AI

How to generate a picture that has a "given style"?

note

Picture $\longrightarrow$ matrix $k \times l$

0/1's if grey scale

o/w RBG, encoded by $\{0, \dots, 255\}$ each

So # possibilities is $256^{3k l}$ ← huge!

$k = l = 256$: 65,536 pixels

→ 196,608 dimensions

512 → 786,432 dimensions

1024 → 3,145,728 dimensions

4k (2160 × 3840) : 25 million dimensions

Suppose Picasso paintings are all stored. We want to generate a <u>new</u> random one that looks like a Picasso.

Imagine Picasso is a random generator of paintings.

So: probability mass function

we want to guess it. tough problem

Better: generate samples from it

Suppose $\bar{\Phi}_0$ is unknown.

But we have samples from it:

$x_0^{(1)}, \dots, x_0^{(M)}$ (M samples)

We want to generate new samples. But <u>not</u> by picking one of $x_0^{(i)}$ at random. (Bootstrapping)

Suppose P is the transition matrix of some irreducible MC.

If we know: $\bar{\Phi}_n = \bar{\Phi}_0 P^n$

we can retrieve $\bar{\Phi}_0 = \bar{\Phi}_n P^{-n}$

But to know $\bar{\Phi}_n$ we need $\bar{\Phi}_0$!

We know that starting at <u>any</u> $x_0^{(i)}$, the distribution of X_n will be $\longrightarrow \bar{\Phi}_\infty$ as $n \to \infty$.

So could try $\bar{\Phi}_n \sim \bar{\Phi}_\infty$

But $\bar{\Phi}_\infty = \bar{\Phi}_\infty P \Rightarrow \bar{\Phi}_\infty P^{-1} = \bar{\Phi}_\infty \Rightarrow \bar{\Phi}_\infty P^{-n} = \bar{\Phi}_\infty$

not $\bar{\Phi}_0$

Idea: use Bayes rule to reverse the MC!

$X_{n-1} \to X_{n-2}$

$$\Phi_n \to \Phi_{n-1} \to \cdots \to \Phi_0$$

reversed MC $X_n \to X_{n-1}$

unknown

but approximate w/ Φ_∞.

How to reverse the MC?

$$\underbrace{P(X_{n-1}=j \mid X_n = i)}_{\text{unknown}} = \frac{P(X_n = i \mid X_{n-1}=j)\, P(X_{n-1}=j)}{P(X_n = i)}$$

known, known, unknown

call it $\overleftarrow{P}_n(i,j)$

learn it from data.

$$\left.\begin{array}{l}
X_0^{(1)} \longrightarrow X_1^{(1)} \to \cdots \to X_n^{(1)} \\
X_0^{(2)} \longrightarrow X_1^{(2)} \to \cdots \to X_n^{(2)} \\
\quad \vdots \qquad\qquad \cdots \\
X_0^{(M)} \longrightarrow X_1^{(M)} \to \cdots \to X_n^{(M)}
\end{array}\right\} \text{can be quickly generated}$$

Now, for each k:

use $\left(X_{k-1}^{(1)}, X_k^{(1)}\right) \dots \left(X_{k-1}^{(M)}, X_k^{(M)}\right)$

to <u>estimate</u> $\overleftarrow{P}_k(i,j)$

Neural network:

$\overleftarrow{P}_k(i,j)$ unknown function of i & j

Want $\overleftarrow{P}_k\left(x_k^{(m)}, x_{k-1}^{(m)}\right)$ as large as possible

w/ constraints: $\sum_j \overleftarrow{P}_k(i,j) = 1$

Linear regression:

$$\overleftarrow{P}_k(i,j) = \alpha_k i + \beta_k j + \gamma_k$$

estimate $\alpha_k, \beta_k, \gamma_k$

But $\overleftarrow{P}_k$ is not linear.

Neural network: Similar to but w/ a nonlinear form.

(<u>still</u> has parameters to estimate and a [rather sophisticated] method to estimate these parameters from the data)

Once all these parameters are estimated, can use f_k as an approximation of p_k

Start MC at Φ_∞ ← known b/c we designed P

Run MC backward using f_k

to approximately sample from Φ_0!

$$X_n \sim \Phi_\infty \xrightarrow{f_n} X_{n-1} \xrightarrow{f_{n-1}} X_{n-2} \to \cdots \xrightarrow{f_1} \underline{\underline{X_0}}$$

approximation of Φ_n

approximations of $\overleftarrow{P}_k$

close to $\underline{\Phi_0}$ ✓