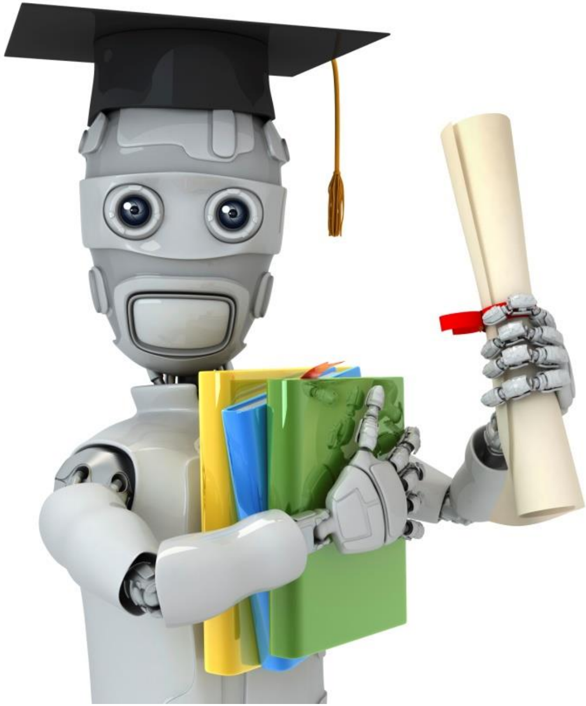


Machine Learning

Anomaly detection

Problem motivation

Anomaly detection example

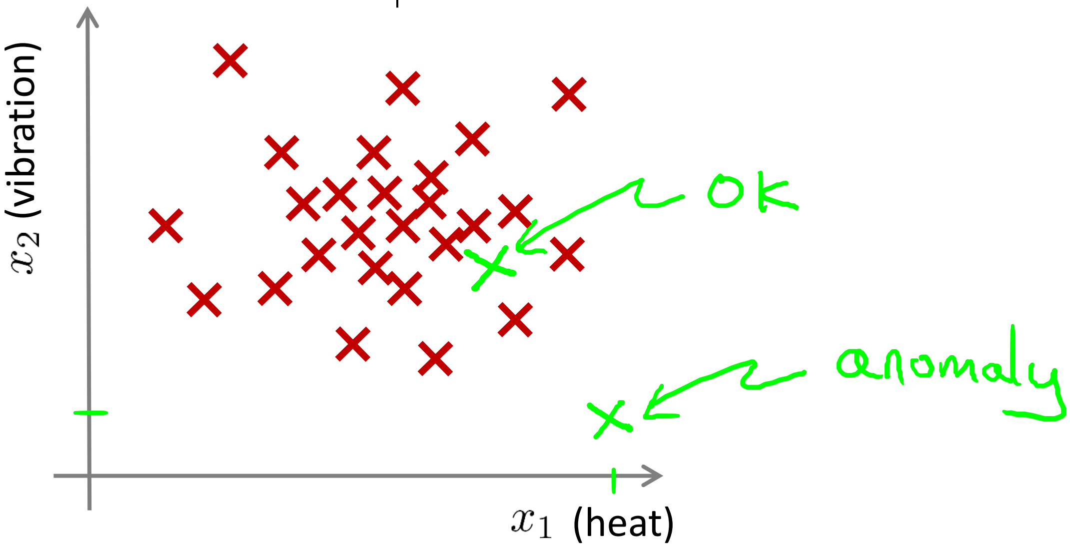
Aircraft engine features:

- x_1 = heat generated
- x_2 = vibration intensity

...

Dataset: $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$

New engine: x_{test}



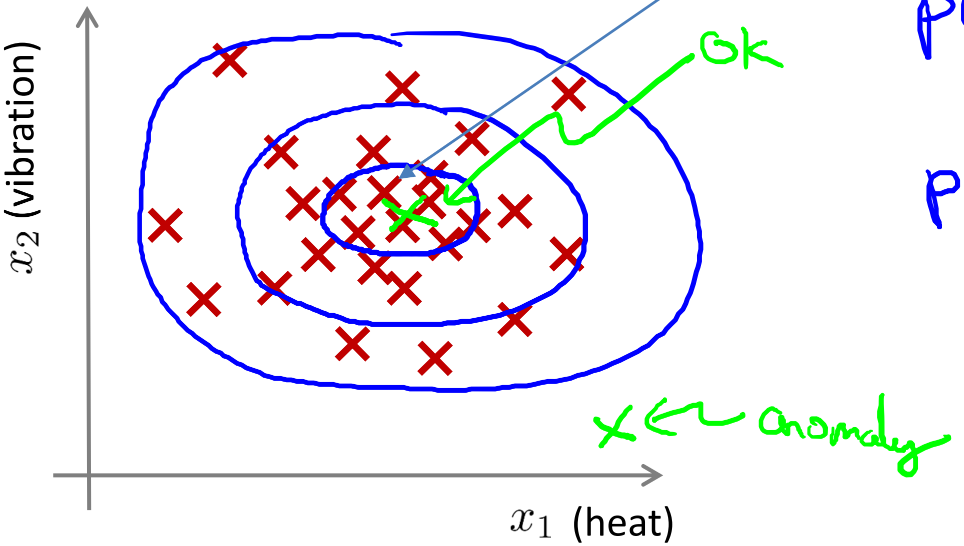
Andrew Ng

Density estimation

(Probability is high at the centre)

- Dataset: $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$
- Is x_{test} anomalous?

Model $p(x)$.



$p(x_{test}) < \varepsilon \rightarrow$ flag anomaly

$p(x_{test}) \geq \varepsilon \rightarrow$ OK

Andrew Ng

Anomaly detection example

- Fraud detection:
  - $x^{(i)}$ = features of user i's activities
  - Model $p(x)$ from data.
  - Identify unusual users by checking which have $p(x) < \varepsilon$

  x_1 x_2 x_3 x_4 $p(x)$
- Manufacturing
- Monitoring computers in a data center.
  - $x^{(i)}$ = features of machine i

    x_1 = memory use, x_2 = number of disk accesses/sec,

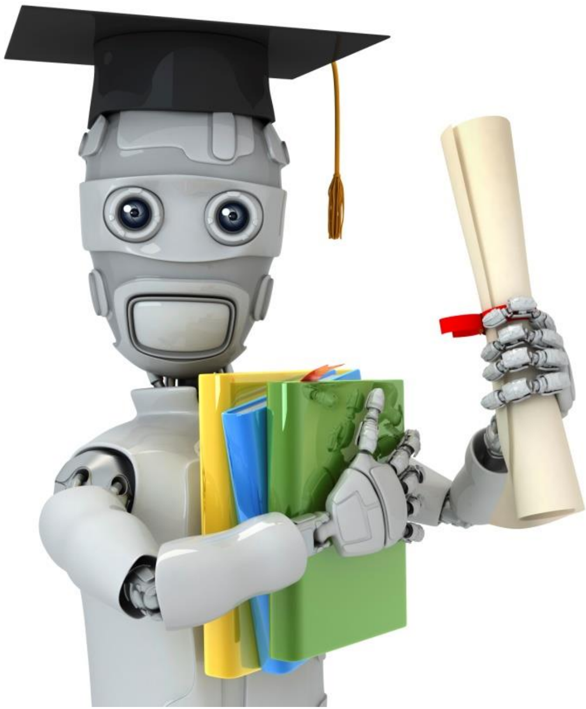
    x_3 = CPU load, x_4 = CPU load/network traffic.

    … $p(x) < \varepsilon$

Your anomaly detection system flags x as anomalous whenever $p(x) \leq \epsilon$. Suppose your system is flagging too many things as anomalous that are not actually so (similar to supervised learning, these mistakes are called false positives). What should you do?

- Try increasing ϵ.
- Try decreasing ϵ.

Correct Response



Machine Learning

Anomaly detection

Gaussian distribution

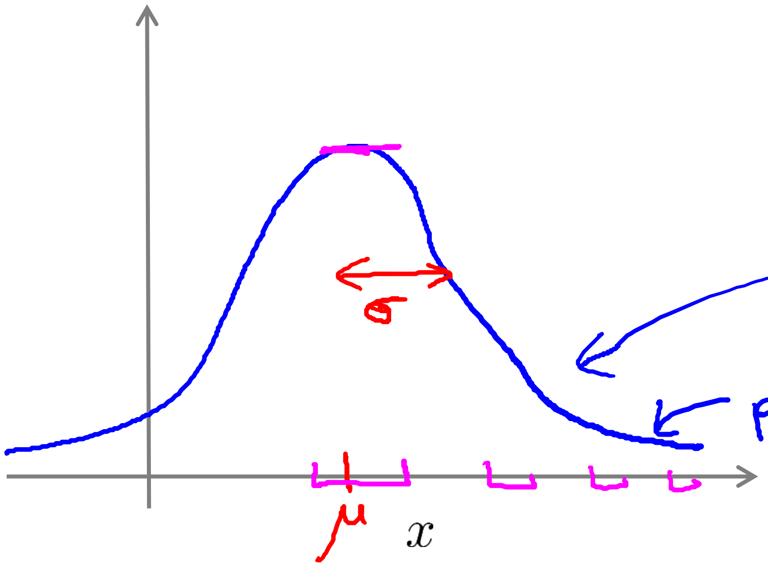
Gaussian (Normal) distribution

Say $x \in \mathbb{R}$. If x is a distributed Gaussian with mean μ, variance σ^2.

$x \sim \mathcal{N}(\mu, \sigma^2)$

"distributed as"

σ standard deviation

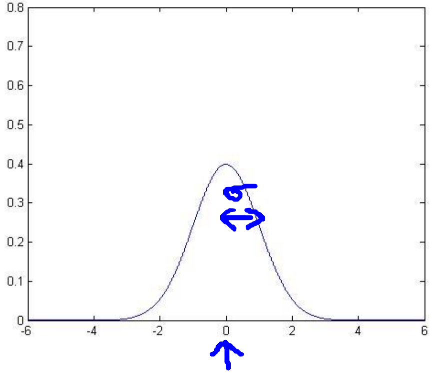


$p(x; \mu, \sigma^2)$

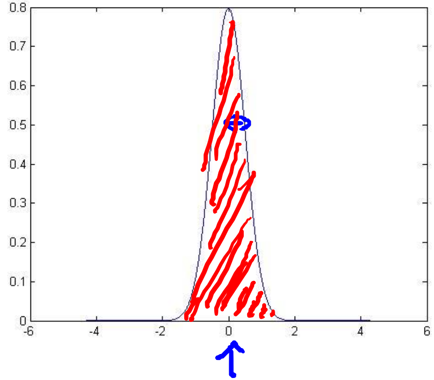
$$p(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Gaussian distribution example

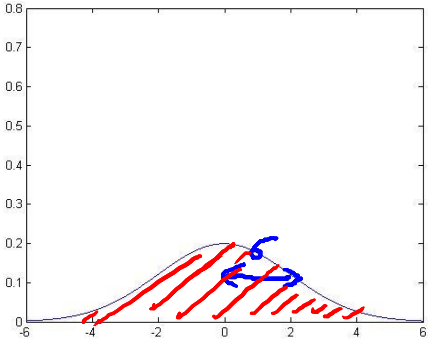
$\mu = 0, \sigma = 1$



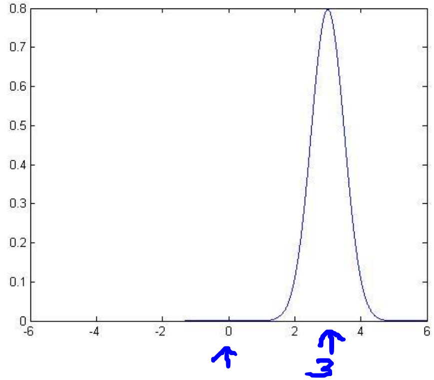
$\mu = 0, \sigma = \underline{0.5}$ $\sigma^2 = 0.25$



$\mu = 0, \sigma = 2$



$\mu = 3, \sigma = 0.5$

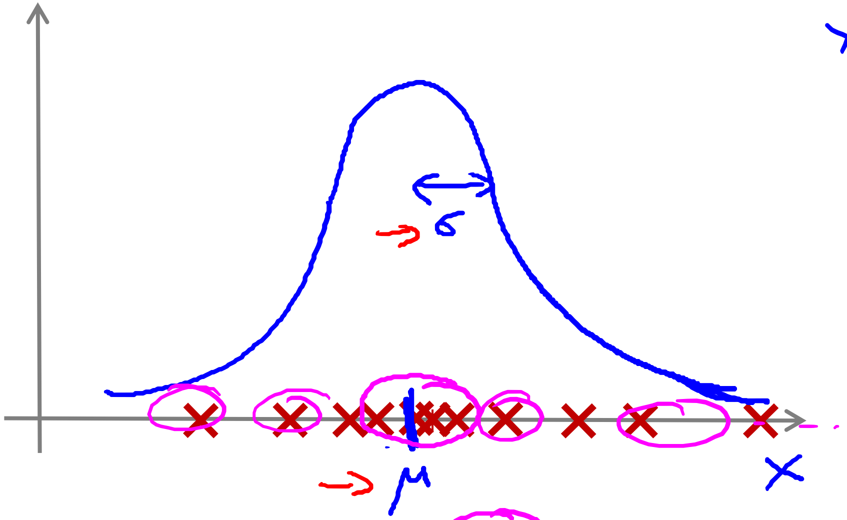


Parameter estimation

(Given the dataset we want to estimate the value of mu and sigma)

Dataset: $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$ $\quad \underline{x^{(i)} \in \mathbb{R}}$

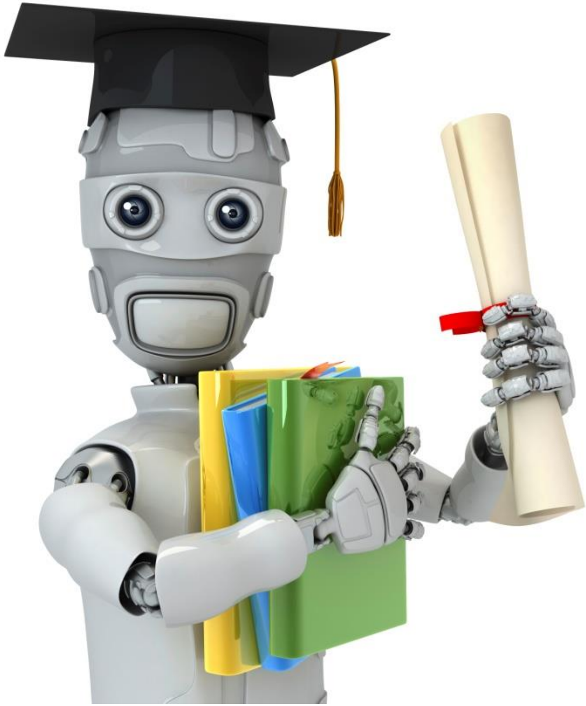
$x^{(i)} \sim \mathcal{N}(\mu, \sigma^2)$



$\mu = \frac{1}{m} \sum_{i=1}^{m} x^{(i)}$

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

$m-1$ $\quad \frac{1}{m-1}$



Machine Learning

Anomaly detection

Algorithm

Density estimation

Training set: $\{x^{(1)}, \ldots, x^{(m)}\}$

Each example is $x \in \mathbb{R}^n$

$x_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$

$x_2 \sim \mathcal{N}(\mu_2, \sigma_2^2)$

$x_3 \sim \mathcal{N}(\mu_3, \sigma_3^2)$

$p(x)$

$= \boxed{p(x_1; \mu_1, \sigma_1^2)\, p(x_2; \mu_2, \sigma_2^2)\, p(x_3; \mu_3, \sigma_3^2) \cdots p(x_n; \mu_n, \sigma_n^2)}$

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

$\sum_{i=1}^{n} i = 1 + 2 + 3 + \cdots + n$

$\prod_{i=1}^{n} i = 1 \times 2 \times 3 \times \cdots \times n$

Given a training set $\{x^{(1)}, \ldots, x^{(m)}\}$, how would you estimate each μ_j and σ_j^2 (Note $\mu_j \in \mathbb{R}, \sigma_j^2 \in \mathbb{R}$.)

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

Correct Response

Anomaly detection algorithm

1. Choose features x_i that you think might be indicative of anomalous examples.

$\{x^{(1)}, \ldots, x^{(m)}\}$

2. Fit parameters $\mu_1, \ldots, \mu_n, \sigma_1^2, \ldots, \sigma_n^2$

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

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

$p(x_j; \mu_j, \sigma_j^2)$

$\mu_1, \mu_2, \ldots, \mu_n$

$$\mu = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{bmatrix} = \frac{1}{m} \sum_{i=1}^{m} x^{(i)}$$

3. Given new example x, compute $p(x)$:

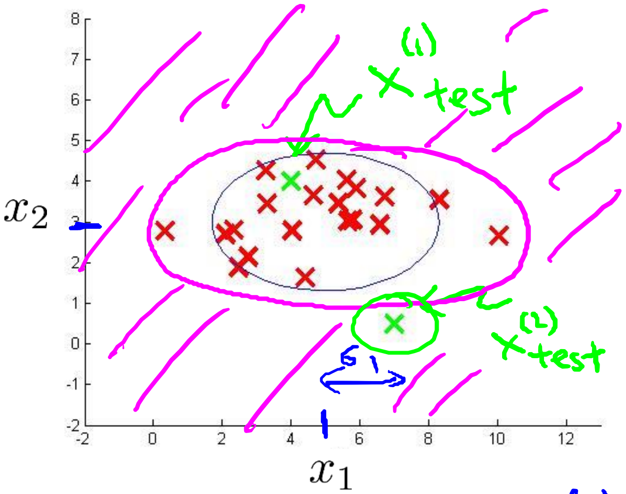
$$p(x) = \prod_{j=1}^{n} p(x_j; \mu_j, \sigma_j^2) = \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left(-\frac{(x_j - \mu_j)^2}{2\sigma_j^2}\right)$$

Anomaly if $p(x) < \varepsilon$

Andrew Ng

Anomaly detection example

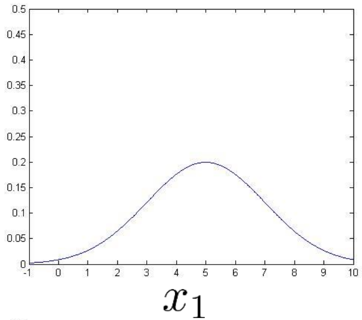
$\rightarrow p(x) = p(x_1; \mu_1, \sigma_1^2) \times p(x_2; \mu_2, \sigma_2^2)$



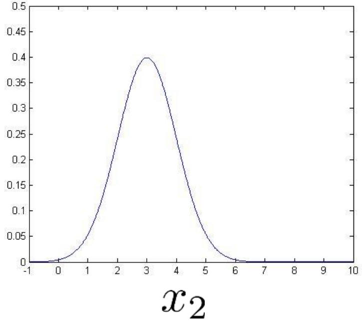
σ_1^2, σ_2^2

$\sigma_1^2 = 4$

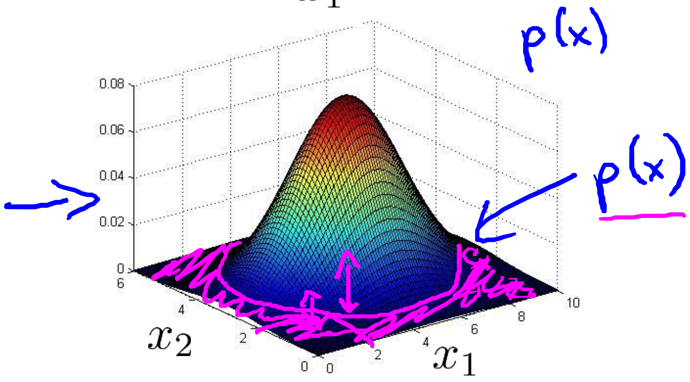
$$\boxed{\begin{aligned}\mu_1 &= 5, \sigma_1 = 2\\ \mu_2 &= 3, \sigma_2 = 1\end{aligned}}$$



$p(x_1; \mu_1, \sigma_1^2)$



$p(x_2; \mu_2, \sigma_2^2)$



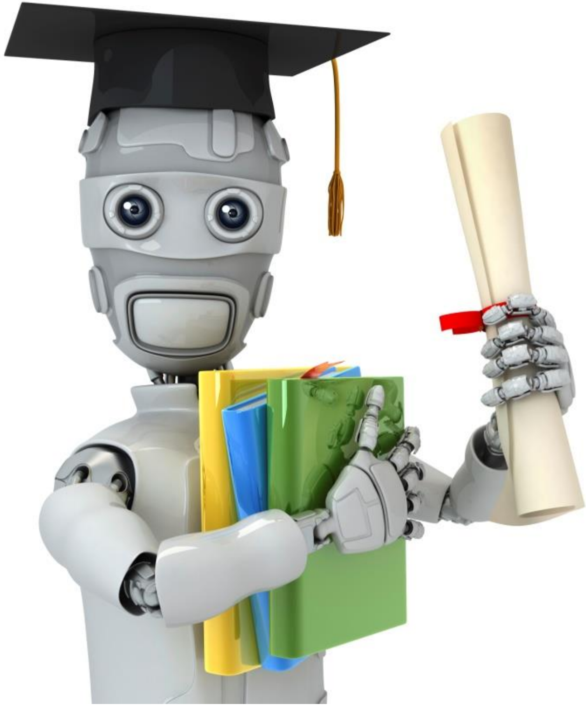
$p(x)$

$$\varepsilon = 0.02$$

$$p(x_{test}^{(1)}) = 0.0426 \quad \geq \varepsilon$$

$$p(x_{test}^{(2)}) = 0.0021 \quad < \varepsilon$$

Andrew Ng



Machine Learning

Anomaly detection

Developing and evaluating an anomaly detection system

The importance of real-number evaluation

When developing a learning algorithm (choosing features, etc.), making decisions is much easier if we have a way of evaluating our learning algorithm.

- Assume we have some labeled data, of anomalous and non-anomalous examples. ($y = 0$ if normal, $y = 1$ if anomalous).
- Training set: $x^{(1)}, x^{(2)}, \ldots, x^{(m)}$ (assume normal examples/not anomalous)
- Cross validation set: $(x_{cv}^{(1)}, y_{cv}^{(1)}), \ldots, (x_{cv}^{(m_{cv})}, y_{cv}^{(m_{cv})})$
- Test set: $(x_{test}^{(1)}, y_{test}^{(1)}), \ldots, (x_{test}^{(m_{test})}, y_{test}^{(m_{test})})$

y=1

Aircraft engines motivating example

10000 good (normal) engines

20 flawed engines (anomalous) 2-50

$y=1$

$\mu_1, \sigma_1^2, \ldots, \mu_n, \sigma_n^2$.

Training set: 6000 good engines $(y=0)$ $p(x) = p(x_1; \mu_1, \sigma_1^2) \cdots p(x_n; \mu_n, \sigma_n^2)$

CV: 2000 good engines $(y = 0)$, 10 anomalous $(y = 1)$

Test: 2000 good engines $(y = 0)$, 10 anomalous $(y = 1)$

Alternative:

Training set: 6000 good engines

CV: 4000 good engines $(y = 0)$, 10 anomalous $(y = 1)$

Test: 4000 good engines $(y = 0)$, 10 anomalous $(y = 1)$

Algorithm evaluation

- Fit model $p(x)$ on training set $\{x^{(1)}, \ldots, x^{(m)}\}$
- On a cross validation/test example x, predict

$(x_{test}^{(i)}, y_{test}^{(i)})$

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

$y = 0$

Possible evaluation metrics:

- True positive, false positive, false negative, true negative
- Precision/Recall
- F_1-score

CV

Test set.

Can also use cross validation set to choose parameter ε

Suppose you have fit a model p(x). When evaluating on the cross validation set or test set, your algorithm predicts:

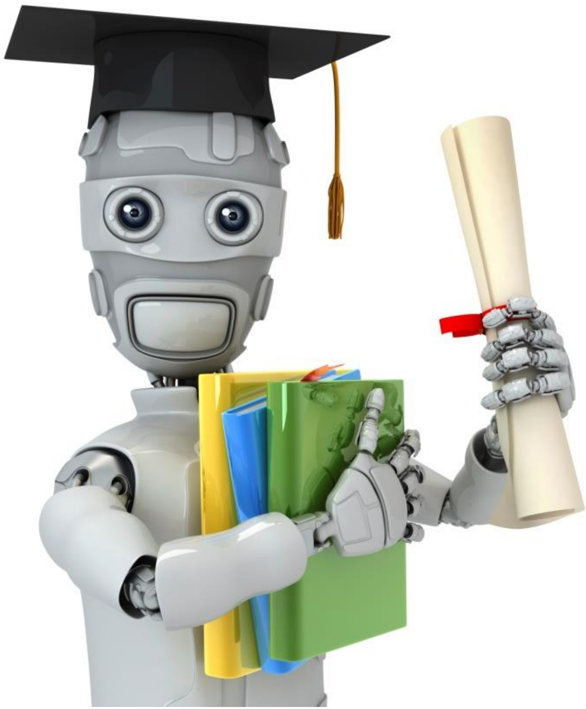
$$y = \begin{cases} 1 & \text{if } p(x) \leq \epsilon \\ 0 & \text{if } p(x) > \epsilon \end{cases}$$

Is classification accuracy a good way to measure the algorithm's performance?

- ○ Yes, because we have labels in the cross validation / test sets.
- ○ No, because we do not have labels in the cross validation / test sets.
- ◉ No, because of skewed classes (so an algorithm that always predicts y = 0 will have high accuracy).

Correct Response

- ○ No for the cross validation set; yes for the test set.



Machine Learning

Anomaly detection

Anomaly detection vs. supervised learning

Anomaly detection vs. Supervised learning

Anomaly detection

- Very small number of positive examples ($y = 1$). (0-20 is common).
- Large number of negative ($y = 0$) examples. $\boxed{p(x)}$
- Many different "types" of anomalies. Hard for any algorithm to learn from positive examples what the anomalies look like;
- future anomalies may look nothing like any of the anomalous examples we've seen so far.

Supervised learning

- Large number of positive and negative examples.
- Enough positive examples for algorithm to get a sense of what positive examples are like, future positive examples likely to be similar to ones in training set.

Spam

Anomaly detection vs. Supervised learning

Anomaly detection	Supervised learning
• Fraud detection $y=1$	• Email spam classification
• Manufacturing (e.g. aircraft engines)	• Weather prediction (sunny/rainy/etc).
• Monitoring machines in a data center	• Cancer classification
⋮	⋮

Andrew Ng

Which of the following problems would you approach with an anomaly detection algorithm (rather than a supervised learning algorithm)? Check all that apply.

- [x] You run a power utility (supplying electricity to customers) and want to monitor your electric plants to see if any one of them might be behaving strangely.

Correct Response

- [ ] You run a power utility and want to predict tomorrow's expected demand for electricity (so that you can plan to ramp up an appropriate amount of generation capacity).

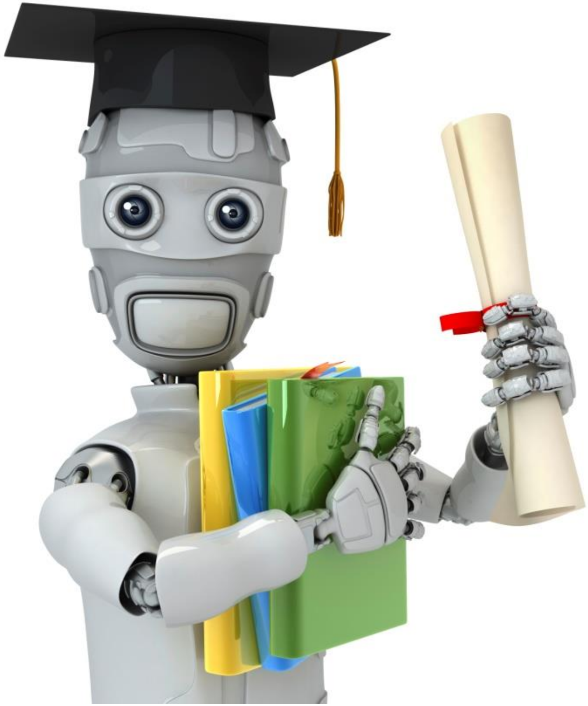
Correct Response

- [x] A computer vision / security application, where you examine video images to see if anyone in your company's parking lot is acting in an unusual way.

Correct Response

- [ ] A computer vision application, where you examine an image of a person entering your retail store to determine if the person is male or female.

Correct Response



Machine Learning

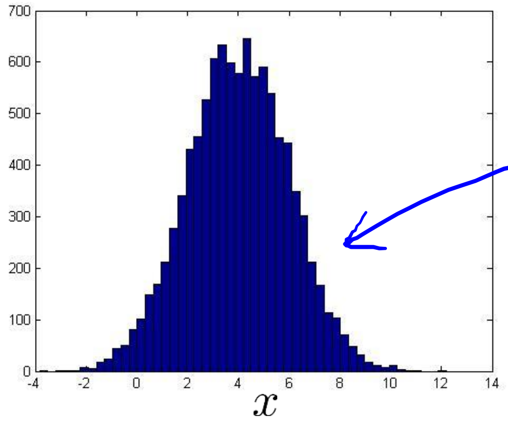
Anomaly detection

Choosing what features to use

Non-gaussian features

$p(x_i; \mu_i, \sigma_i^2)$

hist

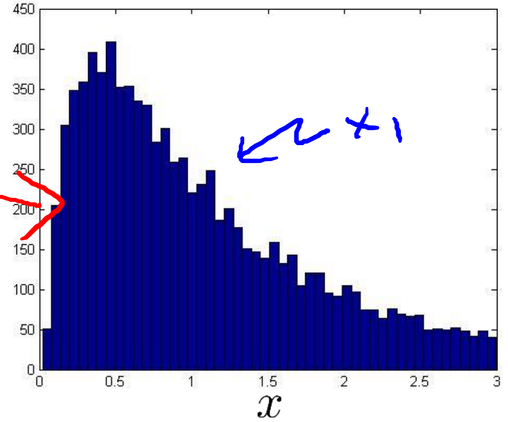


$x_1 \leftarrow \log(x_1)$

$x_2 \leftarrow \log(x_2+1)$ $\quad \log(x_2 + c)$

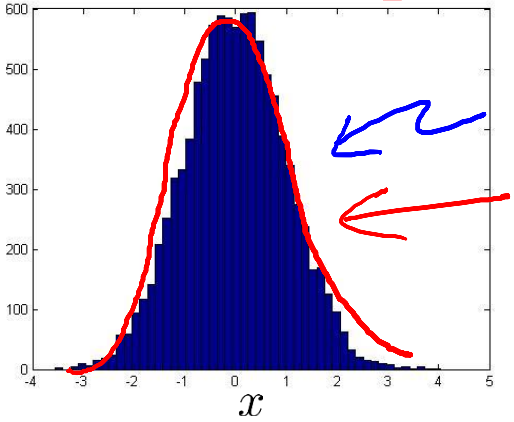
$x_3 \leftarrow \sqrt{x_3} = x_3^{\frac{1}{2}}$

$x_4 \leftarrow x_4^{\frac{1}{3}}$



x_1

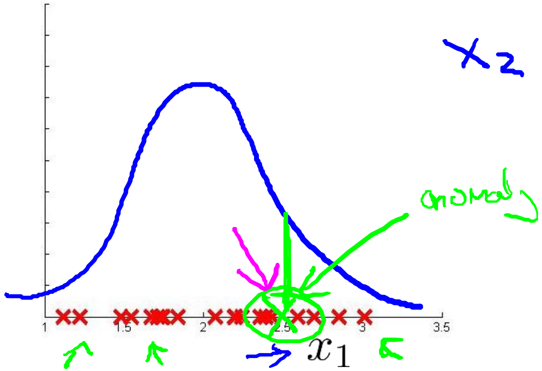
$\underline{\log(x)}$

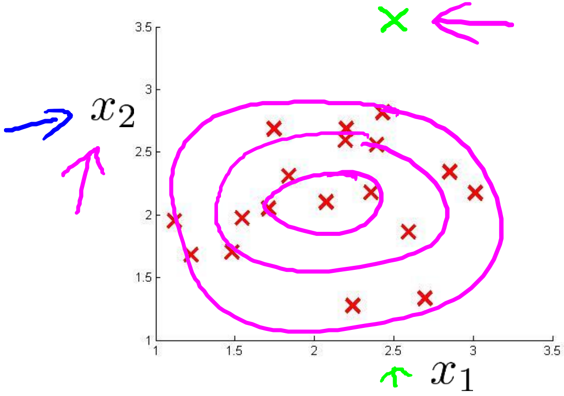


Error analysis for anomaly detection

Want $p(x)$ large for normal examples x.
$p(x)$ small for anomalous examples x.

Most common problem:
$p(x)$ is comparable (say, both large) for normal and anomalous examples





Monitoring computers in a data center

Choose features that might take on unusually large or small values in the event of an anomaly.

- x_1 = memory use of computer
- x_2 = number of disk accesses/sec
- x_3 = CPU load
- x_4 = network traffic

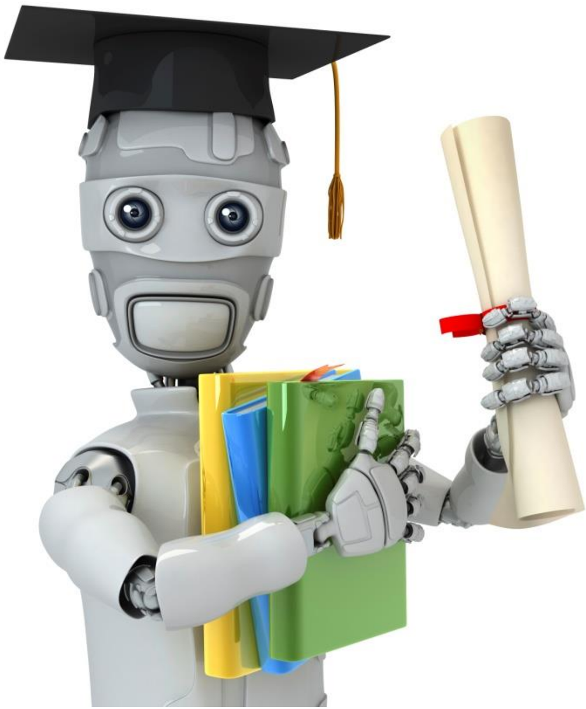
$$x_5 = \frac{\text{CPU load}}{\text{network traffic}} \qquad x_6 = \frac{(\text{CPU load})^2}{\text{network traffic}}$$

Suppose your anomaly detection algorithm is performing poorly and outputs a large value of p(x) for many normal examples and for many anomalous examples in your cross validation dataset. Which of the following changes to your algorithm is most likely to help?

- ○ Try using fewer features.
- ◉ Try coming up with more features to distinguish between the normal and the anomalous examples.

Correct Response

- ○ Get a larger training set (of normal examples) with which to fit p(x).
- ○ Try changing ϵ.

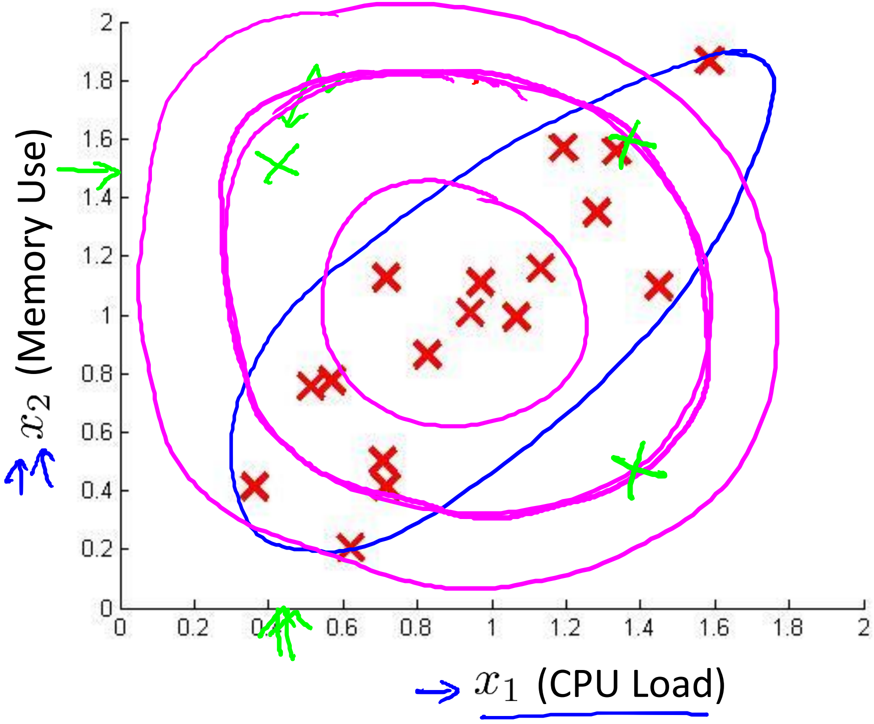


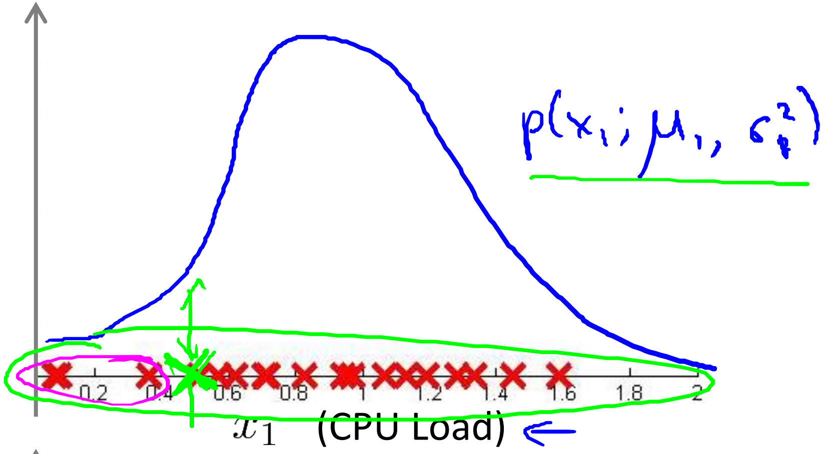
Machine Learning

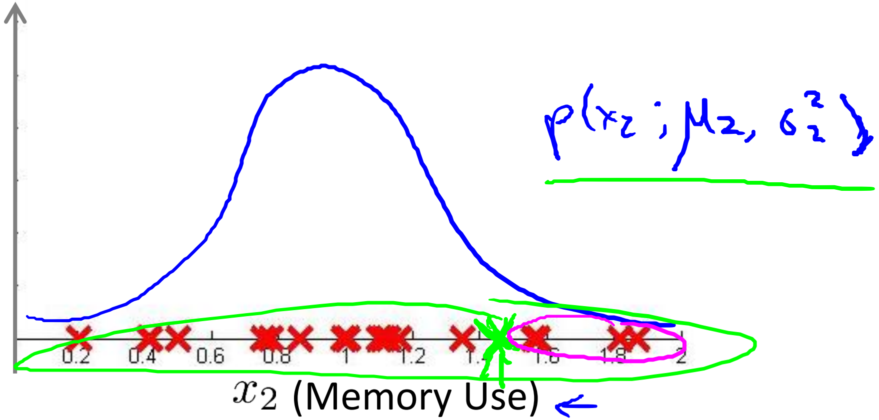
Anomaly detection

Multivariate Gaussian distribution

Motivating example: Monitoring machines in a data center







Andrew Ng

Multivariate Gaussian (Normal) distribution

$x \in \mathbb{R}^n$. Don't model $p(x_1), p(x_2), \ldots,$ etc. separately.
Model $p(x)$ all in one go.
Parameters: $\mu \in \mathbb{R}^n$, $\boxed{\Sigma \in \mathbb{R}^{n \times n}}$ (covariance matrix)

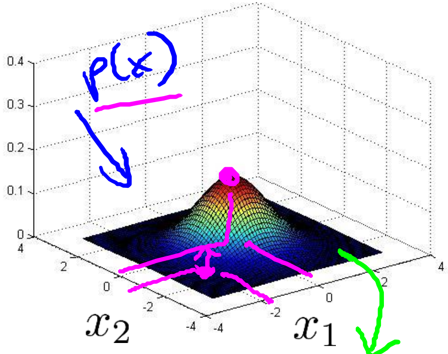
$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)$$

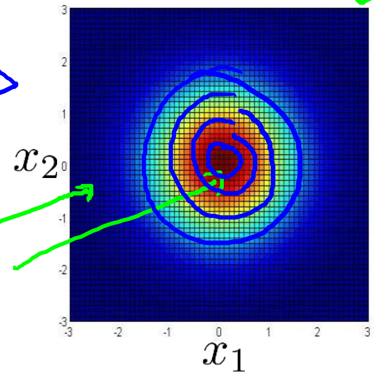
$|\Sigma|$ = determinant of Σ | det(Sigma)

Andrew Ng

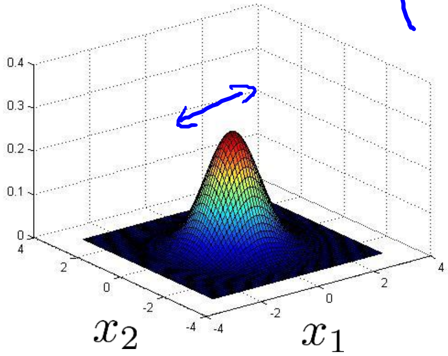
Multivariate Gaussian (Normal) examples

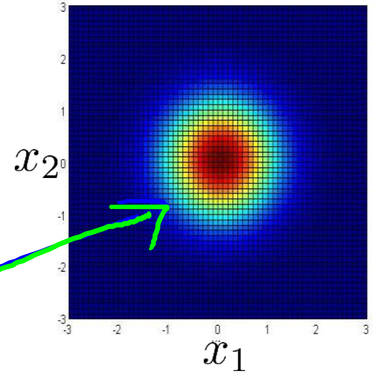
$$\boxed{\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}$$



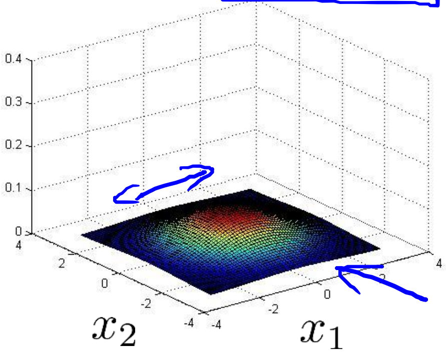


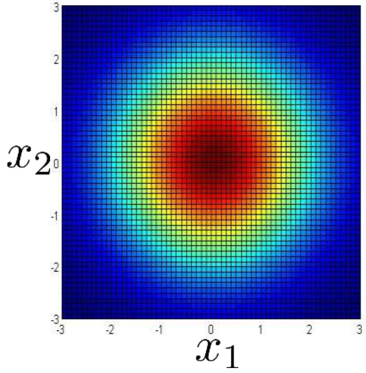
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \boxed{\Sigma = \begin{bmatrix} 0.6 & 0 \\ 0 & 0.6 \end{bmatrix}}$$





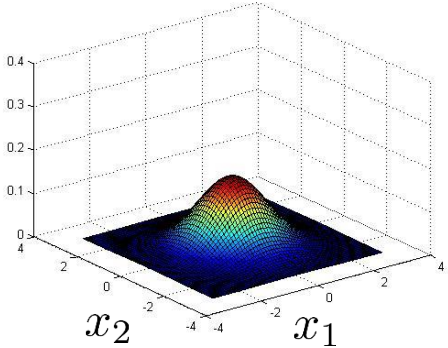
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \boxed{\Sigma = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}}$$

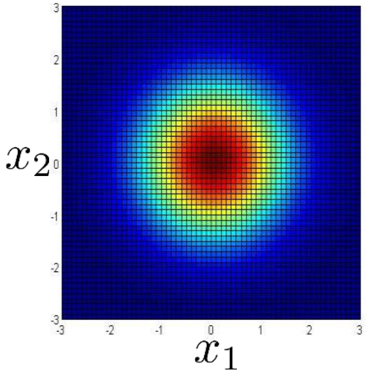




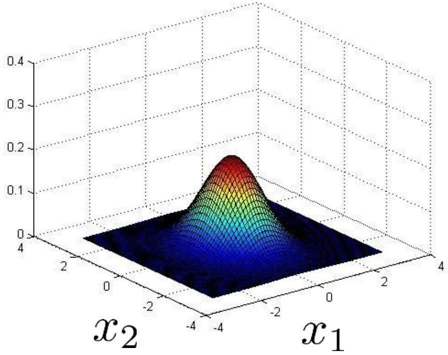
Multivariate Gaussian (Normal) examples

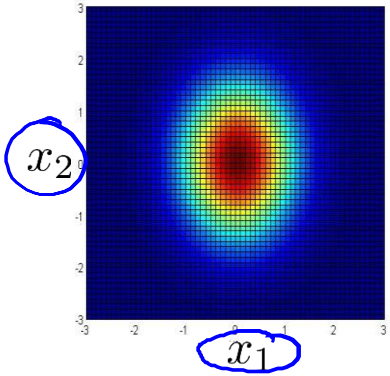
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$



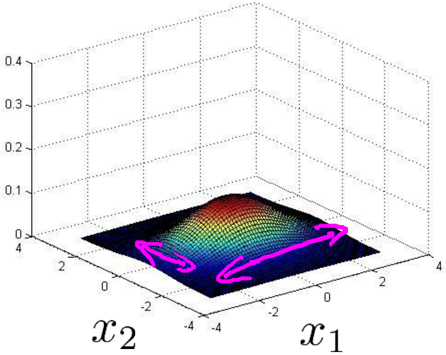


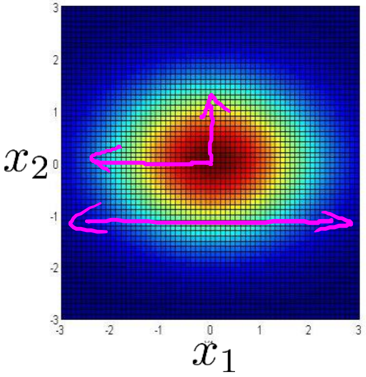
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 0.6 & 0 \\ 0 & 1 \end{bmatrix}$$





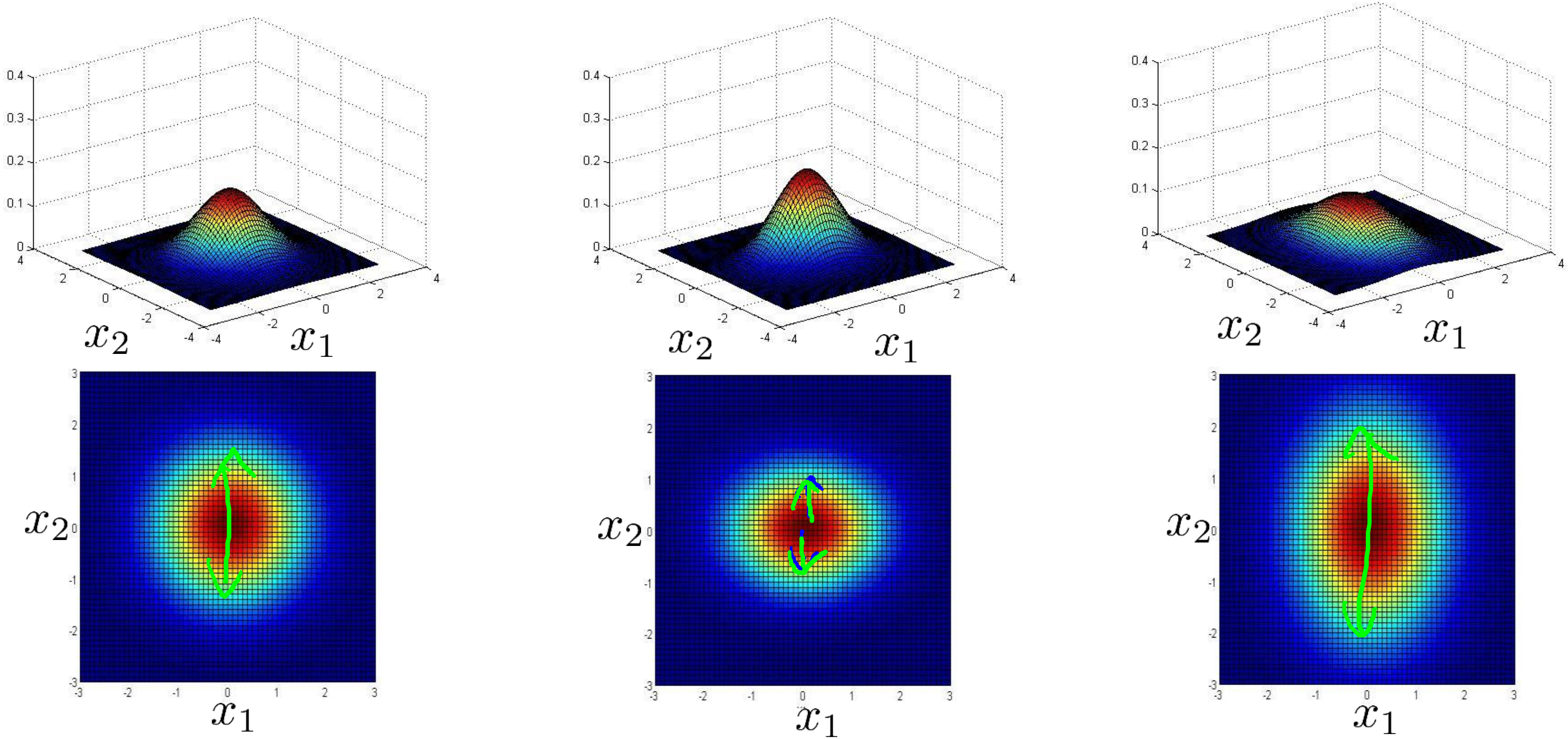
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$$





Multivariate Gaussian (Normal) examples

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 0.6 \end{bmatrix} \qquad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$$



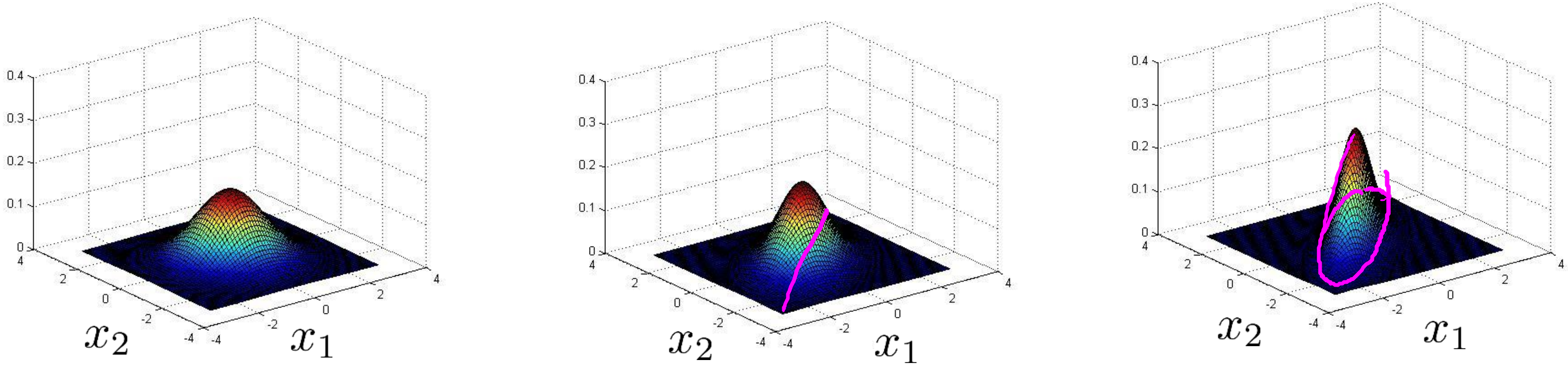
Andrew Ng

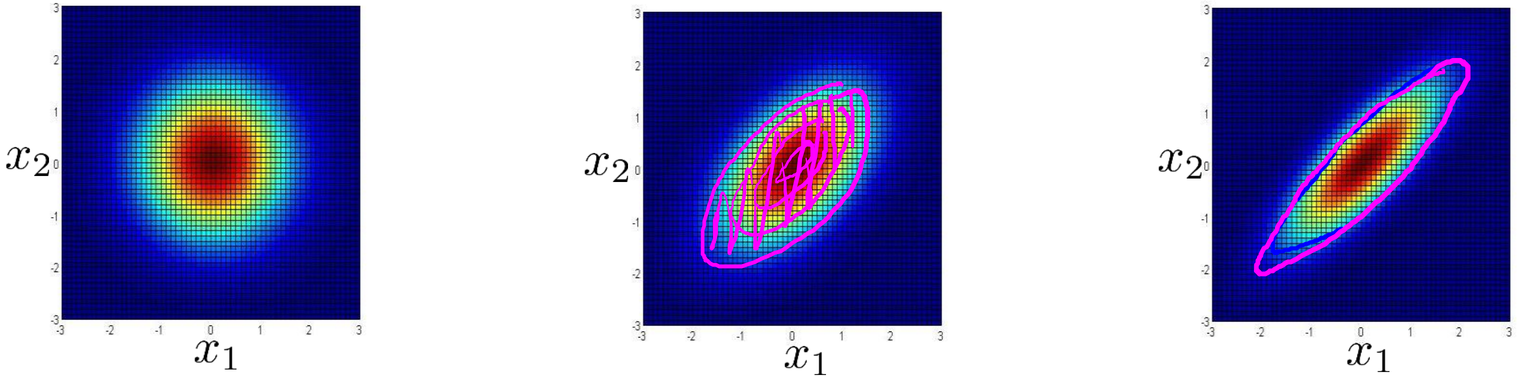
Multivariate Gaussian (Normal) examples

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

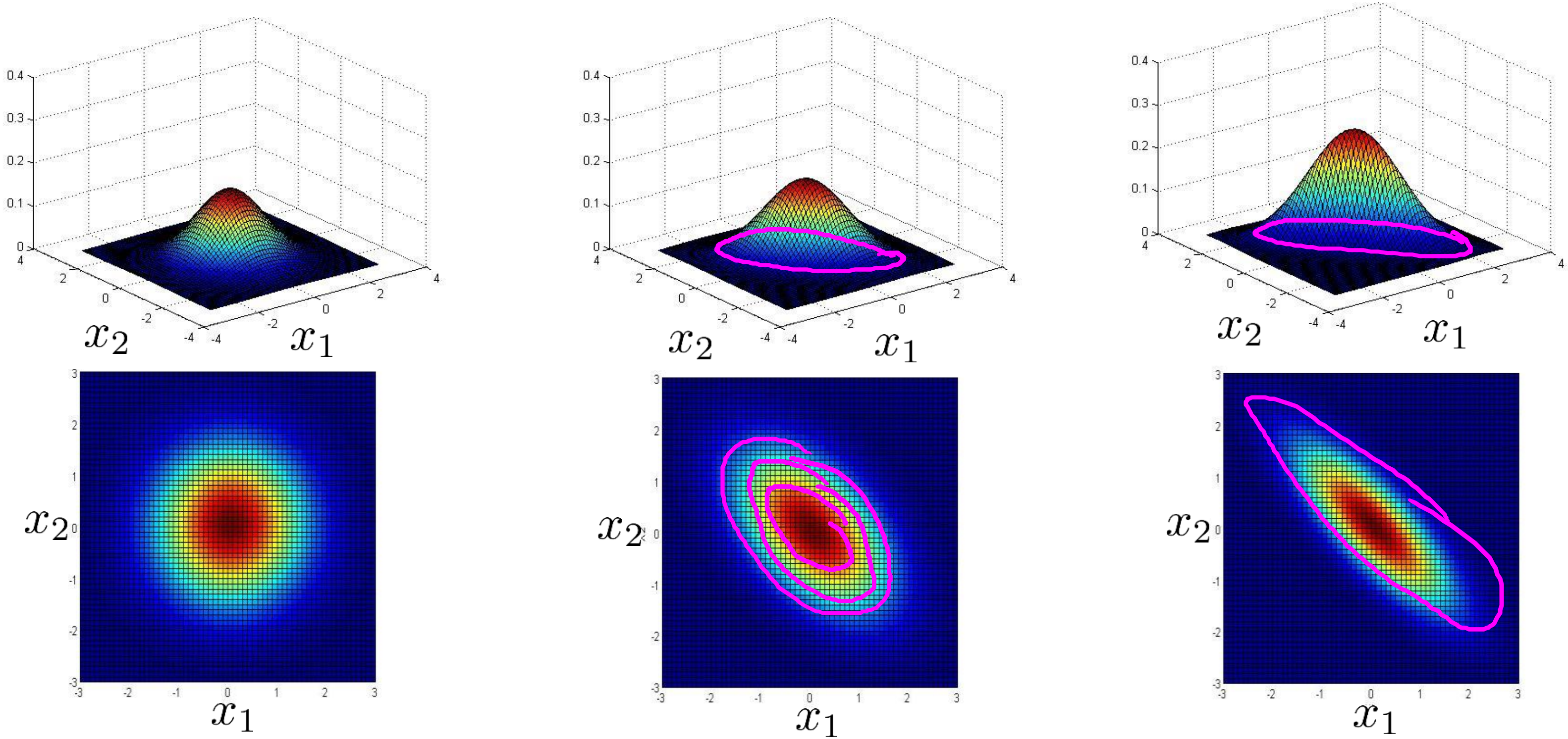
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Sigma = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$





Multivariate Gaussian (Normal) examples

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \; \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \; \Sigma = \begin{bmatrix} 1 & -0.5 \\ -0.5 & 1 \end{bmatrix} \qquad \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \; \Sigma = \begin{bmatrix} 1 & -0.8 \\ -0.8 & 1 \end{bmatrix}$$



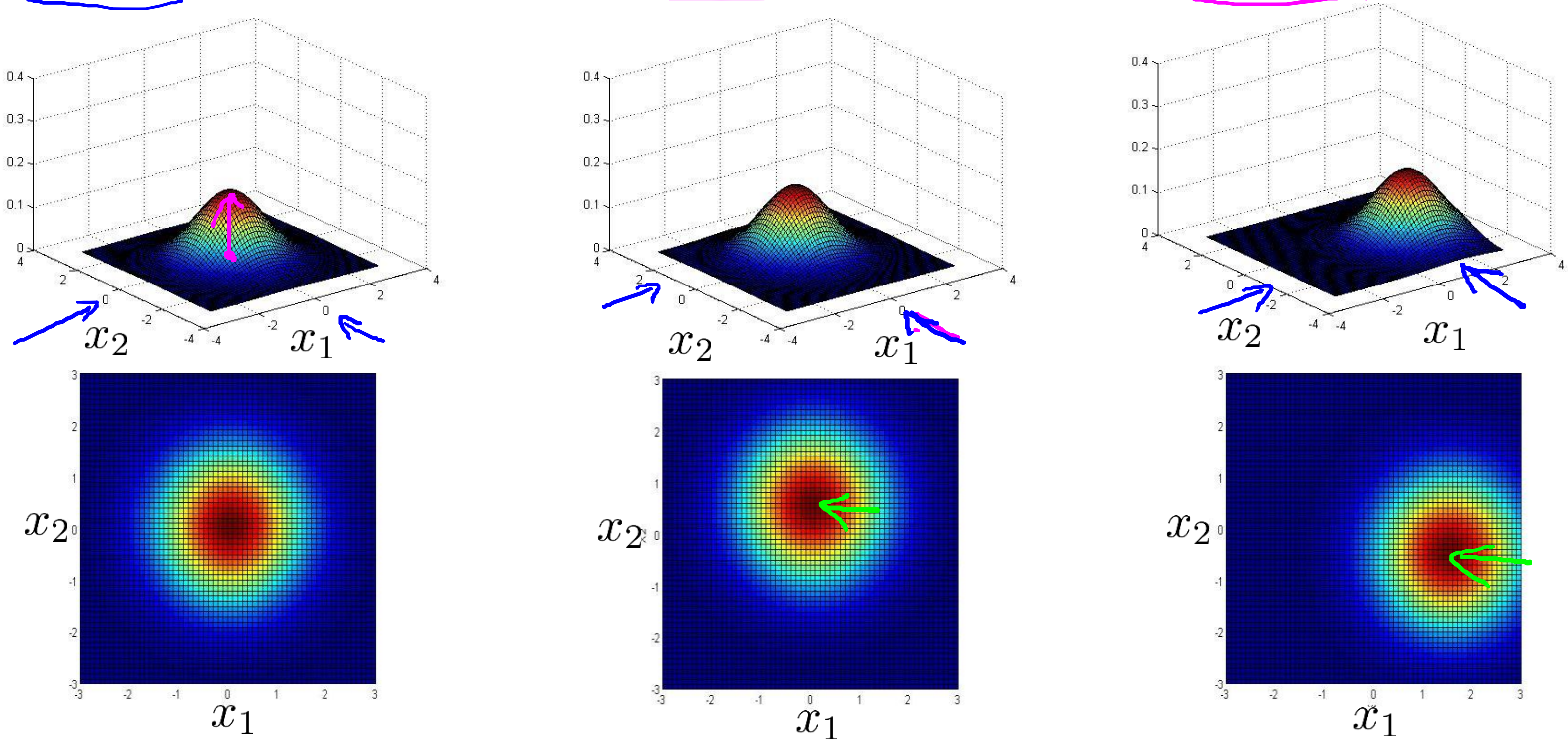
Andrew Ng

Multivariate Gaussian (Normal) examples

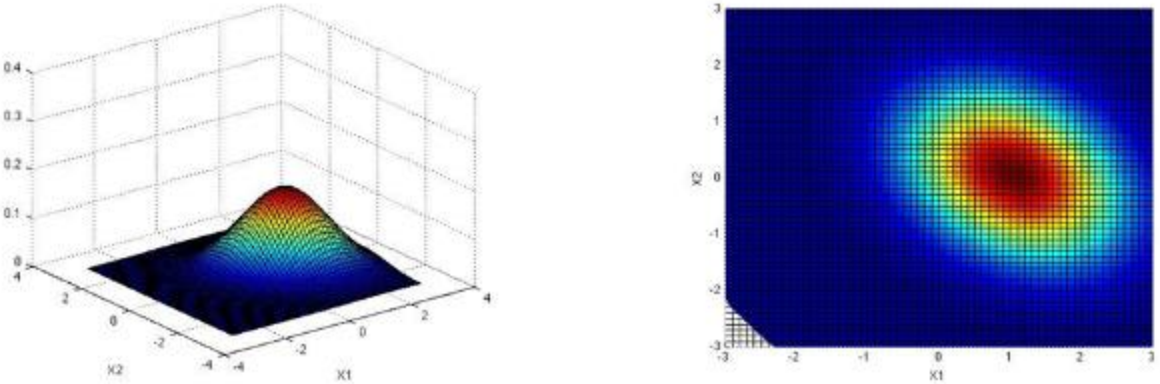
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mu = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} \quad \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mu = \begin{bmatrix} 1.5 \\ -0.5 \end{bmatrix} \quad \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$



Consider the following multivariate Gaussian:

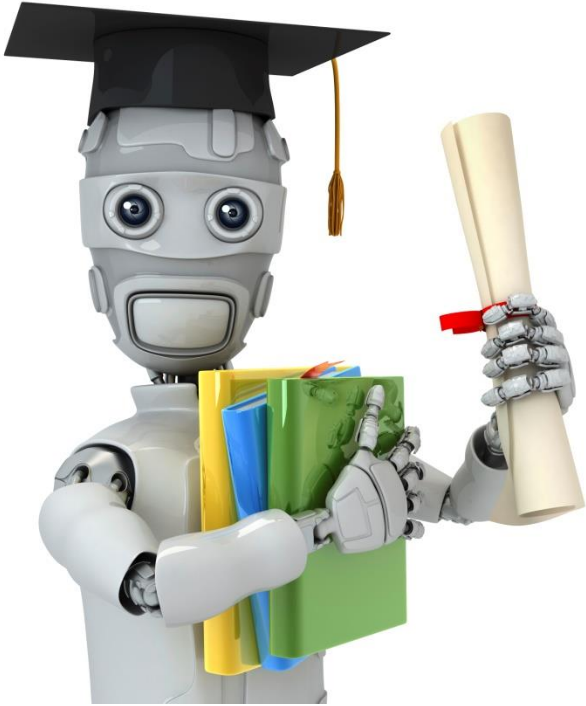


Which of the following are the μ and Σ for this distribution?

- $\mu = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \Sigma = \begin{bmatrix} 1 & 0.3 \\ 0.3 & 1 \end{bmatrix}$
- $\mu = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \Sigma = \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1 \end{bmatrix}$

  Correct Response
- $\mu = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \Sigma = \begin{bmatrix} 1 & 0.3 \\ 0.3 & 1 \end{bmatrix}$
- $\mu = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \Sigma = \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1 \end{bmatrix}$

Andrew Ng



Machine Learning

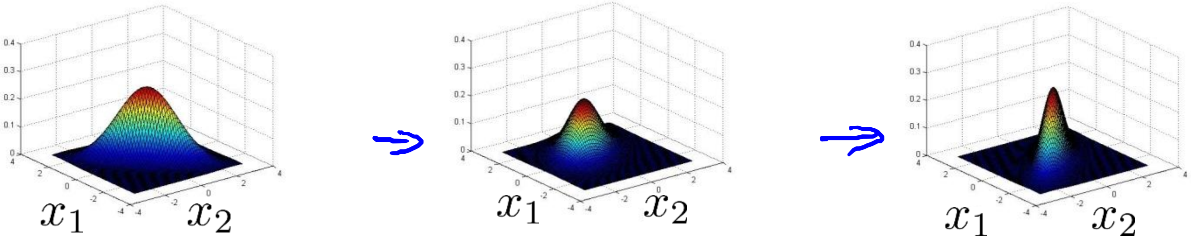
Anomaly detection

Anomaly detection using the multivariate Gaussian distribution

Multivariate Gaussian (Normal) distribution

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

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



Parameter fitting:

Given training set $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$ $x \in \mathbb{R}^n$

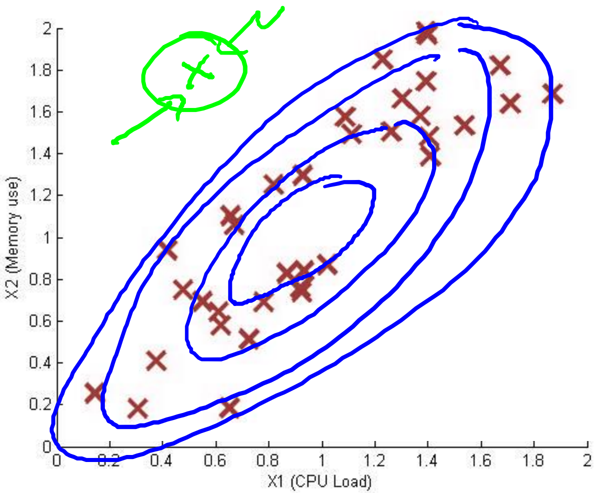
$$\boxed{\mu} = \frac{1}{m}\sum_{i=1}^{m} x^{(i)} \qquad \boxed{\Sigma} = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)}-\mu)(x^{(i)}-\mu)^T$$

Anomaly detection with the multivariate Gaussian

1. Fit model $p(x)$ by setting

$$\mu = \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$$

$$\Sigma = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - \mu)(x^{(i)} - \mu)^T$$



2. Given a new example x, compute

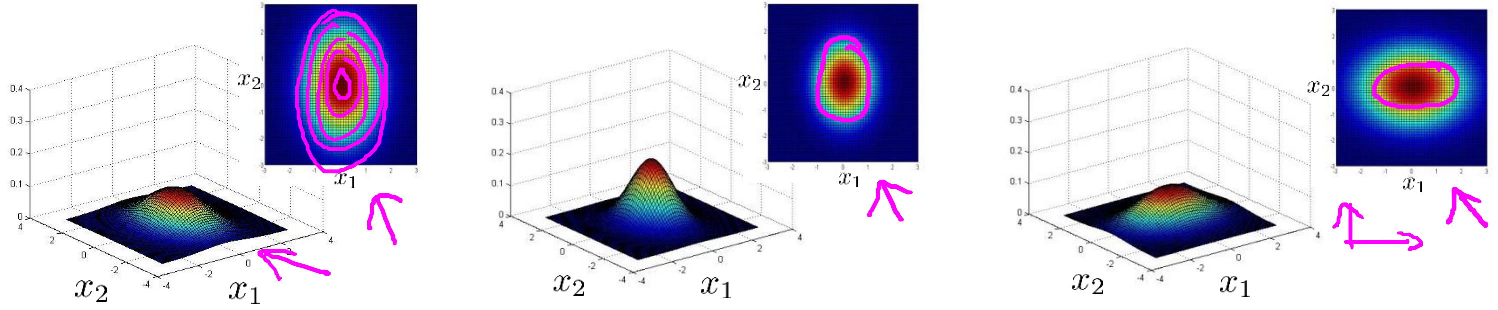
$$p(x) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}}\exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

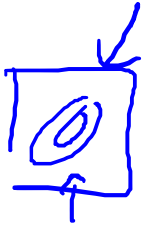
Flag an anomaly if $p(x) < \varepsilon$

Relationship to original model

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

$p(x)$





$\Sigma = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$

Corresponds to multivariate Gaussian

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

where $\Sigma = \begin{bmatrix} \sigma_1^2 & & 0 \\ & \sigma_2^2 & \\ & & \ddots & \\ 0 & & & \sigma_n^2 \end{bmatrix}$

Original model vs. Multivariate Gaussian

$p(x_1;\mu_1,\sigma_1^2) \times \cdots \times p(x_n;\mu_n,\sigma_n^2)$

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

Manually create features to capture anomalies where x_1, x_2 take unusual combinations of values.

$\boxed{X_3 = \frac{x_1}{x_2} = \frac{\text{CPU load}}{\text{memory}}}$

Automatically captures correlations between features

$\Sigma \in \mathbb{R}^{n \times n}$ Σ^{-1}

Computationally cheaper (alternatively, scales better to large n)

$n = 10{,}000$, $n = 100{,}000$

Computationally more expensive

Σ $\sim \frac{n^2}{2}$

$x_1 = \cancel{x_2}$

$\cancel{x_3} = x_4 + x_5$

OK even if m (training set size) is small

Must have $\boxed{m > n}$, or else Σ is non-invertible.

$\boxed{m \geq 10n}$

Consider applying anomaly detection using a training set $\{x^{(1)}, \ldots, x^{(m)}\}$ where $x^{(i)} \in \mathbb{R}^n$. Which of the following statements are true? Check all that apply.

- [x] The original model $p(x_1; \mu_1, \sigma_1^2) \times \cdots \times p(x_n; \mu_n, \sigma_n^2)$ corresponds to a multivariate Gaussian where the contours of $p(x; \mu, \Sigma)$ are axis-aligned.

Correct Response

- [ ] Using the multivariate Gaussian model is advantageous when m (the training set size) is very small (m < n).

Correct Response

- [x] The multivariate Gaussian model can automatically capture correlations between different features in x.

Correct Response

- [x] The original model can be more computationally efficient than the multivariate Gaussian model, and thus might scale better to very large values of n (number of features).

Correct Response