&A) - REGRESSÃO

2. Suponha que as características do seu conjunto de treinamento tenham escalas muito diferentes. Quais algoritmos podem sofrer com isso e como? O que você pode fazer a respeito?

R: Caso as características em seu conjunto de treinamento tenham escalas muito diferentes, a função de custo terá o formato de uma tigela alongada, de modo que os algoritmos de gradiente descendente levarão muito tempo para convergir. Para solucionar, você deve escalonar os dados antes de treinar o modelo. A equação normal ou SVD funciona bem sem escalonamento. Além disso, os modelos regularizados podem convergir para uma solução abaixo do ideal se as features não forem escalonadas: uma vez que a regularização penaliza grandes pesos, features com valores menores costumam ser ignoradas em comparação com features de valores maiores.

QUESTION & ANSWER (Q&A) - REGRESSÃO

3. O gradiente descendente pode ficar empacado em um mínimo local, quando trainamos um modelo de regressão logística?

R: O gradiente descendente não pode ficar empacado em um mínimo local ao treinar um modelo de regressão logística porque a função de custo é convexa, ou seja, se você desenhar uma linha reta entre quaisquer dois pontos da curva, a linha nunca atravessará a curva.

QUESTION & ANSWER (Q&A) - REGRESSÃO

4. Todos os algoritmos GD resultam no mesmo modelo, contando que você os execute por tempo suficiente?

R: Se o problema de otimização for convexo (como regressão linear ou regressão logística) e assumindo que a taxa de aprendizado não seja muito alta, todos os algoritmos de gradiente descendente se aproximarão do ideal global e acabarão gerando modelos bastante semelhantes. No entanto, a menos que você reduza gradativamente a taxa de aprendizado, o GD estocástico e o GD batch nunca convergirão; em vez disso, eles continuarão pulando para frente e para trás em torno do ideal global. Isso significa que, mesmo se você deixa-lo rodar por muito tempo, esses algoritmos de GD gerarão modelos ligeiramente diferentes.

QUESTION & ANSWER (Q&A) - REGRESSÃO

5. Vamos supor que você utilize GD batch e plote o erro de validação a cada época. Caso perceba que o erro aumenta constantemente, o que provavelmente está acontecendo? Como consertar isso?

R: Se o erro de validação aumentar consideravelmente após cada época, uma possibilidade é que a taxa de aprendizado esteja muito alta e o algoritmo esteja divergindo. Se o erro de treinamento também aumentar, isso se torna claramente um problema e você deve reduzir a taxa de aprendizagem. Contudo, se o erro de treinamento não aumentar, seu modelo está se sobreajustando ao conjunto de treinamento e você deve interromper o treinamento.

QUESTION & ANSWER (Q&A) - REGRESSÃO

6. É uma boa interromper o GD mini-batch logo que o erro de validação aumentar?

R: Devido à natureza aleatória, nem o SGD nem o GD mini-batch representam garantia de progresso em cada iteração de treinamento. Ou seja, caso pare de treinar imediatamente quando o erro de validação aumentar, você pode parar muito cedo, antes que o ideal seja alcançado. A melhor opção é salvar o modelo em intervalos regulares; assim, quando não tiver melhorado por muito tempo (o que significa que provavelmente nunca baterá o recorde), você pode reverter para o melhor modelo salvo.

QUESTION & ANSWER (Q&A) - REGRESSÃO

7. Qual algoritmo GD chegará mais rápido e próximo da solução ideal? Qual deles convergirá? Como você pode fazer para os outros convergirem também?

R: O SGD apresenta iteração de treinamento mais rápida, pois considera somente uma instância de treinamento por vez. Logo, ele geralmente é o primeiro a atingir a vizinhança do ideal global (ou GD mini-batch com batch muito pequeno). No entanto, apenas o GD batch realmente convergirá, se houver tempo de treino suficiente.

QUESTION & ANSWER (Q&A) - REGRESSÃO

8. Imagine que esteja usando regressão polinomial. Você plota as curvas de aprendizado e percebe que existe uma lacuna enorme entre erro de treinamento e erro de validação. O que está acontecendo? Quais são as três maneiras de resolver isso?

R: Se o erro de validação for muito maior que o erro de treinamento, é provável que seu modelo esteja se sobreajustando ao conjunto de treinamento. Uma forma de tentar corrigir isso é reduzir o grau polinomial: um modelo com menos graus de liberdade tem menos probabilidade de sobreajuste. Outra coisa que você pode tentar é regularizar o modelo, por exemplo adicionando Ridge (l2) ou Lasso (l1) à função de perda. Isso também reduz os graus de liberdade. Por último, pode tentar aumentar o tamanho do conjunto de treinamento.

QUESTION & ANSWER (Q&A) - REGRESSÃO

9. Suponha que você esteja usando a regressão Ridge e note que o erro de treinamento e o erro de validação são quase iguais e bastante altos. Você diria que o modelo tem um viés alto ou uma variância alta? Você deve aumentar o hiperparâmetro alpha da regularização ou reduzi-lo?

R: Caso o erro de treinamento e o erro de validação forem quase iguais e altos, o modelo estará se sobreajustando ao conjunto de treinamento, o que significa que ele tem um viés alto. Pode tentar reduzir o hiperparâmetro de regularização.

QUESTION & ANSWER (Q&A) - REGRESSÃO

10. Por que você usaria?
a) Regressão Ridge ao invés de regressão linear simples (sem regularização)
b) Regressão Lasso, em vez de Ridge
c) Regressão Elastic Net, em vez de regressão de Lasso?

R: a) um modelo com um pouco de regularização normalmente tem um desempenho melhor do que um modelo sem qualquer regularização. Assim, talvez você prefira a regressão Ridge à simples.
b) A regressão Lasso usa penalidade l1, que costuma empurrar os pesos para exatamente zero. Isso resulta em modelos esparsos, em que todos os pesos são zero, exceto os pesos mais importantes. Essa é uma forma de seleção automática de features, o que é bom se você suspeitar que apenas algumas features importam. Quando não tiver certeza, prefira Ridge.
c) Via de regra, prefere-se elastic net à lasso, já que a última pode se comportar de forma errática em alguns casos (quando várias features estão fortemente correlacionadas ou quando há mais características do que instâncias de treinamento). No entanto, ela adiciona um hiperparâmetro extra para ajustar. Caso prefira a regressão Lasso sem o comportamento errático, use elastic net com l1_ratio próximo de 1

QUESTION & ANSWER (Q&A) - REGRESSÃO

11. Vamos supor que queira classificar as fotos como externas/internas E diurnas/noturnas. Você deve implementar dois classificadores de regressão logística ou um classificador de regressão softmax?

R: uma vez que não são classes exclusivas (ou seja, todas as quatro combinações são possíveis), basta treinar dois classificadores de regressão logística

QUESTION & ANSWER (Q&A) - REGRESSÃO

12. Implemente um GD batch com uma early stopping para a regressão softmax (sem usar scikit-learn)

R: https://github.com/ageron/handson-ml2/blob/master/04_training_linear_models.ipynb