MAC5832 – Introdução ao Aprendizado de Máquina

Lista 3

Aluno: Ciro B Rosa

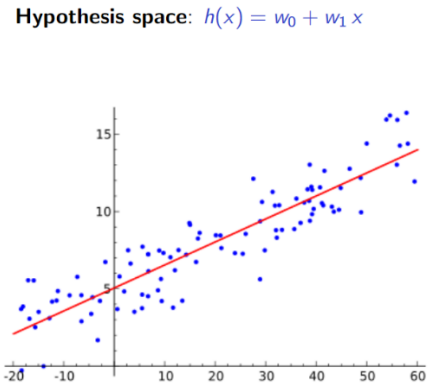
No USP: 2320769

E-mail: ciro.rosa@alumni.usp.br

Data: 20/07/2021

Questão 1

A regressão linear consiste na obtenção de equação de reta (espaço de hipóteses) que melhor se adeque à coleção de pontos observados, como no exemplo a seguir. Desta forma, a inferência sobre pontos adicionais seria feita de modo que estes obedeçam à equação desta reta.



No caso de regressão polinomial, o espaço de hipóteses muda para uma equação polinomial do tipo:

$$h(x) = w_0 + w_1 x + w_2 x^2 + \cdots$$

Esta equação seria então utilizada para a minimização da função de custo, de forma semelhante à regressão linear.

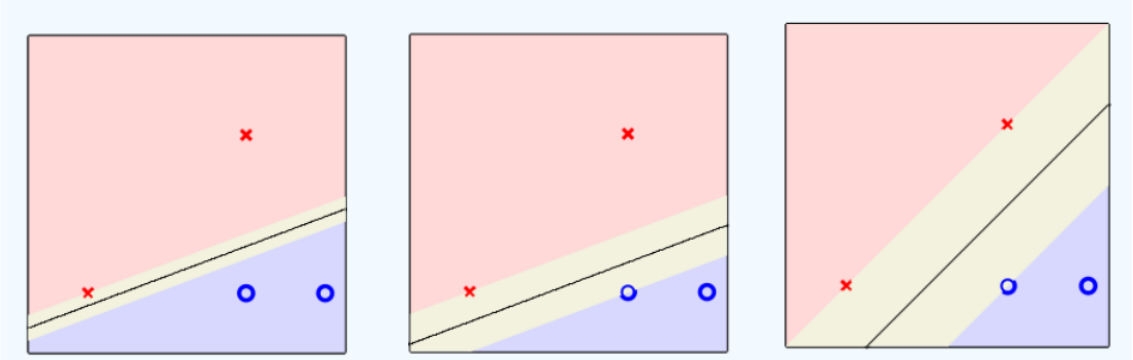
$$J(w) = \frac{1}{N}\sum_{n=1}^{N}[h(x^{(n)}) - y(x^{(n)})]^2$$

Questão 2

A primeira diferença observada entre a Regressão Logística e o SVM é que a primeira não dá como resposta direta a divisão do conjunto de pontos em classes (verdadeiro/falso, positivo/negativo, etc), mas sim uma probabilidade (variável contínua entre [0; 1]) de que determinado ponto pertença a uma certa classe. Já no SVM, a separação em classes é a resposta final.

Vale a ressalva, no entanto, de que a Regressão Logística pode ser usada para classificação, contanto que um parâmetro adicional, o limiar de probabilidade, seja definido. Por exemplo: para probabilidades menores ou iguais a 0,7, pontos seriam classificados como negativos. Já acima de 0,7, como positivos.

Uma segunda consideração diz respeito à forma com que o SVM seleciona o "melhor" hiperplano de separação de classes. Seu critério matemático procura maximizar a distância de cada ponto próximo ao hiperplano de classificação, conforme exemplificado na figura a seguir. Neste exemplo, os três hiperplanos separam os pontos em dicotomias idênticas, porém a última possui a maior margem de ruído (em amarelo).



Por fim, vale ressaltar a similaridade de implementação das duas funções no Python com o Scikit-Learn, consistindo em três etapas:

- Criação de instância do modelo;
- Geração do modelo;
- Geração de respostas Y previstas pelo modelo.

Como diferença na execução, o SVM se utiliza de função denominada de PCA (Principal Component Analysis), transformação linear de coordenadas objetivando uma melhor análise da variabilidade de cada componente, além de redução dimensional da representação dos classificadores X, antes do treinamento propriamente dito do modelo.

Questão 3

Com base no aprendizado em aula e exercícios em classe, principalmente durante o EP4, duas características se destacaram:

- Ambos os modelos obtiveram uma performance (f1-score) tanto elevadas (acima de 0,97) quanto semelhantes. O SVM foi mais eficiente no quesito de "Custo computacional", principalmente durante o treinamento do modelo.
- Por outro lado, o SVM necessariamente demanda pelo conhecimento de um conjunto de pontos (X, y) para treinamento de modelo. Já as redes neurais podem prescindir do conhecimento prévio do resultado y para execução de treinamento de modelo (neste caso, treinamento do tipo não supervisionado).

Questão 4

De forma genérica, um certo conjunto de dados é dividido em subconjuntos para Treinamento e Teste de um dado modelo de Machine Learning. Os dados de treinamento são utilizados para geração do modelo, sendo de certa forma "incorporados" por este. Para que a verificação do modelo seja feita de forma independente, utiliza-se o conjunto de dados de Teste. Deve-se ressaltar que o conjunto de testes só é apresentado ao modelo gerado nesta última e final etapa.

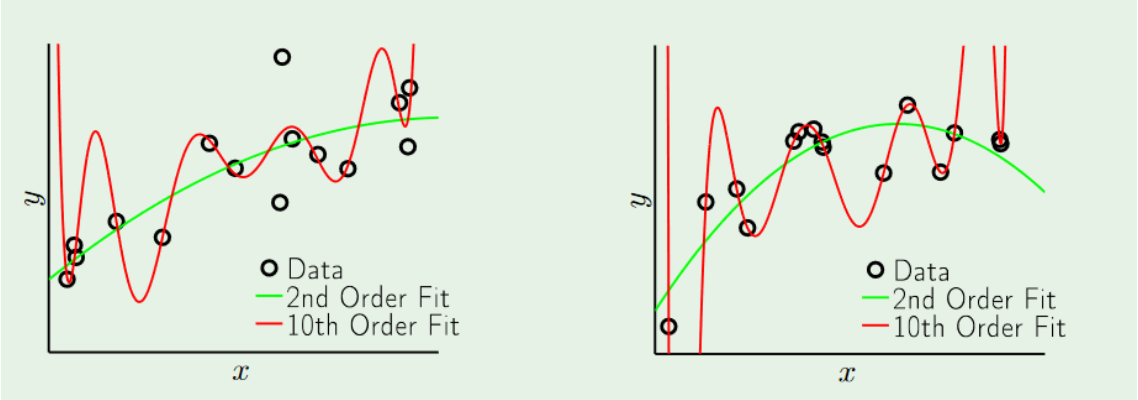
Ocorre que a etapa de treinamento demanda pela sintonia de vários hiperparâmetros do modelo. Em outras palavras, cada conjunto de hiperparâmetros irá, em teoria, gerar um modelo diferente. O projetista deve, portanto, executar uma série de experimentos para obtenção de modelos com diferentes ajustes e medição de performance. Para este objetivo, a sintonia dos hiperparâmetros, os dados de treinamento são subdivididos entre Treinamento (mesmo nome que antes, porém de menor tamanho) e Validação. Desta forma, os hiperparâmetros são sintonizados com os dados de Treinamento e Validação.

Uma vez escolhido o melhor modelo dentre as várias possibilidades de hiperparâmetros, o conjunto de Testes entra em cena, como última etapa e como um conjunto de dados nunca "visto" anteriormente pelo modelo escolhido.

Em resumo, Validação é o conjunto de dados usados para sintonia de hiperparâmetros e escolha do melhor modelo, e Testes é o conjunto de dados para confirmação de desempenho deste melhor modelo.

Questão 5

Os exemplos a seguir ilustram o conceito de overfitting. Em ambas as figuras, as curvas em verde se ajustam de forma aproximada ao conjunto de dados. Já as curvas em vermelho, passam exatamente em todo o conjunto de dados – ou, pelo menos, em sua grande maioria. As curvas em vermelho são exemplos de overfitting de determinado modelo a um conjunto de dados.



Ocorre que a coleta de dados é sujeita a incertezas e ruídos. Dados são essencialmente variáveis aleatórias, medições. Sendo assim, um modelo com overfitting ajusta-se com precisão a dados ruidosos. Quando novos dados são apresentados, o modelo pode simplesmente ser completamente inadequado.

Um modelo com overfitting apresenta Eout significativamente maior que Ein. Vale lembrar que Ein é o erro observado na predição de resultados com dados da própria amostra, enquanto Eout é o erro de todo o universo de dados possíveis. Na prática, Eout é estimado através do uso de dados independentes (não utilizados no treinamento do modelo).

Duas técnicas são passíveis de uso para o combate do overfitting: a Regularização e a Validação.

A Regularização consiste em diminuir a complexidade do vetor w de pesos. Já a validação consiste em dividir o dataset para projeto em Treinamento, Validação e Testes, da forma já discutida anteriormente, cada um com os respectivos propósitos:

- Para treinamento do modelo;
- Para ajuste de hiperparâmetros. Importante notar que, embora não utilizados diretamente para treinamento do modelo, o modelo “enxerga” estes dados durante esta fase – fato este que os retira da classificação de dados de Teste os quais não devem ser utilizados até a última etapa de verificação.
- Para verificação final do modelo sintonizado quanto a desempenho e overfitting.

Questão 6

- No total, julgo ter tido uma boa compreensão de 80% do assunto coberto.
- Destaques sobre tópicos onde ocorreram maiores avanços na compreensão: toda a parte teórica (fundamentos do perceptron, desigualdade de Hoeffding, VC bound, SVM, PCA, redes de perceptrons e redes neurais. Destaque também para os exercícios práticos, como melhor forma de consolidar a teoria.
- Partes que necessitarão de uma nova leitura/aprofundamento: regularização (principalmente) e validação, e ocasionalmente o VC bound.
- Grau de aproveitamento: acima de 8.

Questão 7

- Em todas as aulas, assisti previamente o material do Prof Mostafa. Após cada aula, leitura do livro-texto. Para responder as listas, consulta às notas de aula conforme necessário.
- EPs entregues: 1 a 4. EP a fazer como parte da pesquisa: EP-5.
- QTs não entregues: QT-4 apenas.
- Listas não entregues: Lista 1 (Listas 2 e 3 entregues).
- Justificativa para não entregar QT4 e L1: acúmulo de tarefas.
- Minha frequência em aula foi superior a 90%, sendo que faltas foram compensadas pelas aulas gravadas.

Questão 8

Meus objetivos pessoais foram atingidos, já que consegui obter boa quantidade do conhecimento teórico que me faltava sobre o assunto.

Destaques positivos para as aulas gravadas, recurso inestimável para recapitular de forma rápida conceitos mais complexos, e para os EPs, todos arrematando com precisão a parte teórica.

Pontos a melhorar: talvez pensar em um trabalho prático para, por exemplo, estimar o tamanho da base de dados necessária para treinamento minimamente viável de modelos, com base na desigualdade de Hoeffding, VC bound, etc. A definição do tamanho da amostra para treinamento de modelo com base em dados a serem coletados em campo é de grande importância para estimativa do esforço e custo associados.