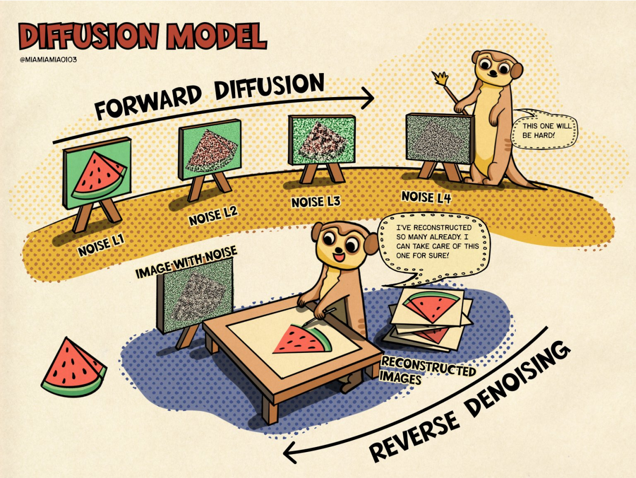
Modelos Generativos

Nahuel Costa

Grado en Ciencia e Ingeniería de Datos

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Modelos de difusión



Introducción

Los modelos de difusión se inspiran en la termodinámica del no equilibrio. Añaden ruido a los datos para después aprender a invertir el proceso para reconstruir las muestras a partir del ruido.

Introducción

En concreto, se puede aplicar un **proceso de difusión** para convertir gradualmente los datos observados x_0 en una versión con ruido x_T haciendo pasar los datos a través de T pasos de un encoder estocástico $q(x_t|x_{t-1})$. Después de suficientes pasos, se tendría $x_T \approx \mathcal{N}(0|I)$, o cualquier otra distribución de referencia conveniente. Después, se aprende un proceso inverso para deshacer este proceso, pasando el ruido a través de T pasos de un decoder $p(x_{t-1}|x_t)$ hasta generar x_0 de nuevo.

Introducción

Estos modelos son similares a los VAE jerárquicos, en el sentido de que tienen un encoder y un decoder y siguen una secuencia de pasos o jerarquías y también a los normalizing flows ya que aplican una secuencia de transformaciones a los datos.

Introducción

Sin embargo, los modelos de difusión se aprenden con un procedimiento fijo y la variable latente tiene la misma dimensionalidad que los datos originales. Además, el encoder es un modelo simple prefijado y el decoder se comparte en todos los pasos. Estas restricciones dan lugar a una función de error simple, que facilita el entrenamiento y evita el riesgo del posterior collapse.

Introducción

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x_t}; \sqrt{1-\beta_t}\mathbf{x_{t-1}}, \beta_t\mathbf{I})$$

Data

Noise

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})$$

Varios modelos generativos basados en difusión han sido propuestos, incluidos *diffusion probabilistic models* [1], *noise-conditioned score network* [2], *denoising diffusion probabilistic models* [3] y *variational diffusion models* [4].

Encoder (forwards diffusion)

El proceso de codificación hacia delante se define como un modelo lineal gaussiano simple:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t|\sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$$

donde los valores de $\beta_t \in (0,1)$ se eligen según un noise scheduler. La distribución conjunta sobre los estados latentes es

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$$

Encoder (forwards diffusion)

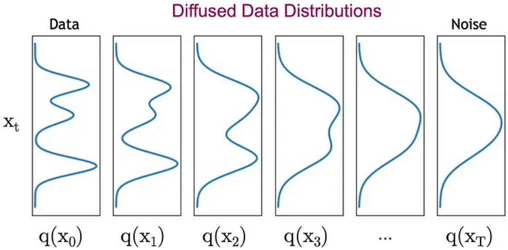
La distribución $q(\mathbf{x}_t|\mathbf{x}_0)$ se conoce como *diffusion kernel*. Aplicar esto a la distribución de datos de entrada y luego calcular los marginales incondicionales resultantes es equivalente a la convolución gaussiana:

$$q(\mathbf{x}_t) = \int q_0(\mathbf{x}_0) q(\mathbf{x}_t|\mathbf{x}_0) d\mathbf{x}_0$$

Encoder (forwards diffusion)

A medida que t aumenta, los marginales se vuelven más simples, como se muestra en la siguiente figura. En imágenes, este proceso primero eliminará el contenido de alta frecuencia (es decir, detalles de bajo nivel, como la textura), y luego eliminará el contenido de baja frecuencia (información de alto nivel o “semántica”, como la forma).

Encoder (forwards diffusion)



Modelo de difusión sobre datos 1D. El proceso de difusión hacia delante transforma gradualmente la distribución empírica de datos $q(x_0)$ en una distribución objetivo simple $q(x_T) = N(0, I)$.

Decoder (reverse diffusion)

En el proceso de difusión inversa queremos invertir el proceso. Si se conoce la entrada x_0, se puede calcular la inversa matemáticamente. Si se quiere generar un nuevo punto de datos, aunque no se conozca x_0, se entrena al generador/decoder para que se aproxime a la distribución de los datos sobre el proceso inverso. Por lo tanto, sigue la forma $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}|\mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t))$

La distribución conjunta correspondiente sobre todas las variables generadas está dada por $p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)$, donde establecemos $p(x_T) = \mathcal{N}(0, I)$

Decoder (reverse diffusion)

Para generar datos a partir del modelo, se muestrea un punto $x_T \approx N(0, I)$, y luego se ejecuta la cadena de Markov hacia atrás, muestreando $x_t \approx p_\theta(x_t | x_{t+1})$ hasta obtener una muestra en el espacio de datos original, x_0.

Model fitting

Los modelos de difusión, al igual que con los VAE, pueden utilizar el ELBO como función de error. En este contexto, el ELBO es una suma de términos que corresponden a la divergencia KL entre las distribuciones de ruido añadido en cada paso del proceso hacia delante y las distribuciones predichas por el proceso inverso. Al optimizar esta ELBO, el modelo aprende a invertir el proceso de adición de ruido, aprendiendo efectivamente a eliminar el ruido de los datos en cada paso.

Model fitting

En lugar de entrenar el modelo para predecir la media de la versión con ruido $xt - 1$ dada su entrada ruidosa x_t, se entrena el modelo para predecir el ruido, a partir del cual se puede calcular la media.

Por razones de tiempo no haremos una demostración de las fórmulas, pero podéis consultar el siguiente post para una explicación más detallada.

Model fitting

Los modelos de difusión están estrechamente relacionados con el **score matching**, una técnica en la que el modelo aprende el gradiente (score) de la densidad de datos logarítmica. El ELBO proporciona un método para estimar los parámetros del proceso inverso, que está relacionado con el score matching en el sentido de que se aprende la función de puntuación del proceso inverso.

Model fitting

Maximizar el ELBO equivale a minimizar la diferencia entre la distribución real de los datos y la distribución del modelo tras el proceso de difusión inverso. El ELBO suele incluir:

- Un término de reconstrucción que asegura que el modelo pueda reconstruir los datos a partir de una versión ruidosa.
- Un término de regularización (divergencia KL) que garantiza que el ruido añadido en el proceso hacia delante sigue la distribución esperada.

[1] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International conference on machine learning*, pp. 2256–2265, PMLR, 2015.

[2] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," *Advances in neural information processing systems*, vol. 32, 2019.

[3] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[4] D. Kingma, T. Salimans, B. Poole, and J. Ho, "Variational diffusion models," *Advances in neural information processing systems*, vol. 34, pp. 21696–21707, 2021.