Sid

Gaussian Distribution

$x \sim \mathcal{N}(\mu, \sigma^2)$ — parameterized by mean and variance

"distributed as"

$$\boxed{p(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}}$$

$$\boxed{\mu = \frac{1}{m}\sum_{i=1}^{m} x^{(i)} \quad , \quad \sigma^2 = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)}-\mu)^2}$$

Algorithm

Training set: $\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$ $\quad x^{(i)} \in \mathbb{R}^n$

$p(x) = p(x_1;\mu_1,\sigma_1^2) \times p(x_2;\mu_2\,\sigma_2^2) \cdots \times p(x_n;\mu_n\,\sigma_n^2)$

$= \prod_{j=1}^{n} p(x_j;\mu_j,\sigma_j^2)$

1) Choose features x_i that may be anomalous examples

2) Fit parameters $\mu_1, \dots, \mu_n \mid \sigma_1^2, \dots, \sigma_n^2$

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x_j^{(i)}$$

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m}(x_j^{(i)}-\mu_j)^2$$

3) Given new example $x \longrightarrow$ compute $p(x)$:

$$p(x) = \prod_{j=1}^{n} p(x_j;\mu_j;\sigma_j^2)$$

Anomaly if $p(x) < \varepsilon$

Evaluate Learning Algorithm:

- Fit model $p(x)$ on training set $\{x^{(1)} \rightarrow x^{(m)}\}$
- On CV example x, predict:

$$y = \begin{cases} 1 & p(x) < \varepsilon \quad \text{(anomaly)} \\ 0 & p(x) \geq \varepsilon \quad \text{(normal)} \end{cases}$$

- Evaluation metrics:
  - Precision / Recall - F_1 score

* CV test set to choose ε

Anomaly Detection vs Supervized Learning

Anomaly Detection	Supervized Learning
• small # of + ($y=1$) examples & large # of − ($y=0$) examples	• large # of + and − examples
• many types of anomalies (less predictable)	• more predictable anomalies

Choosing Features :

- transformations on feature x:
  - $\log(x+c)$, $\sqrt{x}$, x^c, ...

- Goal: $p(x)$ large for normal examples,
  $p(x)$ small for anomalous examples

Multivariate Gaussian Distribution

$\mu \in \mathbb{R}^n, \ \Sigma \in \mathbb{R}^{n \times n}$

$$p(x;\mu,\Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)$$

- original model: axis-aligned
    - computationally cheaper
    - OK if small m
- Multi: automatically capture correlation between different features of x
    - must have $m > n$
    - more expensive