Exercícios Deep Learning
Aula 14

October 17, 2019

1 Tarefas supervisionadas e não supervisionadas

1- Qual a diferença entre aprendizado supervisionado e não supervisionado? Para cada uma das definições, cite um exemplo de aplicação.

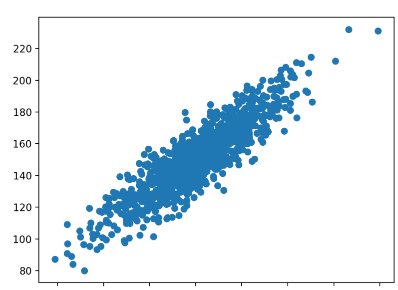
2 PCA

2- Seja $\rho \in (0, 1)$. Mostre que a matriz de covariância

$$\Sigma = \begin{bmatrix} 1 & \rho & \rho \\ \rho & 1 & \rho \\ \rho & \rho & 1 \end{bmatrix}$$

do vetor aleatório $\mathbf{X} = (X_1, X_2, X_3)'$ possui um autovetor igual a $\mathbf{v} = (1, 1, 1)/\sqrt{3}$. Qual o autovalor associado com este autovetor?

3- Considere a distribuição mostrada no gráfico abaixo. Esboce aproximadamente o vetor-direção do primeiro e segundo componente principal.



3 Autoencoder

4- Qual é a ideia principal de um autoencoder? Explique os componentes que fazem parte da arquitetura de um autoencoder e como ele é treinado.

5- Descreva três possibilidades de aplicações utilizando autoencoders.

6- Considere um autoencoder linear (sem regularização ou denoising). O que acontece se o número de nós da camada "bottleneck" (espaço latente) for maior que a dimensionalidade do espaço de dados?

7- Qual é a vantagem de utilizar regularização com um autoencoder?

8- Suponha que você tenha um autoencoder de uma camada escondida já treinado e queira visualizar a função computada por cada nó da camada escondida. Sua entrada é composta de n pixels x_i. Considere o caso onde a entrada obedece a seguinte restrição: $||x||^2 = \sum_{i=1}^{n} x_i^2 \leq 1$. Mostre que a entrada que maximiza a ativação do nó i é encontrada quando o pixel x_j segue a seguinte forma (desconsidere o bias):

$$x_j = \frac{W_{ij}}{\sqrt{\sum_{i=1}^{n}(W_{ij})^2}}$$

9- Considere um autoencoder convolucional com uma camada escondida, a primeira camada é uma convolução 2D (valid padding). Qual será a última camada?

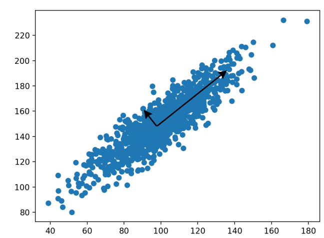
10- Suponha que queremos decompor a matriz D de dimensão $n \times d$ nas matrizes U e V de dimensões $n \times k$ e $d \times k$ respectivamente, ou seja, $D \approx UV'$. Mostre como é possível resolver este problema através de um autoencoder.

Solução

1- Aprendizagem supervisionada é a tarefa de encontrar uma função a partir de dados de treinamento rotulados. Exemplos: regressão e classificação. Na aprendizagem não-supervisionada temos menos informação sobre os objetos, em particular, o conjunto de treinamento não é rotulado. Exemplos: agrupamento e redução de dimensionalidade.

2- Basta mostrar que $\Sigma v = \lambda v$ onde λ é um valor real e positivo. De fato, a multiplicação matricial elementar produz $\Sigma v = (2\rho + 1)v$ e, portanto, v é autovetor com autovalor igual a $(2\rho + 1)$.

3-



4- Autoencoders visam aprender o mapeamento de características inversas $h = g_e(x; \theta_e)$, isto é, um codificador. Entretanto essa função não pode ser aprendida diretamente, já que não podemos medir h. A ideia de um autoencoder é aprender junto com o encoder uma função de decoder $x = g_d(h; \theta_d)$, tal que uma entrada x pode ser reconstruída como $\hat{x} = g_d(g_e(x))$. O autoencoder pode ser treinado usando uma função de perda quadrática: $L(\theta) = \sum_i \|x_i - \hat{x}_i\|^2$

Como o autoencoder é forçado a reconstruir com precisão as entradas do espaço latente aprendido, ele precisa aprender representações compactas e expressivas das entradas. As distâncias no espaço latente são, portanto, mais significativas do que no espaço de características original.

5- Possíveis aplicações:

- Redução de dimensionalidade: A dimensionalidade do espaço latente é geralmente menor que a dimensionalidade do espaço de dados, de tal forma que o modelo aprende uma representação informativa de menor dimensão dos dados.
- Pré-treinamento: Um autoencoder pode ser usado como modelo pré-treinado para uma nova rede neural, e.g., um classificador. Por exemplo, podemos pegar o codificador treinado, adicionar algumas camadas densas para previsão e treinar novamente o modelo resultante de ponta a ponta com uma loss de classificação, como a entropia cruzada binária.

- Detecção de outliers: O erro de reconstrução é geralmente grande para entradas que são muito diferentes dos padrões usuais observados nos dados de treinamento. Isso pode ser usado para detecção de anomalias/outliers usando simplesmente o erro de reconstrução como um score de outlier.

6- Nesse caso, o autoencoder simplesmente aprenderá a função de identidade e obterá um erro de reconstrução zero. Como o espaço latente tem um número suficiente de dimensões, o autoencoder pode simplesmente copiar a entrada para o espaço latente e vice-versa. Esse é um problema geral se o autoencoder tiver muita capacidade. Ao introduzir um gargalo, o autoencoder é forçado a aprender representações compactas e expressivas. Modificações como Denoising (DAE) ou usando regularização adicional (Sparse AE ou CAE) ajudam a evitar tais efeitos.

7- Ainda que o espaço latente seja grande, a regularização forçará o autoencoder a descobrir estruturas nos dados, pois os neurônios tendem a não ser ativados na maioria das vezes.

8- Como a função de ativação é monotônica crescente, maximizar $f(x)$ é equivalente a maximizar x. Devemos então maximizar a função (para o nó i):

$$W_i \cdot x$$

Note que o produto escalar de dois vetores é maximizados quando o ângulo entre eles é 0 (pois $W \cdot X = ||W_i||\,||x|| \cos\theta$). Como a norma de x é 1, então o valor de x que maximizará a função é o vetor na mesma direção que W_i com norma 1, portanto:

$$x = \frac{W_i}{||W_i||},\ x_j = \frac{W_{ij}}{\sqrt{\sum_{i=1}^{n}(W_{ij})^2}}$$

9- Uma convolução transposta com o filtro do mesmo tamanho do na camada de convolução.

10- O autoencoder possui somente uma camada escondida com k nós com função de ativação linear e d nós na camada de entrada e saída. Dado a entrada D, temos então que as ativações dos nós escondidos são as linhas de U e os pesos da camada do decoder são V'. Os pesos da camada do encoder formam a pseudo-inversa de V', igual a $(VV')^{-1}V$ (não é necessário derivar esta última parte).