Anomaly detection example

Aircraft engine features:

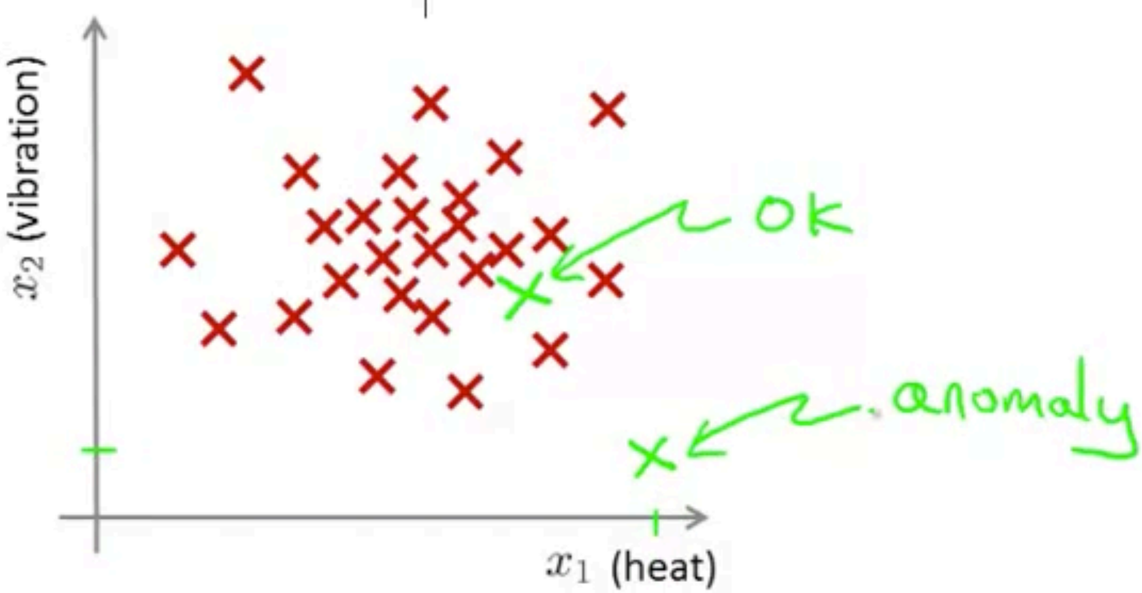
x_1 = heat generated

x_2 = vibration intensity

...

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

New engine: x_{test}

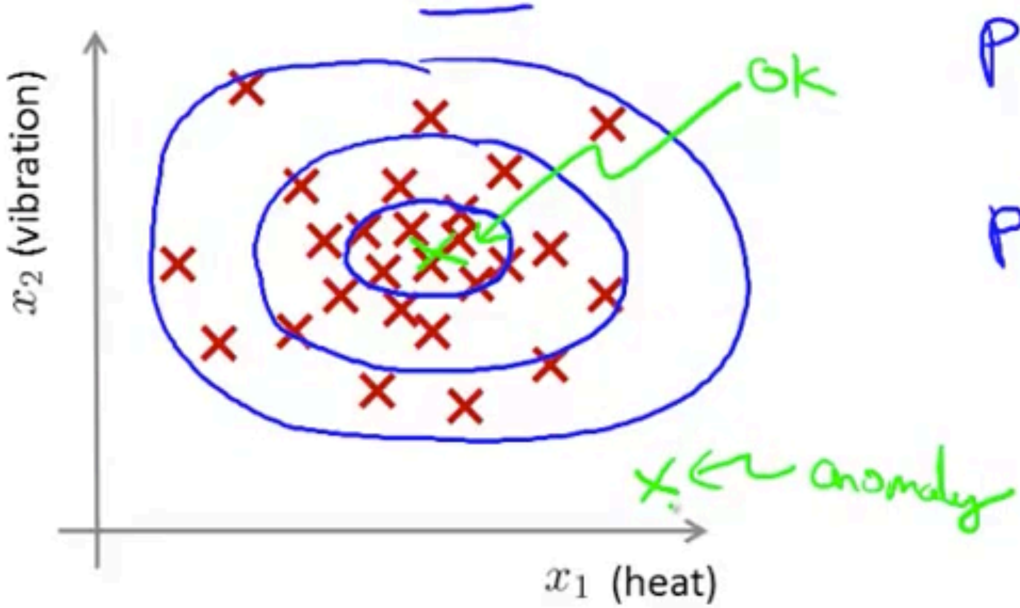


Andrew Ng

Density estimation

- 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

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 $p(x) < \varepsilon$
- 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.

    ...

x_1
x_2
x_3
x_4

$p(x)$

$p(x) < \varepsilon$

Andrew Ng

Gaussian (Normal) distribution

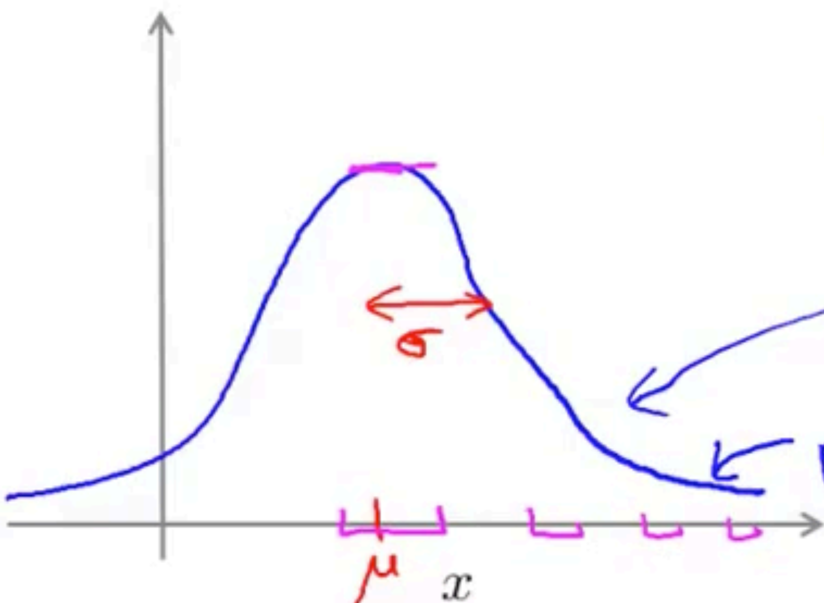
Say $x \in \mathbb{R}$. If x is a distributed Gaussian with mean μ, variance σ^2.

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

"distributed as"

σ standard deviation

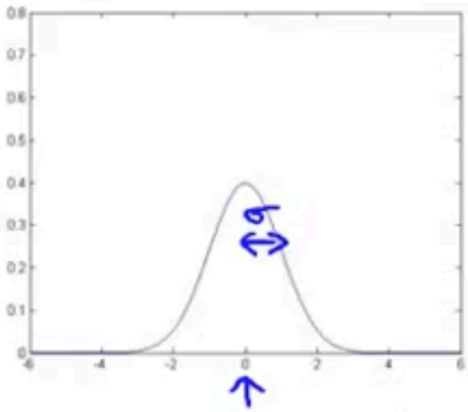
$$p(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}\,\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$



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

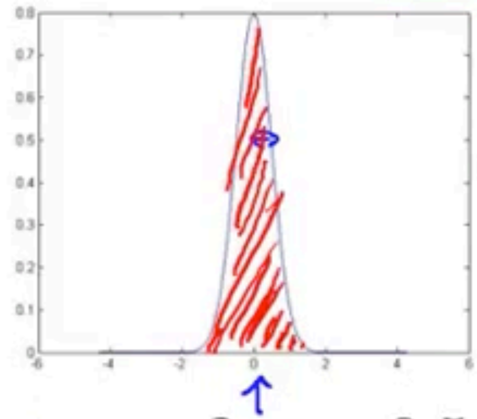
Gaussian distribution example

$\mu = 0, \sigma = 1$

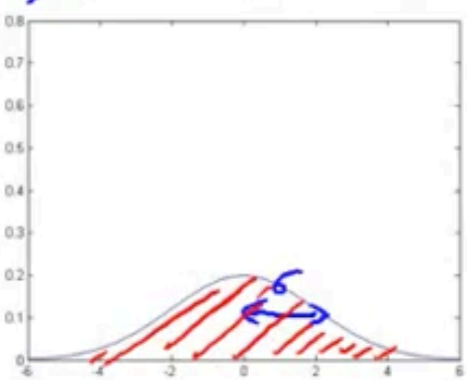


$\mu = 0, \sigma = \underline{0.5}$

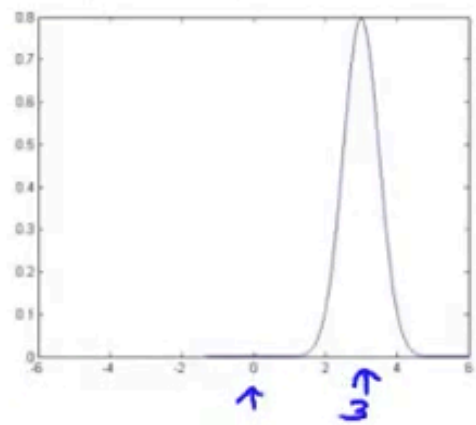
$\sigma^2 = 0.25$



$\mu = 0, \sigma = 2$



$\mu = 3, \sigma = 0.5$

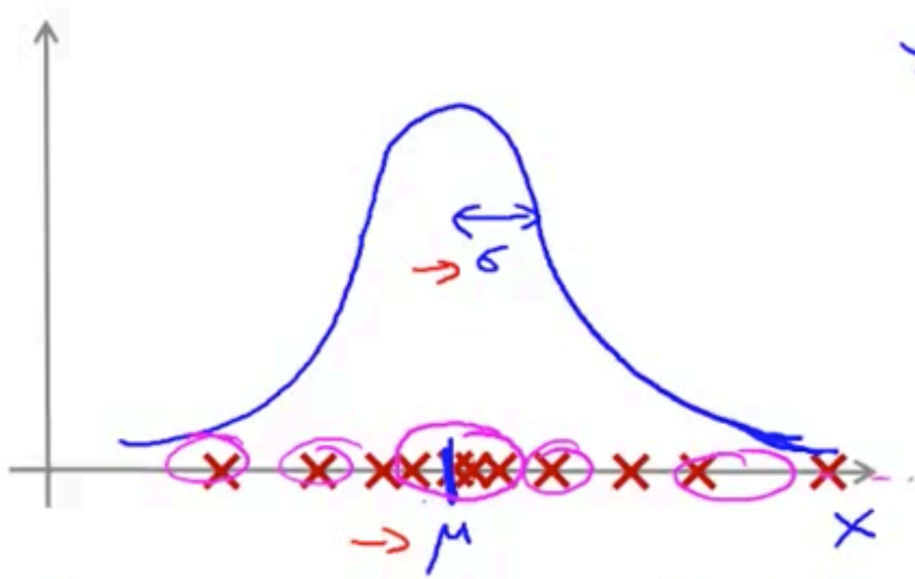


Andrew

Parameter estimation

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

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



$$\mu = \frac{1}{m} \sum_{i=1}^{m} x^{(i)} \qquad \sigma^2 = \frac{1}{m} \sum_{i=1}^{m} \left(x^{(i)} - \mu\right)^2$$

→ **Density estimation**

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

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

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

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

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

$\rightarrow p(x)$

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

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

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

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

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

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

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

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

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

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

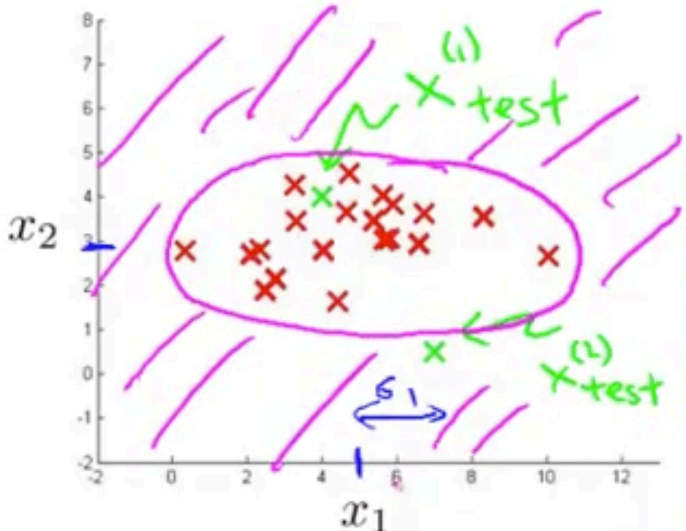
3. Given new example x, compute $p(x)$:

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

Anomaly if $p(x) < \varepsilon$

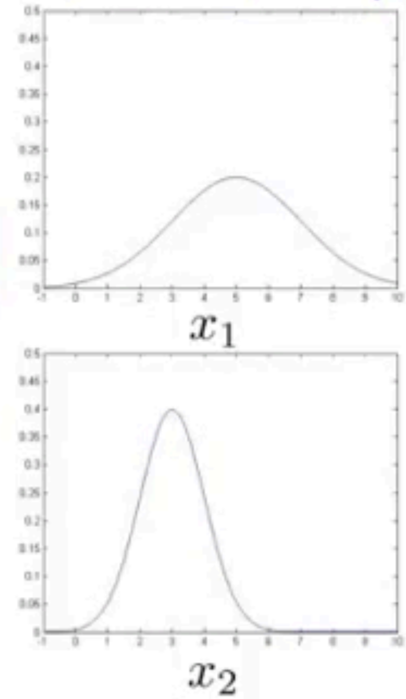
Anomaly detection example

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



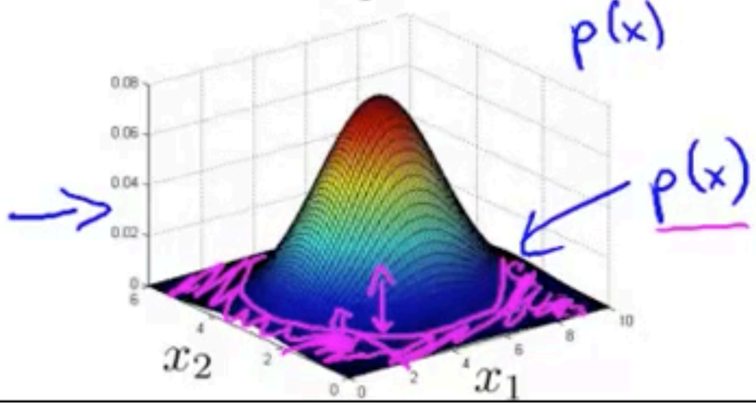
$\sigma_1^2, \sigma_2^2 = 4$

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



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

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



$\underline{\varepsilon = 0.02}$

$p(x^{(1)}_{test}) = \underline{0.0426} \geq \varepsilon$

$p(x^{(2)}_{test}) = \underline{0.0021} < \varepsilon$

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: $(x_{test}^{(1)}, y_{test}^{(1)}), \ldots, (x_{test}^{(m_{test})}, y_{test}^{(m_{test})})$

$y=1$

Andrew Ng

Aircraft engines motivating example

$\boxed{10000}$ good (normal) engines

$\boxed{20}$ flawed engines (anomalous) 2-50 $y=1$

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

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

CV: $\boxed{2000}$ good engines ($y = 0$), $\boxed{10}$ anomalous ($y = 1$)

Test: $\boxed{2000}$ good engines ($y = 0$), $\boxed{10}$ anomalous ($y = 1$)

Alternative:

Training set: $\boxed{6000}$ good engines

CV: $\boxed{4000}$ good engines ($y = 0$), 10 anomalous ($y = 1$)

Test: $\boxed{4000}$ good engines ($y = 0$), $\boxed{10}$ anomalous ($y = 1$)

Algorithm evaluation

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

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

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

$y=0$

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

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

- Fraud detection $y=1$
- Manufacturing (e.g. aircraft engines)
- Monitoring machines in a data center

⋮

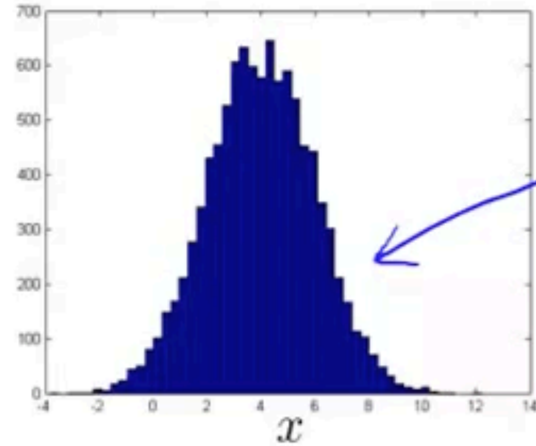
Supervised learning

- Email spam classification
- Weather prediction (sunny/rainy/etc).
- Cancer classification

⋮

Non-gaussian features

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



hist

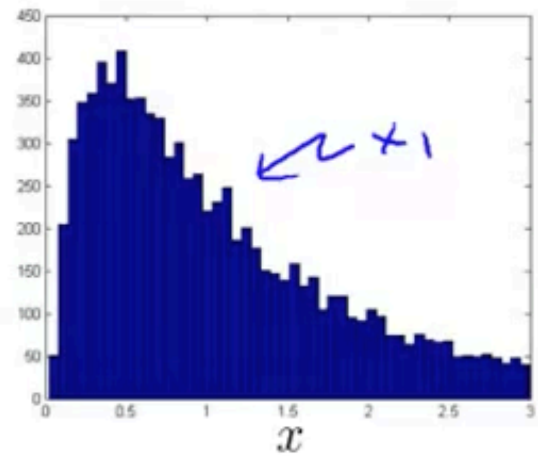
$x_1 \leftarrow \log(x_1)$

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

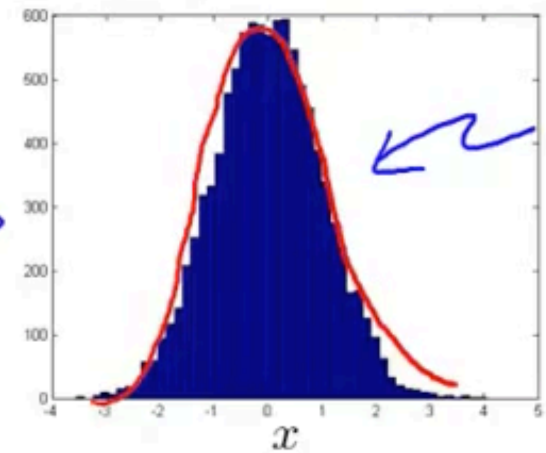
$\log(x_2 + c)$

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

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



$\xrightarrow{\log(x)}$



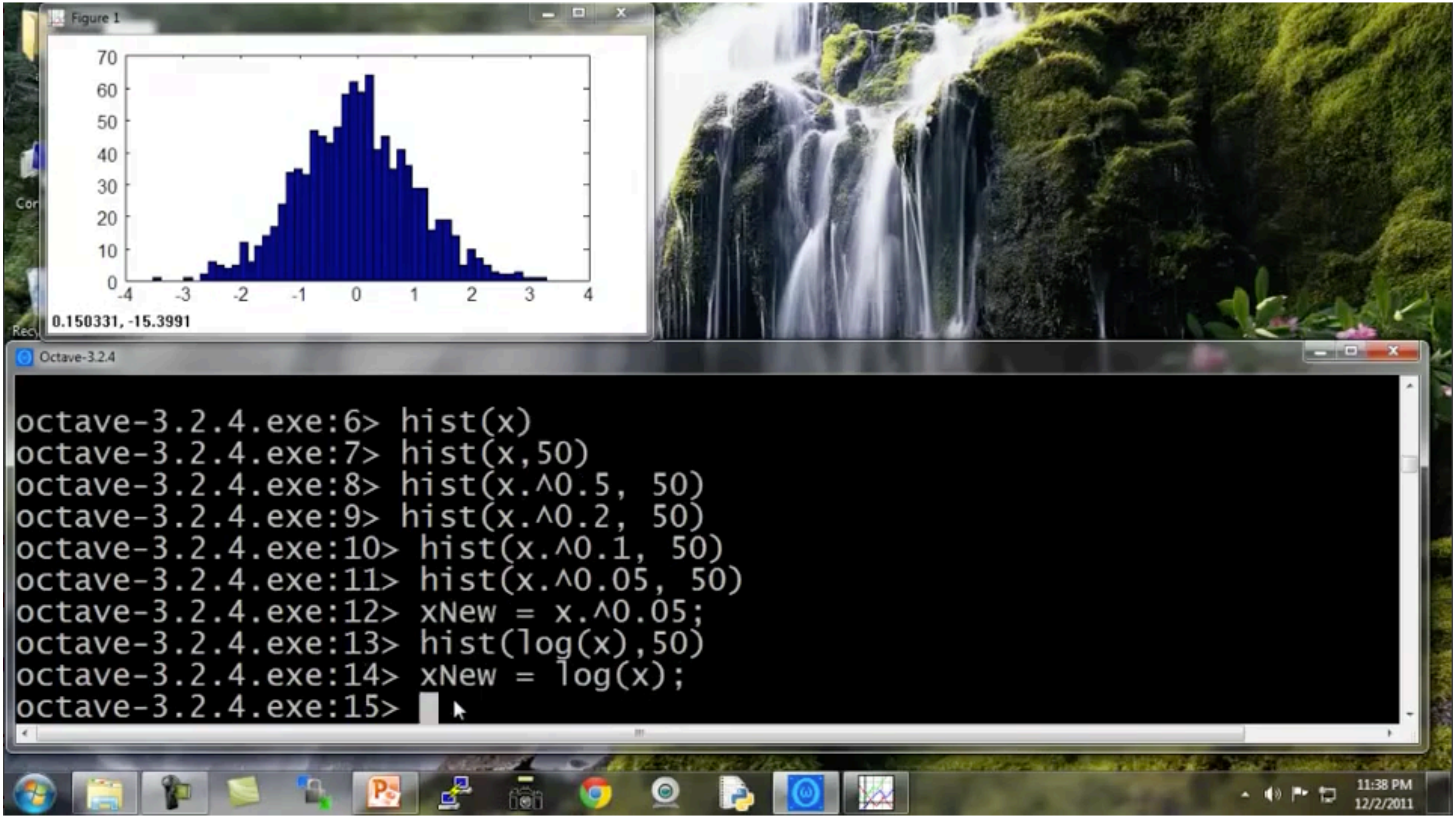


Figure 1
70
60
50
40
30
20
10
0
-4
-3
-2
-1
0
1
2
3
4
0.150331, -15.3991
Octave-3.2.4
octave-3.2.4.exe:6> hist(x)
octave-3.2.4.exe:7> hist(x,50)
octave-3.2.4.exe:8> hist(x.^0.5, 50)
octave-3.2.4.exe:9> hist(x.^0.2, 50)
octave-3.2.4.exe:10> hist(x.^0.1, 50)
octave-3.2.4.exe:11> hist(x.^0.05, 50)
octave-3.2.4.exe:12> xNew = x.^0.05;
octave-3.2.4.exe:13> hist(log(x),50)
octave-3.2.4.exe:14> xNew = log(x);
octave-3.2.4.exe:15>
11:38 PM
12/2/2011

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